ASSAM DOWN TOWN UNIVERSITY

Curriculum and Syllabus

Bachelor of Computer Application on Artificial Intelligence and Deep Learning

OUTCOME BASED EDUCATION FRAMEWORK

CHOICE BASED CREDIT SYSTEM

Version: 1.0

FACULTY OF COMPUTER TECHNOLOGY

July, 2024

PREAMBLE

Assam down town University is a premier higher educational institution which offers Bachelor, Master, and Ph.D. degree programmes across various faculties. These programmes, collectively embodies the vision and mission of the university. In keeping with the vision of evolutionary changes taking place in the educational landscape of the country, the university has restructured the course curriculum as per the guidelines of National Education Policy 2020. This document contains outline of teaching and learning framework and complete detailing of the courses. This document is a guidebook for the students to choose desired courses for completing the programme and to be eligible for the degree. This volume also includes the prescribed literature, study materials, texts, and reference books under different courses as guidance for the students to follow.

Recommended by the Board of Studies (BOS) meeting of the Faculty of Computer Technology held on dated 15/07/2024 and approved by the 51st Academic Council (AC) meeting held on dated 26/07/2024.

Chairperson, Board of Studies

Member Secretary, Academic Council

Vision

To become a Globally Recognized University from North Eastern Region of India, dedicated to the Holistic Development of Students and Making Society Better

Missions

1. Creation of curricula that address the local, regional, national, and international needs of graduates, providing them with diverse and well-rounded education.
2. Build a diverse student body from various socio-economic backgrounds, provide exceptional value-based education, and foster holistic personal development, strong academic careers, and confidence.
3. Achieve high placement success by offering students skill-based, innovative education and strong industry connections.
4. Become the premier destination of young people, desirous of becoming future professional leaders through multidisciplinary learning and serving society better.
5. Create a highly inspiring intellectual environment for exceptional learners, empowering them to aspire to join internationally acclaimed institutions and contribute to global efforts in addressing critical issues, such as sustainable development, Climate mitigation and fostering a conflict-free global society.
6. To be renowned for creating new knowledge through high quality interdisciplinary research for betterment of society.
7. Become a key hub for the growth and excellence of AdtU's stakeholders including educators, researchers and innovators
8. Adapt to the evolving needs and changing realities of our students and community by incorporating national and global perspectives, while ensuring our actions are in harmony with our foundational values and objectives of serving the community.

Programme Details

Programme Overview

The Bachelor of Computer Application (BCA) in Artificial Intelligence and Deep Learning is a specialized undergraduate programme designed to equip students with cutting-edge knowledge and skills in AI and deep learning. It blends core computer science principles with advanced AI techniques, covering programming, data structures, database management, machine learning, neural networks, natural language processing, and computer vision. With hands-on projects, industry collaborations, and real-world case studies, students gain practical expertise in designing AI-powered solutions. Emphasizing problem-solving, critical thinking, and innovation, the programme prepares graduates for AI-driven industries such as healthcare, finance, cybersecurity, robotics, and automation. By integrating theoretical knowledge with practical applications, this programme ensures industry readiness, enabling graduates to excel as AI engineers, data scientists, machine learning specialists, and software developers or pursue higher studies in AI and related fields.

I. Specific Features of the Curriculum

The Bachelor of Computer Application (BCA) in Artificial Intelligence and Deep Learning is meticulously designed to align with current industry demands. It focuses on imparting a strong foundation in AI, deep learning, and data science. Students gain practical skills essential for the AI and IT sectors, preparing them for successful careers. The curriculum emphasizes hands-on experience, ensuring students are ready to tackle real-world challenges in AI-driven applications. By integrating the latest technological trends, the programme ensures relevance in a rapidly evolving field. It aims to produce expert AI programmers and professionals. Graduates are equipped to meet future industry challenges. Continuous updates keep the curriculum effective. The programme builds a skilled workforce for the future of artificial intelligence and deep learning.

II. Eligibility Criteria:

Students should be secured 45 % in 10+2 in Arts/Science/Commerce from AHSEC or any other equivalent Examination recognized as such by the University with preferably Mathematics as one of the subjects in HS (Pass Marks).

Note: Candidate having (10+2) without mathematics/computer Science has to undertake additional bridge courses.

III. Program Educational Objectives (PEO):

PEO1: To produce graduates who have a strong foundation of knowledge and skills in the field of Computer Applications.

PEO2: To produce graduates who can provide solutions to challenging problems in their profession by applying Computer Science theory and practices.

PEO3: To produce graduates who are employable in industries/public sector/Govt.organizations or work as an entrepreneur, as well as can provide leadership and are effective in a multidisciplinary environment

IV. Program Specific Outcomes (PSO):

PSO 1: Advance the graduates with the contemporary trends in industrial/Computer Application environments and also will be capable of innovating novel solutions to prevailing problems by applying software engineering techniques and strategies.

PSO 2: Develop a holistic comprehension of Computer Science and management principles required for the application of sustainable technologies for societal development, and also will be able to communicate effectively in oral, written, visual, and graphic modes as a member and leader in a team, to manage projects in multidisciplinary environments.

PSO 3: Prepare the graduates for the state, national, and international competitive examinations with focused and updated syllabi.

V. Program Outcome (PO):

PO1 Basic Mathematical Knowledge: Apply knowledge of Mathematics & Statistics to the solution of ICT problems.

PO2 Problem Analysis, Design/Development of Solutions: Analysis the IT problem and then design/develop the solutions using Software Engineering principles that meet specified needs with appropriate consideration for cultural, societal, and environmental considerations.

PO3 Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern ICT tools including prediction and modelling with an understanding of the limitations.

PO4 Environment and Sustainability: Understand the impact of professional IT solutions in societal and environmental contexts and demonstrate knowledge of and need for sustainable development.

PO5 Ethics: Apply ethical principles and commit to professional ethics, responsibilities, and norms.

PO6 Communication: Communicate effectively with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO7 Project Management and Finance: Demonstrate knowledge and understanding of Software Engineering and Project management principles and apply these to one's work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO8 Life-long Learning: Recognize the need for and have the preparation and ability to Engage in independent and life-long learning in the broadest context of technological Change.

Total Credits to be Earned: 135

VI. Career Prospects:

After completing a degree in one of the most desirable professions in the current times, Computer Application graduates have a wide array of options at their disposal. There is no one single career option but several after Computer Application -

a. Higher Studies in Technical Field: From there one could go into the research field or complete PhD and work as a Faculty. In order to pursue M. Tech. in India, one will need to clear the GATE Examination, which will open a lot of opportunities to study Master's as well as work in PSUs. From here you can work in Government owned companies like DRDO, BARC, CSIR, ISRO, AAI, ONGC etc.
b. Technical Job: Graduates can work at a good company in various fields like Data science and Analysis, Machine Learning Engineer, Software Developer, Blockchain Developer and Engineer, Computer Network Architect, Database Administration and Management,

Security Analyst, Game Development, Digital Marketing, etc.

c. Government Services: Along with pursuing technical government jobs after appearing for GATE, engineering graduates could also work in non technical fields by applying and studying for various government exams like UPSC, SSC, Banking exams, Defence or IAS. These are very rewarding jobs as they are well paid, and very secure.

Entrepreneurship: If someone is filled with ideas with a need to create a change in the society, and like being one's own boss, one can even commence own start-up. For an entrepreneur, "opportunity" equates to something that solves a problem and addresses a need in a way that the entrepreneurs can profit from or further a cause.

EVALUATION METHODS

The student performance shall be evaluated through In-semester (Sessional) and semester-end examinations. A weightage of 40% or as prescribed by the programme shall be added to the score of the end semester examination.

A. INTERNAL ASSESSMENT:

The teacher who offers the course shall be responsible for internal assessment by conducting in-semester (sessional) examination and evaluating the performance of the students pursuing that course. The components for internal assessment are illustrated in the table given below.

SN	Components/ Examinations	Marks Allotted
1.	In-Sem Exam – I (ISE-I) (Written Examination)*	30
2.	In-Sem Exam – II (ISE-II) (Written Examination)*	30
3.	Assignment	10
4.	Presentation (SP)	10
5.	Quiz	5
6.	Class Performance based score*	5

are compulsory

Note: *Total Internal assessment should be out of 40*

INSTRUCTION

1. If a student fails to appear in the any of the component without any valid reason he/she shall be marked zero in that component. However, the course teacher at his discretion may arrange for the missed test on an alternate date for the absentee students after determining ground with genuine/valid reasons for the absent.
2. The report of evaluation of an activity towards the in-semester (sessional) component of a course shall be duly notified by the concerned course teacher within a week of completion.
3. The program coordinators should upload the in-semester marks to the ERP and forward acknowledgement of all the courses of the program to the Controller of Examinations before the start of the End-semester examination.

B. SEMESTER END EXAMINATION:

Time table for end semester examination is published at least 25 days prior to the start of Examination.

I. Pre-Examination:

Eligibility Criteria for a student to appear in University Examinations:

The student shall only be allowed to appear in a University Examination, if:

i) He/ She is a registered student of the University;

ii) He/ She is of good conduct and character;

iii) He/ She has completed the prescribed Programme of study with minimum percentage of attendance as laid down in the Regulations of the Programme concerned.

Under special cases, a student may be allowed to appear for an examination without being registered in the University but the result of the said student will be kept on hold till the registration of the concerned student is completed.

II. Admit Card:

Admit card for the examination may be downloaded through ERP where the system will generate a Unique ID Cards through online.

The University shall have the right to cancel admission for examination of any candidate on valid grounds.

III. Pattern of Question Papers:

The question paper shall follow the principles of Bloom's Taxonomy.

Table

S. N.	Level	Questions /verbs for test
1	Remember	List, Define, tell, describe, recite, recall, identify, show who, when, where, etc.
2	Understand	Describe, explain, contrast, summarize, differentiate, discuss etc.
3	Apply	Predict, apply, solve, illustrate, determine, examine, modify
4	Analyze	Classify, outline, categorize, analyze, diagrams, illustrate, infer, etc.
5	Evaluate	Assess, summarize, choose, evaluate, recommend, justify, compare etc.
6	Create	Design, Formulate, Modify, Develop, integrate, etc.

Note: No course is to be evaluated on basis of **all 6 knowledge levels.**

The format of the question paper across all the program follow a unique pattern and the total marks is 60

Table 1: Question paper pattern for End semester examination

Sl no	Question pattern	Total marks
1	MCQs (10 Questions)	10
2	2 Marks questions (10 Questions)	20
3	4 Marks questions (5 Questions)	20
4	10 Marks questions (1 Question)	10

IV. Examination Duration:

Each paper of 60 marks shall ordinarily be of two hours duration.

V. Practical Examinations, Viva-Voice etc.:

i) Practical examination shall be conducted in the presence of one external expert and one or more internal examiners.

ii) Viva-Voice, Oral examinations of the Project report, Dissertation etc. shall be undertaken by a Board of Examiners constituted by the respective Dean of Program with the advice of Supervisor(s).

VI. Procedure of Expulsion:

If any candidate is found to be using any unfair-means during the examination, the invigilator may cease his/her answer sheet and report it directly to the Officer-in-Charge. The Office-in-Charge of the center may take appropriate decisions as per the rules and procedure of the examination. The Officer-in-Charge may allow the students to write the exam with new answer sheet or may expel the student from appearing the paper depending on the nature of unfair-means. In case of Computer based test, the students may be directed to write an apology letter and sign in the prescribe expulsion form. The student may not be allowed to write that examination.

VII. Instruction to the Students:

(i) The students shall not bring to the Examination Hall, any electronic gadget used as a means of communication or record except electronic calculator, if required.

(ii) The students shall not receive any book or printed or hand written or photo copy (Xerox) or blank-paper from any other person while he/she is in the examination-room or in laboratory or in any other place to which he/she is allowed to have access during course of examination.

(iii) The students shall not communicate with any other candidate in the examination room or with any other person in and outside the examination-room.

(iv) The students shall not see, read or copy anything written by any other candidate, nor shall he/she knowingly or negligently permit any other candidate to see, read or copy anything written by him/her or conveyed by him/her.

(v) The students shall not write anything on the Question Paper or in other paper or materials during the examination, or pass any kind of paper to any other candidate in the examination-room, or to any person outside the room.

(vi) The students shall not disclose his/her identity to the examiner by writing his/her name or putting any sign / symbol in any part of his answer-script.

(vii) The students shall not use any abusive language or write any objectionable remark or make any appeal to examiner by writing in any part of his answer-script.

(viii) The students shall not detach any page from the answer-script or insert any authorized or unauthorized loose sheet into it. He /she shall also not insert any other answer-script / loose sheet by removing the pins of the origin answer-scripts and re-fixing it.

(ix) The students shall not resort to any disorderly conduct inside the examination-room or misbehave with the invigilator or any other examination official.

VIII. Provision for an Amanuensis (writer):

(i) A candidate may be provided with an Amanuensis (writer) to write down on dictation on his / her behalf on ground of his / her physical disability to write down by himself / herself due to accident or any other reason. The amanuensis may be provided till he / she recovers from the physical disability. The physical disability to write down by himself / herself must be supported by Medical Certificate from a competent Medical Officer.

(ii) The qualifications of the amanuensis so provided must not be equal or higher than that of the candidate. This is also to be supported by Certificate from the Faculty of Study where the Amanuensis is provided.

(iii) Such candidates are to be accommodated in a separate room under the supervision of an invigilator so that the fellow candidates are not disturbed in the process.

C. Credit Point:

It is the product of grade point and number of credits for a course, thus, CP = GP x CR

i. Credit:

A unit by which the course work is measured. It determines the number of hours of instructions required per week. 'Credit' refers to the weightage given to a course, usually in terms of the number of instructional hours per week assigned to it. Credits assigned for a single course always pay attention to how many hours it would take for an average learner to complete a single course successfully.

ii. Grade Point:

Grade Point is a numerical weight allotted to each Grade Letter on a 10-point scale.

iii. Letter Grade:

Letter Grade is an index of the performance of students in a said paper of a particular course. Grades are denoted by letters O, A+, A, B+, B, C, P, F and Abs. Student obtaining Grade F / Grade Abs shall be considered failed/ absent and, will be required to appear in the subsequent ESE. The UGC recommends a 10-point grading system with the following (Table: 1) Letter Grades:

(i) A Letter Grade shall signify the level of qualitative/quantitative academic achievement of a student in a Course, while the Grade Point shall indicate the numerical weight of the Letter Grade on a 10-point scale.

(ii) There shall be 08 (eight) Letter Grades bearing specific Grade Points as listed in Table 1, where the Letter Grades 'O' to 'P' shall indicate successful completion of a course.

(iii) Apart from the 08 (eight) regular Letter Grades listed in Table 1, there shall be 03 (three) additional Letter Grades, which shall be awarded if a Course is withdrawn or spanned over the next Semester or remains incomplete as stated in Table 2.

Table 2: Letter Grades and Grade Points

Letter Grade	Grade Points	Description
O	10	Outstanding
A+	9	Excellent
A	8	Very Good
B+	7	Good
B	6	Above Average
C	5	Average
P	4	Pass
F	0	Fail
Abs	0	Absent
UFM	0	Unfair Means

iv. Grade Point Average:

a. SGPA (Semester Grade Point Average)

The SGPA of a student in a Semester shall be the weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered in that Semester, irrespective of whether he/she could or could not complete the Courses. More specifically, the calculation of SGPA shall take into account the Courses graded with Letter Grades 'O' to 'F' as given in Table 1.

$$\text{SGPA} = \frac{\sum_{i=1}^{n} C_i G_i}{\sum_{i=1}^{n} C_i} \qquad (1.1)$$

The SGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.1) up to two decimal places, where n is the total number of Credit Courses registered by the student in that Semester, Gi is the Grade Point secured in the i^{th} registered Course and Ci is the Credit (weight) of that Course.

b. CGPA (Cumulative Grade Point Average)

(i) The CGPA of a student in a Semester of a Programme shall be the accumulated weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered and successfully completed so far

starting from the enrollment in the Programme. In other words,taking into account all the Courses graded with 'O' to 'P' as given in Table 1.1, generally the CGPA of a student shall be calculated starting from the first Semester of his/her enrolled Programme, while the CGPA of a lateral-entry student shall be calculated starting from the Semester of his/her enrollment.

(ii) The CGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.2) up to two decimal places, where N is the total number of Credit Courses registered and successfully completed so far by the student, Gi is the Grade Point secured in the ith completed Course and Ci is the Credit (weight)of that Course.

$$\text{CGPA} = \frac{\sum_{i=1}^{N} C_i G_i}{\sum_{i=1}^{N} C_i} \qquad (1.2)$$

(iii) The CGPA shall be convertible into equivalent percentage of marks using Equation Conversion of CGPA to percentage marks: = CGPA*10

D. Post-Examination

i. Transcript or Grade Card or Certificate:

A marking certificate shall be issued to all the registered students after every Semester. The Semester mark sheet will display the course details (code, title, number of credits, grade secured) along with total credit earned in that Semester.

ii. Grievance Readdress Mechanism:

Students with any dissatisfaction or grievance regarding the marks awarded in any of the Papers / Courses may appeal to the Controller of Examinations for remedial action such as Re-evaluation within 10 days of the declaration of result.

(i) A student has options to appeal for re-evaluation of his /her answer script to the Controller of Examination.

(ii) Application for re-evaluation / re-scrutiny of answer scripts shall be made in the definite proforma available with the Examination Office through the head of the respective departments within 10 days of declaration of the results of the respective examinations.

(iii) The Controller of Examination may appoint an examiner for re-evaluation and will consider and recognize the evaluation done by a University appointed examiner.

(iv) There shall be no provision for re-evaluation of the Practical Papers, Project Work, and Dissertation etc. However, the students fail in practical examination or viva voce and wish to appear again may apply to be evaluated can do so with the next schedule.

(v) After screening the application for re-evaluation, the CoE may send the answer scripts of the student to the examiners appointed by the CoE with the approval of Vice Chancellor.

(vi) The marks/grades achieved by the students after the re-evaluation shall be final and binding.

(vii) Fresh Marks – sheets / Grade Card shall be issued only if the candidate secures pass marks / passing grade in the re-evaluated paper.

(viii) Revaluation of answer scripts shall be deemed to be an additional facility provided to the students with a view to improving upon their results at the preceding examination result for any reason whatsoever shall not confer any right upon them for admission

to next higher class which matters always be regulated in accordance with the relevant rules or regulations framed by the University.

(ix) If as a result of revaluation of the candidate attracts the provision of condonation of deficiency, the same may be applied to his/her only for fresh attempt.

INSTRUCTION TO TEACHERS AND STUDENTS

(Teaching and Learning Methods)

In all the courses the teacher has to select topics for teacher-method which should not be less than 20 percent. The approach will be direct class room teaching through series of lectures delivering concepts using ITC facilities, white or black board. Notes may also be circulated to the students however; the students are to be involved in preparation of the notes. The teacher will be responsible in selecting the best note for circulation. The teacher- centric methodology has recently fallen out of favor because this strategy for teaching is seen to favor passive students.

1. Student- centric / Constructivist Approach:

The topics of the courses may be selected at the start of the class and assigned one topic to each of the student for studying by themselves, prepare presentations, notes etc., and present at respective class time after consultation and discussion with the course teachers. The teacher facilitate the learning of the students by guiding and providing input and explaining concepts. 60 percent of the course contents may be selected for this purpose. To avoid behavior problems, teachers must lay a lot of groundwork in student- centric classrooms. Typically, it involves instilling a sense of responsibility in students. In addition, students must learn internal motivation.

a. Project-Based Learning: The teacher may select 5 percent of topics for the purpose and may conduct visit to the laboratory for experiments or field and survey. The selection of the topic may be done considering the available facility for the purpose. However, in the final semester of each of the programme the student has to undergo a project-Based learning at least 4 months duration. This approach will help the student to think critically, evaluate, analyze, make decisions, collaborate, and more.

b. Inquiry-Based Learning: The teacher/ students are supposed to list at least five questions in each contact hour and student solve these question or search for answer which becomes the home work for the students "question-driven" learning approach. The teacher may look for the correctness of the solution or the best possible answer and discuss in the successive class. This will help in the preparation for various competitive examination and develop a habit for search for solutions.

c. Flipped Classroom: About 10 percent of the course content has to be completed by this method. In this approach the students are asked to watchvideo or lecture prepared by the teacher or any video available (relevant to the course). A set of questions may be given to the students for searching answers by the students. The idea is that students should have more time in-classroom focusing on achieving these higher levels of thinking and learning. The Flipped classroom is also an acronym. The letters FLIP represent the four pillars included in this type of learning: Flexible environment, Learning culture shift, Intentional content, and Professional educator. As you can see, the second pillar refers to a culture shift from the traditional approach where students are more passive to an approach where students are active participants. As a result, this approach is also a student- centric teaching method.

d. Cooperative Learning: The remaining five percent has to be completed by cooperative learning approach. In this approach the students are allotted with problems. During the library hours the student along with the teacher visits library search probable solution for the assigned problem. The same has to be done in group so that the students

discuss among themselves for the appropriate answers. Essentially, cooperative learning believes that social interactions can improve learning. In addition, the approach recreates real-world work situations in which collaboration and cooperation are required.

The percentage categorization for the completion of a theory course

Teacher- centric or Direct Classroom Teaching: Delivery by series of lectures	20%
Student- centric Approach, Student present and deliver lectures in presence of teacher and supervised by teacher	60%
Student visit fields or perform experiments or teacher perform demonstration	05%
Flipped Classroom approach	10%
Cooperative learning approach	05%

Inquiry based approach has to be followed in all of the classes

Teacher has to distribute the topics to be considered for teaching by the above-mentioned approaches and prepare lesson plan for execution and maintain a file.

Breakdown of Credits

Sl. No	Category		Total number of Credits
1	University Core(UC)	Skill Enhancement Course (SEC)	4
		Ability Enhancement Course(AEC)	-
		Field Training	3
		Discipline Specific Elective (DSE)	-
		Value Added Course (VAC)	-
2	University Elective (UE)	Multidisciplinary Course (MDC)	13
		Value Added Course (VAC)	3
3	Program Core(PC)	Discipline Specific Core(DSC)	80
		Field Training	-
		Research /Industry Internship	14
		Summer Internship	3
4	Program Elective (PE)	Discipline Specific Elective (DSE)	11
		Value Added Course (VAC)	4
5	Faculty Core(FC)	Skill Enhancement Course (SEC)	-
		Ability Enhancement Course(AEC)	-
Total			**135**

Breakdown by categories of courses

Sl no	Category	Credits	%
1	Science	7	5.18
2	Engineering	127	94.07
3	Commerce and Management	1	0.75
Total		**135**	**100%**

SEMESTER WISE COURSE DISTRIBUTION

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM1101R	DIGITAL ELECTRONICS	DSC (Minor)	3	3	0	0	0	0	0	40	60	0	100
2	24BCAM1102R	SOFTWARE FOUNDATION AND PROGRAMMING (C++)	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAM1105R	MINI PROJECT I	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
4	24BCAM1104R	FUNDAMENTALS OF COMPUTER APPLICATIONS	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
5	24UBPD1102R	ELEMENTARY ENGLISH	AEC	2	0	0	4	0	0	0	0	0	100	100
6	24BCAM1103R	MATHEMATICS FOR COMPUTER APPLICATION	MDC	3	3	0	0	0	0	0	40	60	0	100
7	24MOSY1101R 24MOSY1102R 24MOSY1103R	1. FOUNDATION OF DIGITAL MARKETING AND E-COMMERCE 2. INTERNATIONAL LEADERSHIP AND ORGANIZATIONAL BEHAVIOUR 3. INTRODUCTION TO GIT AND GITHUB	VAC	2	0	0	0	0	0	0	0	0	100	100
8	24UBCC1101	CO-CURRICULAR ACTIVITIES	Co-Curricular	1	0	0	0	4	0	0	0	0	100	100
Credit Without Brdge Course				**20**	**11**	**0**	**8**	**12**	**0**	**0**	**160**	**240**	**600**	**1000**
9	24BCAM1106R	ALGORITHMS FOR PROBLEM SOLVING	Bridge Course	2	1	0	2	0	0	0	40	60	100	200
10	24BCAM1107R	FUNDAMENTALS OF STATISTICS	Bridge Course	2	1	0	2	0	0	0	40	60	100	200
Credit with Brdge Course				**24**	**13**	**0**	**12**	**12**	**0**	**0**	**240**	**360**	**800**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM1201R	DATA STRUCTURE USING C	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAM1202R	COMPUTER ORGANIZATION AND ARCHITECTURE	DSC (Minor)	2	2	0	0	0	0	0	40	60	0	100
3	24BCAM1203R	INTRODUCTION TO WEB TECHNOLOGY	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAM1205R	CLOUD FUNDAMENTALS	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAM1206R	MINI PROJECT II	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
5	24UBES1001R	ENVIRONMENTAL STUDIES	MDC	2	2	0	0	0	0	0	40	60	0	100
6	24UBPD1203R	ADVANCED IMPLICIT ENGLISH	AEC	2	0	0	4	0	0	0	0	0	100	100
7	24BCAM1204R	FUNDAMENTALS OF MANAGEMENT	VAC	2	1	0	2	0	0	0	40	60	100	200
8	24UBEC1201	EXTRA CURRICULAR	Extra-Curricular	1	0	0	0	0	0	4	0	0	100	100
Credit Without Brdge Course				**23**	**15**	**0**	**10**	**8**	**0**	**4**	**240**	**360**	**700**	**1300**
9	24BCAM1206R	BASICS OF GRAPH THEORY	Bridge Course	2	1	0	2	0	0	0	40	60	0	100
10	24BCAM1207R	MATHEMATICS FOR PROGRAMMING DESIGN	Bridge Course	2	1	0	2	0	0	0	40	60	0	100
Credit with Brdge Course				**27**	**17**	**0**	**14**	**8**	**0**	**4**	**320**	**480**	**900**	**1700**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM2101R	DATABASE MANAGEMENT SYSTEMS	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAM2102R	OBJECT ORIENTED PROGRAMMING PARADIGM IN JAVA	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAM2103R	MINI PROJECT III	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
4	24BCAM2104R	WEB AND MOBILE PROGRAMMING TECHNOLOGIES	DSC (Minor)	3	2	0	2	0	0	0	0	0	100	100
5	24BCAO2105R	Data Visualization	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
6	24UULS2002R	BLSS	MDC	1	0	0	2	0	0	0	0	0	100	100
7	24UUFL2002R	FL	MDC	1	0	0	2	0	0	0	0	0	100	100
8	24MOSY2101R 24MOSY2102R 24MOSY2103R	1. REACT BASICS 2. SPRING FRAMEWORK SPECIALIZATION 3. JAVASCRIPT BASICS	MDC	1	0	0	0	0	0	0	0	0	100	100
9	24UBPD2103R	ADVANCED ENGLISH LANGUAGE FOR EXCELLENCE	AEC	2	0	0	4	0	0	0	0	0	100	100
10	24BCAM2106R	LOGICAL RESONING FOR COMPUTER SCIENCE	SEC	3	2	0	2	0	0	0	0	0	100	100
11	24BCAO2107R	FIELD BASED LEARNING	Field Training	1	0	0	0	0	0	8	0	0	100	100
				25	**12**	**0**	**18**	**8**	**0**	**8**	**120**	**180**	**1100**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM2201R	DESIGN AND ANALYSIS OF ALGORITHMS	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100
2	24BCAM2202R	BASICS OF PYTHON PROGRAMMING	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAM2203R	COMPUTER NETWORKS	DSC (Major)	3	2	0	2	0	0	0	40	60	100	200
4	24BCAM2204R	OPERATING SYSTEM	DSC (Major)	3	2	0	2	0	0	0	40	60	100	200
5	24BCAM2205R	Predictive Analysis	DSC (Major)	3	2	0	2	0	0	0	40	60	0	100
6	24UULS2201R	BAS	VAC	1	0	0	2	0	0	0	0	0	100	100
7	24UCDL2201R	COMPUTATIONAL SYSTEMS AND DIGITAL WORLD	VAC	1	0	0	0	4	0	0	0	0	100	100
8	24UBPD2203R	PDP	AEC	2	0	0	4	0	0	0	0	0	100	100
9	24BCAM2207R	QUANTITIVE APITUTUTE FOR COMPUTER SCIENCE	SEC	2	1	0	2	0	0	0	40	60	0	100
10	24BCAM2208R	Mini Project I	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
				22	**13**	**0**	**16**	**4**	**0**	**0**	**240**	**360**	**600**	**1200**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM3101R	ADVANCE APPLICATION DEVELOPMENT	DSC (Major)	4	2	0	2	0	0	0	40	60	100	200
2	24BCAM3102R	UI/UX DESIGN	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAM3103R	Aritificial Intelligence and Deep Learning	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	24BCAM3104R	PROJECT - I	Project / Research	6	0	0	0	4	6	0	0	0	100	100
6	24BCAM3105R	DOMAIN TRAINING	SEC	2	1	0	2	0	0	0	40	60	100	200
7	24BCAM3106R	DESIGN THINKING AND ENTERPREUSHIP	SEC	2	1	0	2	0	0	0	40	60	100	200
8	24BCAM3107R	SUMMER INTERNSHIP	Summer Internship	1	0	0	0	16	0	0	0	0	100	100
				23	**13**	**0**	**12**	**20**	**6**	**0**	**240**	**360**	**800**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAM3201R	COMPUTER VISION	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAM3202R	INTRODUCTION TO CYBER SECURITY	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAM3203R / 24BCAM3204R	PE I	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAM3205R / 24BCAM3206R	PE III	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	24BCAM3207R	PROJECT - II	Project / Research	6	0	0	0	8	24	0	0	0	100	100
				22	**12**	**0**	**8**	**8**	**24**	**0**	**160**	**240**	**500**	**900**

***IA: Internal Assessment, SEE: Semester End Examination,**
PE: Practical Examination

SEMESTER – I									
Course Title	**Digital Electronics**								
Course code	**24BCAM1101R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-Requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To familiarize with the different number systems, logic gates, and combinational and sequential circuits utilized in the different digital circuits and systems.								
	2. The course will help in design and analysis of the digital circuit and system.								
	3. To impart how to design Digital Circuits.								
CO1	Interpret different number systems, binary codes and Boolean algebra to minimize logic expressions								
CO2	Develop K-maps to minimize and optimize logic functions up to 5 variables								
CO3	Infer the knowledge about various logic gates and logic families and analyze basic circuits of these families								
CO4	Design and implement Combinational and Sequential logic circuits.								
CO5	Describe and compare various memory systems, shift registers and analog to digital and digital to analog conversion circuits								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Fundamentals of Digital Systems and logic families:** Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Boolean algebra, examples of IC gates, number systems-binary, signed binary, octal hexadecimal number, binary arithmetic, one's and two's complements arithmetic, codes, error detecting and correcting codes, characteristics of digital ICs, digital logic families, TTL, Schottky TTL and CMOS logic, interfacing CMOS and TTL, Tri-state logic	**10**	Remembering and understanding the different logic gates and applying them for different circuit.	1,2,3
II	**Combinational Digital Circuits:** Standard representation for logic functions, K-map representation, simplification of logic functions using K-map, minimization of logical Faculty of Engineering& Technology| Assam down town University functions. Don't care conditions, Multiplexer, DeMultiplexer/Decoders, Adders, Subtractors, BCD arithmetic, carry lookahead adder, serial adder, ALU, elementary ALU design, popular MSI chips, digital comparator, parity checker/generator, code converters, priority encoders, decoders/drivers for display devices, QM method of function realization	**12**	Applying and analyzing different digital circuits.	3,4
III	**Sequential circuits and systems:** A 1-bit memory, the circuit properties of	**10**	Applying and analyzing different sequential circuits	3,4

	Bistable latch, the clocked SR flip flop, J- K-T and D types flip flops, applications of flip flops, shift registers, applications of shift registers, serial to parallel converter, parallel to serial converter, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, counters design using flip flops, special counter IC's, asynchronous sequential counters, applications of counters.			
IV	**A/D and D/A Converters:** Digital to analog converters: weighted resistor/converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter ICs, sample and hold circuit, analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D converter, A/D converter using voltage to frequency and voltage to time conversion, specifications of A/D converters, example of A/D converter ICs	**7**	Describe, Differentiate, and analyzing A/D and D/A converters	2,4
V	**Semiconductor memories and Programmable logic devices:** Memory organization and operation, expanding memory size, classification and characteristics of memories, sequential memory, read only memory (ROM), read and write memory(RAM), content addressable memory (CAM), charge de coupled device memory (CCD), commonly used memory chips, ROM as a PLD, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).	**6**	Understanding, analyzing and explaining different memories and logic.	1,2,4

TEXT BOOKS:

T1: Modern Digital Electronics, R. P. Jain, McGraw Hill Education, 2009.

REFERENCE BOOKS:

R1: Digital logic and Computer design, M. M. Mano, Pearson Education India, 2016.

R2: Fundamentals of Digital Circuits, A. Kumar, Prentice Hall India, 2016.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Interpret different number systems, binary codes and Boolean algebra to minimize logic expressions	**1 and 3**
2	Develop K-maps to minimize and optimize logic functions up to 5 variables	**2,3, and 4**
3	Infer the knowledge about various logic gates and logic families and analyze basic circuits of these families	**1,2, and 7**
4	Design and implement Combinational and Sequential logic circuits.	**1,2,4, and 7**
5	Describe and compare various memory systems, shift registers and analog to digital and digital to analog conversion circuits	**1,2,5, and 7**

SEMESTER – I									
Course Title	**Software Foundation with C++**								
Course code	**24BCAM1102R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To introduce fundamental programming concepts using C++. 2. To develop proficiency in problem-solving and algorithm development. 3. To understand the principles of object-oriented programming (OOP) and their implementation in C++. 4. To gain practical experience in software development and debugging using C++.								
CO1	Understand basic programming constructs and syntax in C++.								
CO2	Develop algorithms and implement them using C++.								
CO3	Design and implement classes and objects following object-oriented principles.								
CO4	Apply debugging techniques to identify and resolve errors in C++ programs.								
CO5	Develop small-scale software projects using C++.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to C++ Programming:** Overview of programming languages, Introduction to C++ programming language, Basic syntax and structure of C++ programs, Data types, variables, and constants, Input and output operations in C++	**8**	Describe, illustrate, and explain computer, algorithms, flowchart, syntax, data types, variables, constants, i/o operations.	1, 2
II	**Control Structures and Functions:** Decision-making statements (if, else if, switch) Looping statements (for, while, do-while) Functions and function prototypes Function overloading Recursion and its applications	**8**	Describe, illustrate, and explain Decision making statements, loop , function and recursion.	1, 2

III	**Arrays and Pointers:** Arrays and their manipulation, Pointers and their applications, Dynamic memory allocation, Pointer arithmetic	**8**	Describe, illustrate, and explain Array, pointer and dynamic memory allocation	1, 2
IV	**Object-Oriented Programming (OOP) in C++:** Introduction to object-oriented programming, Classes and objects, Encapsulation, inheritance, and polymorphism, Constructors and destructors, Operator overloading	**8**	Describe, illustrate, and explain oop terminologies.	1, 2
V	**File Handling, Error Handling, and Software Development Practices:** File handling in C++, reading from and writing to files, Exception handling using try, catch, throw, Standard Template Library (STL) for error handling,Software development life cycle, Documentation and commenting practices, Debugging techniques and tools	**8**	Describe, illustrate, and explain file handling	1, 2

TEXT BOOKS:

T1: "C++ Primer" by Stanley B. Lippman, JoséeLajoie, and Barbara E. Moo

T2:"Programming: Principles and Practice Using C++" by Bjarne Stroustrup

REFERENCE BOOKS:

R1: "Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14" by Scott Meyers

R2: "The C++ Programming Language" by Bjarne Stroustrup

R3: "Accelerated C++: Practical Programming by Example" by Andrew Koenig and Barbara E. Moo

OTHER LEARNING RESOURCES:

- C++ Documentation - https://devdocs.io/cpp/
- Cpp Reference - https://en.cppreference.com/w/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand basic programming constructs and syntax in C++.	**1,2 & 8**
2	Develop algorithms and implement them using C++.	**1,2 & 8**
3	Design and implement classes and objects following object-oriented principles.	**1,2 &8**
4	Apply debugging techniques to identify and resolve errors in C++ programs.	**1,2 ,4, 5,6,7 & 8**
5	Develop small-scale software projects using C++.	**1,2 ,4**

SEMESTER – I										
Course Title	**MATHEMATICS FOR COMPUTER APPLICATION**									
Course code	**24BCAM1103R**	**Total credits: 4** **Total hours: 45T**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**	
			3	**0**	**0**	**0**	**0**	**0**	**4**	
Pre-requisite	**Nil**	**Co-requisite**	**Nil**							
Programme	**Bachelor of Computer Application**									
Semester	**Fall/ I semester of the first year of the program**									
Course Objectives (Minimum 3)	1. To make understand to evaluate definite and improper integrals. Apart from these some special functions like Beta and Gamma function must be introduced. 2. To provide the application of differential and integral calculus. 3. To make understand the convergence and divergence of sequence and series.									
CO1	Illustrate the foundation concepts in Mathematics like Matrices, Set, relations and functions									
CO2	Outline Differential Integration and Differentiation and with this background.									
CO3	Learn about Multivariable Calculus for Limit, continuity and partial derivatives									
CO4	Describe matrices like Inverse and rank of a matrix, rank-nullity theorem									
CO5	Apply the foundation concepts in programming and analytical subjects.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	***Calculus:*** Evolutes and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions.	**8**	Describe, illustrate, and explain Integrals and its applications	1, 2
II	***Calculus:*** Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; indeterminate forms and L'Hospital's rule; Maxima and minima..	**12**	Describe, illustrate, and explain various theorems of calculus	1, 2
III	***Sequences and series:*** Convergence of sequence and series, tests for convergence; Power series, Taylor's series, series for exponential, trigonometric and logarithm functions; Fourier series: Half range	**8**	Describe, illustrate, and explain numbers series	1, 2

	sine and cosine series, Parseval's theorem.			
IV	***Multivariable Calculus (Differentiation):*** Limit, continuity and partial derivatives, directional derivatives, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers; Gradient, curl and divergence.	**8**	Describe, illustrate, and explain multivariable calculus	1, 2
V	***Matrices:*** Inverse and rank of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Determinants; Eigenvalues and eigenvectors; Diagonalization of matrices; Cayley-Hamilton Theorem, and Orthogonal transformation.	**9**	Describe, illustrate, and explain types of matrices	1, 2

TEXT BOOKS:

T1:G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.

T2:Erwinkreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.

T3:Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.

REFERENCE BOOKS:

R1: N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.

R2: B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate the foundation concepts in Mathematics like Matrices, Set, relations and functions	1 & 6
2	Outline Differential Integration and Differentiation and with this background.	1 & 6
3	Learn about Multivariable Calculus for Limit, continuity and partial derivatives	1 & 6
4	Describe matrices like Inverse and rank of a matrix, rank-nullity theorem	1 & 6
5	Apply the foundation concepts in programming and analytical subjects.	1 & 6

SEMESTER – I										
Course Title	**Fundamentals of Computer Applications**									
Course code	**24BCAM1104R**	**Total credits: 4** **Total hours: 45L+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To acquire the basic knowledge of Computer Systems and their uses. 2. To know about Application Software and Programming environments 3. Identify and protect from computer viruses and online threats.								
CO1	Explain the working of a computer and its various components.								
CO2	Discuss the use of Software and programming in a computer system.								
CO3	Outline the basic concepts of Computer Networks and Internet Protocols.								
CO4	Develop Proficiency in Identifying different types of computer viruses, worms, and malware to enhance threat awareness.								
CO5	Apply the role of various Software packages for Office Automation								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Basics of Computer** What is Computer and its History, Generation of Computers, Classification of Computers; Components of a Digital Computer: CPU, Memory, I/O devices; Storage Devices (RAMs, ROMs, Hard Disk etc.), Backup System and it is use.	**8**	Understand and explain the different types of computers, working of computer, and identify the various components.	1, 2
II	**Software and Languages** What is Computer Software and its need, Types of Computer Software's; Computer languages, Generation of Computer Languages, Classification of Computer Languages.	**8**	Understand the basics of software and programming concepts, identify different types of programming language.	1, 2
III	**Networks** LAN, MAN, WAN, Internet, Intranet, Browsers, Basic Internet Protocols (http, TELNET, FTP).	**8**	Understand the working of an Internet and Network. Identify and apply the different types of networks.	1, 2, 3

IV	**Computer Viruses and Security**. Type of Virus, Worms, Malware, Firewalls, Phishing, Anti-Virus.	**8**	Understand and identify different types of viruses and worms, firewall and phishing.	1, 2
V	**Software packages for Office Automation** (Any available standard Office Suite may be referred) Word Processing software, Spreadsheet software, Presentation software, Database software, Drawing and Graphics manipulation software.	**12**	Describe the different types of MS Office package. Apply and creating different files using MS Office Software.	3, 5
Practical Components				
Practical 1	Identify the components of a computer	1	Identifying and understanding the components of a computer.	2
Practical 2	Showing and explain the working process of the computer's components.	1	Remembering and understanding the working process of computer.	1,2
Practical 3	A computer is not powering on. List and demonstrate the steps you would take to diagnose and fix the issue.	1	Applying the knowledge of computers to fix some issues related to computers.	3
Practical 4	Access the BIOS/UEFI on a computer and describe how to change the boot order to prioritize booting from a USB drive.	1	Understanding and applying the knowledge on BIOS.	2,3
Practical 5	Demonstrate how to install an operating system (e.g., Windows, Linux) on a computer. Include partitioning the hard drive and configuring system settings.	1	Applying the knowledge of installation to install softwares.	3
Practical 6	Connect and configure a printer or a scanner to a computer. Install the necessary drivers and software, and demonstrate printing or scanning a document.	1	Applying the knowledge to configure a printer.	3

Practical 7	Install a software application (e.g., a web browser or office suite) on a computer and configure its basic settings.	1	Applying the configuration knowledge for software installation.	1,3
Practical 8	Write a simple program in a language of your choice (e.g., C, C++) that takes user input and performs a basic calculation or task.	1	Applying and creating a first program in any language.	3,5
Practical 9	Demonstrate how to use version control software (e.g., Git) to clone a repository, make changes, commit those changes, and push them to a remote repository.	1	Understanding and applying the Git knowledge to clone repositories, etc.	2,3
Practical 10	Use an Integrated Development Environment (IDE) to write, debug, and run a simple program. Explain the benefits of using an IDE.	1	Using different IDEs for creating, solving and running simple programs.	3
Practical 11	Set up a small local area network (LAN) with at least two computers. Ensure that they can communicate with each other by sharing files.	1	Identifying and applying the knowledge on different types of networking.	2,3
Practical 12	Explain the difference between static and dynamic IP addressing. Configure a computer with a static IP address.	1	Explaining and differentiating between static and dynamic IP addressing.	2,4
Practical 13	Use the ping and traceroute (or tracert on Windows) commands to diagnose network connectivity issues between two devices on a network.	1	Using ping and traceroute knowledge to diagnose networks.	3
Practical 14	Install an antivirus program on a computer. Perform a full system scan and demonstrate how to handle detected threats.	1	Understanding and demonstrating how to handle threats.	2,3
Practical 15	Explain safe browsing practices to avoid malware. Show how to configure a web browser with extensions and settings to enhance security.	1	Understanding and explaining safe browser practices.	2,3
Practical 16	Demonstrate how to create strong passwords and manage them using a password manager. Explain the importance of using different passwords for different accounts	1	Explaining and creating strong passwords using different characters and different passwords for different accounts.	2,5

Practical 17	Configure a software firewall on a computer to block all incoming connections except for a specific application (e.g., a web server).	1	Configuring a software firewall to block any connections from malicious users.	3,5
Practical 18	Make a Resume. It must include the features mentioned below— · Tables · Alignment · Bullets · At least 3 Font Styles and 3 Font Sizes One picture	1	Applying and creating a resume with MS Word.	3,5
Practical 19	Create time table in MS Word.	1	Applying and creating a time table in MS Word.	3,5
Practical 20	Create University ID card in MS Word.	1	Applying different styles to create a University ID card.	3,5
Practical 21	Use MS-Word to create Project Reports or Thesis.	1	Applying and creating reports in MS Word.	3,5
Practical 22	Make a presentation of 3 slides in MS PowerPoint. The first slide should have a title andsubtitle, the second slide an image with a caption, and the third slide a bulleted list.	1	Applying and building presentation with MS PowerPoint	3,5
Practical 23	Create time table in MS Excel	1	Creating time table in MS Excel	5
Practical 24	Prepare a stationary order for the month of March in MS Excel.	1	Applying the knowledge of MS Excel to create stationary order details.	3,5
Practical 25	Prepare a student marks distribution table in MS Excel	1	Creating a marks records using MS Excel	5
Practical 26	VLOOKUP and INDEX function in MS Excel	1	Using formulas like VLOOKUP and INDEX in MS Excel.	3,5
Practical 27	Joining two strings in MS Excel	1	Using excel formulas to join two strings.	3,5
Practical 28	Conditional Formatting in MS Excel	1	Applying conditional formatting for different data cells.	3

TEXT BOOKS:

T1: Introduction of Computer Sc. ITL ESL, Pearson Education India.

T2: Computer Fundamentals. Rajaraman, V.

REFERENCE BOOKS:

R1: Computer Fundamentals: Concepts, Systems & Applications Priti Sinha, Pradeep K., Sinha, BPB Publications

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the working of a computer and its various components.	3,8
2	Discuss the use of Software and programming in a computer system.	1,2,3,8
3	Outline the basic concepts of Computer Networks and Internet Protocols.	2,8
4	Develop Proficiency in Identifying different types of computer viruses, worms, and malware to enhance threat awareness.	2,3,8
5	Apply the role of various Software packages for Office Automation	2,3,8

SEMESTER – I									
Course Title	**Soft Skill - I**								
Course code	**24UBPD1102R**	**Total credits: 2** **Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	Bachelor of Business Administration/Bachelor of Hotel Management and Catering Technology/Bachelor of Business Administration (I)/Bachelor of Business Administration (Industry Integrated)/Bachelor of Social Work/Bachelor of Arts in Sociology/Bachelor of Arts in Psychology/Bachelor of Trauma, Emergency and Disaster Management/Bachelor of Science in Biotechnology/Bachelor of Science in Microbiology/Bachelor of Science in Food Nutrition and Dietetics/Bachelor of Computer Application/ Bachelor of Computer Application (P)/Bachelor of Science in Information Technology (P)/ Bachelor of Science in Information Technology/Bachelor of Computer Application on Cloud Technology and Information Security/Bachelor of Computer Application on Artificial Intelligence and Machine Learning/Bachelor of Computer Application on Artificial Intelligence and Machine Learning								
Semester	**Fall/I or Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. To recognize and identify parts of a sentence and their significance in a language. 2. To enhance listening and speaking/skills for self-development. 3. To give insight into English pronunciation and into central concepts in phonetics. 4. Introduction to the various modes of communication will enhance their knowledge of communication.								
CO1	It enables learners to recognize the structure of a sentence and its variations as they learn to understand, speak and write.								
CO2	Introduction to Phonetics and its importance will improve the learners' pronunciation								
CO3	Students will be able to identify, pick and form different kinds of sentences.								
CO4	Knowledge of communication will be enhanced through practical examples.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Unit 1- Basics of Grammar (Flipped classroom)** i. Parts of Speech ii. Articles iii. Auxiliary Verbs iv. Affirmative and Negative Sentences	**4**	Understanding, identifying and analyzing part of speech, articles, auxiliary verbs, affirmative and negative sentences.	1,2,4
II	**Grammar (Flipped classroom)** i. Determiners	**4**	Understand, identify and construct clear sentences,	1,2,3

	ii. Sentence Construction iii. Types of Sentences (Assertive, Imperative, etc.) iv. Degree of Comparison		determiners and degree of comparison.	
III	**Speaking Skills** **i**. Introduction and Greetings ii. Pronunciation, Intonation, Stress iii. Asking and offering information	**6**	Develop the ability to effectively introduce themselves, engage in self-discovery, grasp the basics of phonetics and pronunciation, deliver extempore speeches, and use video recording for self-reflection to enhance their speaking skills.	2,4
IV	**Communication Skills** i. Introduction to Communication ii. Process and Types of Communication, iii. Formal and informal communication iv. Understanding Barriers to Communication	**6**	Understand the fundamentals and importance of communication skills, identify the purpose and types of communication, differentiate between formal and informal communication	5
V	**Presentation Skills** i. Introduction ii. Essential characteristics of a good presentation iii. Use of Visual Aids in Presentation	**8**	Ability to present self-introduction and using visual aids in presentation.	2

TEXT BOOKS:

T1: Chaturvedi, P.D., ChaturvediMukesh, 2011.*Business Communication: Concepts, Cases and Applications*, second edition, Pearson, Noida.

T2: Alex K., Chand, S, 2009. *Soft Skills: Know Yourself and Know the World*, first edition, S.Chand& Company Ltd.: New Delhi.

REFERENCE BOOKS:

R1: Quirk, Randolp. (2010*) A Comprehensive Grammar of the English Language* Randolph Quirk, Sidney Greenbaum, Pearson Education India

R2: Marks, Jonathan. (2017) *IELTS Advantage Speaking and Listening Skills: A step-by-stepguide to a high IELTS speaking and listening score. Book + CD-ROM*, Delta Publishing by Klett

OTHER LEARNING RESOURCES:

- https://youtu.be/bEB8- SWMYhIhttps://youtu.be/- zZau_dttRY

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	It enables learners to recognize the structure of a sentence and its variations as they learn to understand, speak and write.	2 and 6
2	Introduction to Phonetics and its importance will improve the learners' pronunciation	6 and 8
3	Students will be able to identify, pick and form different kinds of sentences.	2 and 6
4	Knowledge of communication will be enhanced through practical examples.	2 and 6

SEMESTER – I									
Course Title	**CO-CURRICULAR ACTIVITIES**								
Course code	**24UBCC1101**	**Total credits: 1** **Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**4**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	It is to develop the social and soft skills and to promote a holistic development of the learners								
CO1	Connect and adapt cultural diversity among communities.								
CO2	Enhance team for working toward a shared vision								
CO3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals								
CO4	Learn to effectively communicate, delegate responsibilities and motivate team members.								
CO5	Develop strong teamwork and collaboration skills by engaging in group activities.								
Course Contents	AdtU encourages activities that are aimed to develop the social and soft skills of students and promote a holistic development of the learners. Keeping in mind the 360 degree learning methodology, the students are engaged in different activities other than their regular classes. Experts are invited to conduct workshops that benefit the students. Technical quizzes are conducted to enhance the knowledge of the students. Technical seminars help them to develop their public speaking abilities.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Connect and adapt cultural diversity among communities.	4,5,6,7,8
2	Enhance team for working toward a shared vision	4,5,6,7,8
3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals	4,5,6,7,8
4	Learn to effectively communicate, delegate responsibilities and motivate team members.	5,6,7,8
5	Develop strong teamwork and collaboration skills by engaging in group activities.	5,6,7,8

SEMESTER – I									
Course Title	**MOOCS I (INTRODUCTION TO GIT AND GITHUB)**								
Course code	**24MOSY1103R**	**Total credits: 1** **Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			0	**0**	**0**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To learn and apply basic Git commands (init, clone, add, commit, status, log) to manage local repositories and track changes. 2. To create and manage branches, perform merges, and resolve conflicts to maintain a smooth development workflow. 3. To create a GitHub account, set up remote repositories, and perform push and pull operations to synchronize local and remote repositories. 4. To practice collaborative workflows on GitHub, including forking repositories, creating pull requests, reviewing code, and managing repository settings. 5. To utilize advanced Git features such as rebasing, tagging, and stashing to enhance your version control practices.								
CO1	Acquire a solid foundation in using Git for version control, including initializing repositories, committing changes, and managing version history								
CO2	Master advanced Git functionalities such as branching, merging, and resolving conflicts to manage and streamline development workflows.								
CO3	Develop the skills to effectively use GitHub for hosting remote repositories, collaborating on projects, and managing code reviews.								
CO4	Learn best practices for collaborating with others using Git and GitHub, including forking, pull requests, and managing repository settings.								
CO5	Create and maintain a professional portfolio on GitHub, showcasing projects and demonstrating the ability to use Git and GitHub effectively.								
Unit	**Contents**	**Contact Hrs.**	**Learning Outcome**						**BL**
I	**Introduction to Version Control:** In this module, introduction to the concept of version control, which will make managing and rolling back the code look super easy. Learn how to differentiate between files and the tools at your disposal to make this happen. Next, you'll be introduced to Git and how you can leverage that platform to improve your coding abilities. Once you've got a grasp on what Git is, you'll install it and start using it to create and clone code repositories. Last up, you'll deep dive into Git in order to get more familiar with the different tools and commands it has to offer.	**9**	Understanding and applying the knowledge of Git and GitHub and its version control.						1,3

II	**Using Git Locally:** In this module, you'll dive into advanced Git interactions by skipping the staging area for small code changes, and understand how Git uses the HEAD alias to represent checked-out snapshots. Next, you'll explore how to move and remove files before finally getting a study guide to help you in your Git explorations! The next step of your learning will include how to undo changes before committing and how to amend commits once they're submitted. Finally, you'll be able to identify errors in commits that were submitted a while back. In the final section of this module, you'll explore the concept of branching and merging. You'll learn what a branch is, how to create one, and how they work in harmony with you and your code. Once you're comfortable with branching, you'll dive into merging, how it works with branched data, and how to deal with merge conflicts.	**6**	Remembering, understanding, and applying git knowledge to use different functions.	1,2,3
III	**Working with Remotes:** In this module, you'll be introduced to GitHub and learn how it works with Git. You'll create new repositories and clone those repositories onto your computer. Next, we'll explain what a remote repository is, how we can work with them, and how we can host them. You'll get familiar with commands like modify, stage, and commit, which will be used for local changes, as well as the fetch command, which can pull any changes from remote repositories. We'll cover secure shell protocol and when to use API keys. Our final lesson will focus on learning about conflicts. This will allow you to explore the concepts of pull-merge-push workflows, pushing remote branches and rebasing your changes.	**5**	Understanding and applying the knowledge of git in GitHub and creating new repositories.	1,3,5

IV	**Collaboration:** In this module, you'll continue to explore the collaboration tools available in Git. You'll learn about the tools that are available to help improve the quality of your code and to better track your code. This includes an overview of pull requests and how the typical workflow of a pull request looks like on GitHub. Next, you'll dive into how you can squash changes in your code. We'll finish up by providing you with a study guide on fork and pull requests. Next up, we'll cover what code reviews are and what the code review workflow looks like. Then, you'll learn about how to use code reviews on GitHub. The final lesson of this module will focus on managing projects. We'll take a rundown of best practices on managing projects and how to manage collaboration within those projects. We'll explore different ways of tracking issues and finish up by discussing the concept of continuous integration with your projects.	**6**	Understanding, analyzing, and applying new tools available.	1,3,4

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Acquire a solid foundation in using Git for version control, including initializing repositories, committing changes, and managing version history	3 and 8
2	Master advanced Git functionalities such as branching, merging, and resolving conflicts to manage and streamline development workflows.	2 and 3
3	Develop the skills to effectively use GitHub for hosting remote repositories, collaborating on projects, and managing code reviews.	3 and 7
4	Learn best practices for collaborating with others using Git and GitHub, including forking, pull requests, and managing repository settings.	6 and 7
5	Create and maintain a professional portfolio on GitHub, showcasing projects and demonstrating the ability to use Git and GitHub effectively.	7 and 8

SEMESTER – I									
Course Title	**MINI PROJECT-I**								
Course code	**24BCAM1105R**	**Total credits: 2 Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	0	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**						**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: ● theoretical studies ● computer simulations ● hardware construction; 2. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 3. Deliver a seminar on the general area of work being	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.						3, 4, 5

	undertaken and specific contributions to that field; 4. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

TEXT BOOKS:
REFERENCE BOOKS:
OTHER LEARNING RESOURCES:
As given by the Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written in oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – I									
Course Title	**BRIDGE I (ALGORITHMS FOR PROBLEM SOLVING)**								
Course code	**24BCAM1106R**	**Total credits:2 Total hours: 45L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To understand the basics of algorithm and computational knowledge. 2. To understand computational thinking. 3. To apply algorithmic problem-solving techniques to real world problems. 4. To write efficient code for solving computational challenges.								
CO1	Understand computational thinking and its four pillars.								
CO2	Understand and apply algorithms for various problems.								
CO3	Understand and analyze the principle of divide and conquer.								
CO4	Understanding graph theory and representation.								
CO5	Understanding linked list and trees in data structure.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to Computational Thinking: About computational thinking, four pillars of computational thinking.	**8**	Understanding and applying computational thinking for different real-world computer-based problems.	1,3
II	Introduction to algorithm: About algorithm, basic algorithm like search and sort, pseudocode	**10**	Understanding and applying different algorithm and analyzing the algorithm	1,3, 4
III	Divide and Conquer: Principles of divide and conquer, Quick Sort, Merge Sort	**8**	Understanding, applying divide and conquer method, and analyzing quick sort and merge sort.	1,3, 4
IV	Graph Algorithms: Basics of graph theory, Graph representation and traversal. Shortest path algorithm.	**10**	Understanding, applying graph structure, and analyzing shortest path algorithm.	1,3, 4
V	Data Structures: Basic data structures, linked list, trees, minimum spanning trees	**9**	Understanding data structure, and analyzing minimum spanning trees.	1,4

TEXTBOOKS:

T1: "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.

T2: “Computational Thinking: A Beginner's Guide to Problem-Solving and Programming” by Karl Beecher.

T3: "C Programming Absolute Beginner's Guide" by Perry and Miller

REFERENCES:

R1: "Code: The Hidden Language of Computer Hardware and Software" by Charles Petzold.

R2: "Algorithms to Live By: The Computer Science of Human Decisions" by Brian Christian and Tom Griffiths.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand computational thinking and its four pillars.	1,2,3,4,5,6,7, and 8
2	Understand and apply algorithms for various problems.	1,2,3,4,5,6,7, and 8
3	Understand and analyze the principle of divide and conquer.	1,2,3,4,5,6,7, and 8
4	Understanding graph theory and representation.	1,2,3,4,5,6,7, and 8
5	Understanding linked list and trees in data structure.	1,2,3,4,5,6,7, and 8

SEMESTER – I									
Course Title	**BRIDGE II (FUNDAMENTALS OF STATISTICS)**								
Course code	**24BCAM1107R**	**Total credits: 2** **Total hours: 45L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To introduce the foundations of probability and statistical methods. 2. To explain the concepts in random variables and several distributions in engineering applications. 3. To learn the concepts of correlation, regression and estimations and their properties and to explain the concept of testing of hypotheses.								
CO1	Understanding the fundamental concepts of probability and statistics.								
CO2	Understanding measures of central tendency and measures of dispersion.								
CO3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.								
CO4	Explain the concept of sampling distributions and estimation.								
CO5	Analyze the concepts of hypothesis testing.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to statistics, population vs sample, collection of data, primary and secondary data, types of variables: dependent, independent, categorical and continuous variables, data visualization,	**10**	Understanding and applying statistical knowledge.	1,3
II	Measures of Central Tendency (Mean, median, mode, Skewness and symmetry) Measures of Dispersion (Range, variance, standard deviation, Interquartile range)	**8**	Understanding and applying different ways of measuring the central tendency.	1,3
III	Introduction to Probability: Probability rules (addition, multiplication, complement) Probability Distributions: Discrete and continuous distributions, Binomial, Poisson, and normal distributions.	**10**	Understanding, applying probability and its method.	1,3

IV	Sampling and Sampling Distributions, Introduction to Random sampling, Confidence Intervals: Basic idea of estimation, Constructing simple confidence intervals.	**8**	Understanding, applying sampling distribution, and analyzing confidence interval.	1,3,4
V	Hypothesis Testing: Introduction to Hypothesis Testing, Formulating hypotheses, Basic understanding of p values, Common Tests.	**9**	Understanding hypothesis testing, and analyzing different hypothesis testing method.	1,4

TEXT BOOKS:

T1: Richard A. Johnson, "Miller & Freund's probability and statistics for engineers", 9th Edition, Pearson, 2017.

T2: S.C. Gupta and V.K. Kapoor, "Fundamentals of Mathematical Statistics", 11th Edition, Sultan Chand & Sons Educational Publications, 2012.

REFERENCE BOOKS:

R1: W. Feller, "An Introduction to Probability Theory and its Applications", 3rd Edition, Wiley, 1968.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understanding the fundamental concepts of probability and statistics.	**1,2,3,4,6,7, and 8**
2	Understanding measures of central tendency and measures of dispersion.	**1,2,3,4,6,7, and 8**
3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.	**1,2,3,4,6,7, and 8**
4	Explain the concept of sampling distributions and estimation.	**1,2,3,4,6,7, and 8**
5	Analyze the concepts of hypothesis testing.	**1,2,3,4,6,7, and 8**

SEMESTER – II									
Course Title	**Data Structure using C**								
Course code	**24BCAM1201R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45L+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Fundamentals of programming logic	**Co-requisite**	C Programming						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To impart the basic concepts of data structures and algorithms. 2. To understand basic concepts about stacks, queues, lists, trees and graphs. 3. To understand concepts about searching and sorting techniques								
CO1	Illustrate the Basic concepts of Data Structures.								
CO2	Apply Data Structure techniques on computing problem.								
CO3	Analyse and develop algorithms to solve real world problems.								
CO4	Implement and developed program for various concepts of data structures including array, stack, queue, graphs and trees.								
CO5	Demonstrate and analyze various sorting algorithms and hashing techniques.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis..	**8**	Analyze the algorithms to determine the time and computation complexity and justify the correctness	1,2, 3
II	**Stacks and Queues**: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis..	**10**	Analyze the problem of stacks, queues and linked list to determine the time and computation complexity	2,3

III	**Linked Lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.	**10**	Analyze the problem of stacks, queues and linked list to determine the time and computation complexity	3
IV	**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis	**9**	Implement basic tree based traversal and search algorithms, learn about applications using tree	1.3
V	**Sorting and Hashing:** Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing. **Graph:** Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.	**8**	Summarize Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity. Graph search and traversal algorithms and determine the time and computation complexity	2,3
Practical Component				
Practical 1	Array implementation of stack using C program	1	Understand the basics of stack	2,3
Practical 2	Array implementation of queue using C program	1	Understand the basics of queue	2,3
Practical 3	Array implementation of list using C program	1	Understand the basics of array implementation	2
Practical 4	Linked list implementation of list using C program [singly linked list]	1	Understand the basics of linked list	2,3
Practical 5	Linked list implementation of stack using C program	1	Understand and implement stack using linked list	2,3

Practical 6	Linked list implementation of queue using C program	1	Understand and implement queue using linked list	2,3
Practical 7	Applications of list polynomial Addition and subtraction using C program	1	Understand and apply polynomial addition and subtraction	4
Practical 8	Infix to postfix using C program	1	Understand and convert infix to postfix using C program.	3
Practical 9	Expression evaluation using C program	1	Evaluate expression using C program.	3,4
Practical 10	Implementation of avl trees using C program	1	Understand AVL tree and implement using array.	3
Practical 11	Implementation of heap using priority Queues using C program	2	Develop heap using priority queue.	3,4
Practical 12	Representation of graph using C program	2	Understand graph and it representation.	3
Practical 13	Graph traversal-breadth first traversal using C program	2	Understand the concepts of traversal breadth first traversal .	3
Practical 14	Graph traversal-depth first Traversal using C program	2	Understand and apply the concepts of traversal depth first traversal .	3
Practical 15	Linear search using C program	2	Understand and apply linear search	2,3, 4
Practical 16	Binary search using C program	2	Understand and apply binary search	2,3, 4
Practical 17	Insertion sort using C program	2	Implement and apply Insertion sort.	3,4
Practical 18	Bubble sort using C program	2	Implement and apply Bubble sort.	3,4
Practical 19	Quick sort using C program	2	Implement and apply Quick sort.	3,4
Practical 20	Merge sort using C program	2	Implement and apply Merge sort.	3,4

TEXT BOOKS:

T1: “Fundamentals of Data Structures”, Illustrated Edition by Ellis Horowitz, SartajSahni, Computer Science Press.

REFERENCE BOOKS:

R1: Algorithms, Data Structures, and Problem Solving with C++”, Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company
R2: “How to Solve it by Computer”, 2nd Impression by R. G. Dromey, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate the Basic concepts of Data Structures.	1,2,3
2	Apply Data Structure techniques on computing problems.	1,2,3
3	Analyse and develop algorithms to solve real world problems.	1,2,3
4	Implement and developed program for various concepts of data structures including array, stack, queue, graphs and trees.	1,2,3
5	Demonstrate and analyze various sorting algorithms and hashing techniques.	1,2,3,8

SEMESTER – II									
Course Title	**Computer Organization & Architecture**								
Course code	**24BCAM1202R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30L	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. Discuss the organization and architecture of computer systems and electronic computers. 2. Analyze the basic components of computer systems besides the computer arithmetic. 3. Discuss input-output organization, memory organization and management, and pipelining.								
CO1	Outline the basics of instructions sets and their impact on processor design								
CO2	Examine the control unit design approaches, memory design technologies and I/O transfers.								
CO3	Explain the concepts of pipelining in Computer Architecture.								
CO4	Interpret and analyze Parallel Processing Principles and Applications.								
CO5	Demonstrate the concepts of Memory Organization Through Mapping Functions and Replacement Algorithms.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU – registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs.	**9**	Infer the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes	1,2, 3

II	Data representation: signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-andadd, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic. Introduction to x86 architecture.	**10**	Explain the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers. Examine the control unit design approaches, memory design technologies and I/O transfers. Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.	3,4
III	CPU control unit design: hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU. Memory system design: semiconductor memory technologies, memory organization Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB	**10**	Explain the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory. Summarize the concepts of memory organization with mapping functions and replacement algorithms	3,4
IV	Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards. Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.	**8**	Summarize the concepts of memory organization with mapping functions and replacement algorithms.	4
V	Memory organization: Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.	**8**	Summarize the concepts of memory organization with mapping functions and replacement algorithms	1,2
Practical				

Practical 1	Study and verify the truth table of logic gates.	**1**	Understanding and identifying different logic gates.	1,2
Practical 2	Design and verify the circuit and operation of half adder and full adder.	**1**	Understanding, designing and analyzing the circuit and operation of half and full adder.	1,4, 5
Practical 3	Design and verify the circuit and operation of half subtractor and full subtractor.	**2**	Understanding, designing and analyzing the circuit and operation of half and full subtractor.	1,4, 5
Practical 4	Design and verify Ripple Carry Adders.	**2**	Understanding, designing and analyzing the circuit and operation of ripple carry adders.	1,4, 5
Practical 5	Design and verify Carry Look ahead Adders.	**2**	Understanding, designing and analyzing the circuit and operation of carry look ahead adders.	1,4, 5
Practical 6	Design and verify Registers and Counters.	**2**	Understanding, designing and analyzing the circuit and operation of registers and counters.	1,4, 5
Practical 7	Design and verify Combinational Multipliers.	**2**	Understanding, designing and analyzing the circuit and operation of combinational multipliers.	1,4, 5
Practical 8	Design of Memory.	**2**	Designing a memory and analyzing it.	4,5
Practical 9	Design of Associative Cache and Direct Mapped cache.	**2**	Designing a associative and direct cache and analyzing it.	4,5
Practical 10	CPU Design	**2**	Designing a CPU and analyzing it.	4,5
Practical 11	Implement Booth's Algorithm	**2**	Understanding and implementing booth's algorithm	1,3
Practical 12	Design and set up a 4:1 Multiplexer.	**2**	Designing and analyzing 4:1 multiplexer.	4,5
Practical 13	Design and verify a 4-bit combinational shifter.	**2**	Designing and analyzing 4-bit combinational shifter.	4,5
Practical 14	Design and verify a BCD adder and subtractor using IC.	**2**	Designing and analyzing BCD adder and subtractor using IC.	4,5
Practical 15	Design a 4-bit adder subtractor.	**2**	Designing and analyzing 4-bit adder subtractor.	4,5
Practical 16	Design and verify Logic Unit using IC.	**2**	Designing and analyzing Logic unit using IC.	4,5

TEXT BOOKS:

T1: “Computer Organization and Design: The Hardware/Software Interface”, 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.

REFERENCE BOOKS:

R1: “Computer Organization and Embedded Systems”, 6th Edition by Carl Hamacher, McGraw Hill Higher Education.
R2: “Computer Architecture and Organization”, 3rd Edition by John P. Hayes, WCB/McGraw-Hill.
R3: “Computer Organization and Architecture: Designing for Performance”, 10th Edition by William Stallings, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOME

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Outline the basics of instructions sets and their impact on processor design	3,5,6,8
2	Examine the control unit design approaches, memory design technologies and I/O transfers.	2,3,5,6,8
3	Explain the concepts of pipelining in Computer Architecture.	2,3,5,6,7,8
4	Interpret and analyze Parallel Processing Principles and Applications.	2,3,5,7,8
5	Demonstrate the concepts of Memory Organization Through Mapping Functions and Replacement Algorithms.	2,3,7,8

SEMESTER – II									
Course Title	**Introduction to Web Technology**								
Course code	**24BCAM1203R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45L+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To understand the Fundamental principle of Web design, hosting and maintenance. 2. To understand tools required for Web design. 3. To understand server and client.								
CO1	Illustrate elements and attributes of a web page.								
CO2	Build web pages using HTML and Cascading Style Sheets								
CO3	Develop XML documents and Schemas								
CO4	Design and implement static and dynamic website								
CO5	Analyse best technologies for solving web client/server problems								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Basics of Internet:** Client/Server Computing: What is C/S Computing, Middleware, Fat client VS Fat Servers, N-tiered Software Architecture. Web's Robot global access to information, HTML, HTTP, IP address, DNS & ports; Accessing a web server, publishing on web server, secure HTTP, Secure Socket layer, WWW Proxies, IIS, Case study of Apache web server.	**10**	Illustrate elements and attributes of a web page.	1,2, 3
II	**Searching and web casting Technique:** Popular web servers, basic feature; bookmarks, cookies, progress indicators, customization of browsers, browsing tricks, next generation web browsing, search engines, architecture of search engines, search tools, web crawler, focused crawler, agent based crawler, case study of any Web Browser.	**12**	Build web pages using HTML and Cascading Style Sheets Develop XML documents and Schemas Design and implement static and dynamic website	3,4

III	**Web page design:** Designing web pages with HTML-use of tags, hyperlinks, URLs, tables, text formatting, graphics & multimedia, imagemap, frames and forms in web pages. Use of Cascading Style Sheet in web pages. Introduction to DHTML.	**8**	Build web pages using HTML and Cascading Style Sheets. Design and implement static and dynamic website. Analyse best technologies for solving web client/server problems	3,4
IV	**Creating interactive and dynamic web pages with JavaScript:** JavaScript overview; constants, variables, operators, expressions & statements; user-defined & built-in functions; client-side form validation; using properties and methods of built-in objects. **Extensible Markup Language (XML):** Introduction- using user-defined tags in web pages; displaying XML contents; XML DTDs; use of XSL.	**8**	Analyse best technologies for solving web client/server problems	4
V	**Web Server:** Web services and web server functionality; web server composition; registration; conceptual architecture of some typical web servers. Server-side scripting: overview of server-side scripts like CGI, ASP, and JSP.Server side scripting using PHP; Web database connectivity-introduction to ODBC; PHP with database connectivity. **Exposure to Advanced Web Technologies:** Distributed Object based models-DCOM, CORBA, EJB; Web services and Related Technologies-ISAPI, SOAP, UDDI, WSDL; Other Advanced Web Technologies- AJAX, ISAPI, .NET.	**7**	Analyse best technologies for solving web client/server problems	4
Practical Component				
Practical 1	Write a HTML program for the demonstration of Unordered Lists.	1	Learn to create Unordered Lists	2,3
Practical 2	Write a HTML program for the demonstration of ordered Lists	1	Learn to create Ordered Lists	2,3
Practical 3	Write a HTML program for the demonstration of Definition Lists	1	Learn to create Definition Lists	2

Practical 4	Write a HTML program for the demonstration of Nested Lists	1	Learn to create Nested Lists Lists	2,3
Practical 5	Write a HTML program for demonstrating Hyperlinks. Navigation from one page to another and Navigation within the page.	1	Understand hyper link and implement hyperlink navigation from one page to another and within the page	2,3
Practical 6	Write a HTML program for time-table using tables.	1	Understand and implement queue using linked list	2,3
Practical 7	Write a HTML program to develop a static Home Page using frames.	1	Understand and apply polynomial addition and subtraction	4
Practical 8	Write a HTML program to develop a static Registration Form	1	Understand and convert infix to postfix using C program.	3
Practical 9	Write a HTML program to develop a static Login Page.	1	Evaluate expression using C program.	3
Practical 10	Write a HTML program to develop a static Web Page for Catalog.	1	Understand AVL tree and implement using array.	3
Practical 11	Write a HTML program to develop a static Web Page for Shopping Cart.	2	Develop heap using priority queue.	3
Practical 12	Write HTML for demonstration of cascading Embedded stylesheets.	2	Understand graph and it representation.	3
Practical 13	Write HTML for demonstration of cascading External stylesheets	2	Understand the concepts of traversal breadth first traversal.	3
Practical 14	Write HTML for demonstration of cascading Inline styles.	2	Understand and apply the concepts of traversal depth first traversal.	3
Practical 15	Write a JavaScript program to validate USER LOGIN page.	2	Develop a user login page	3
Practical 16	Write a program for implementing XML document for CUSTOMER DETAILS.	2	Develop skills in creating well-formed XML documents.	3
Practical 17	Write an internal Document Type Definition to validate XML for CUSTOMER DETAILS	2	Learn how to create an internal DTD to define the structure and constraints of an XML document.	3
Practical 18	Write an external Document Type Definition to validate XML for CUSTOMER DETAILS	2	Develop skills in managing and maintaining DTD files for multiple XML documents..	3
Practical 19	Write a simple servlet that displays a message	2	Develop skills in handling HTTP requests and generating HTTP responses.	3

Practical 20	Write a servlet that reads parameters from employee login page	2	Understand how to read and process parameters from HTTP requests in a servlet.	3

TEXT BOOKS:

T1: C. Xavier, "Web Technology & Design", New Age Publication, 2003
T2: Austin and Pawlan, "Advanced Programming for JAVA2 Platform", Pearson, 2000

REFERENCE BOOKS:

R1: Oliver, Dick; SAMS Teach Yourself Html 4 in 24 Hours; Techmedia.
R2: Ashbacher, Charles; SAMS Teach Yourself XML in 24 Hours; Techmedia.

OTHER LEARNING RESOURCES:

1. https://books.google.co.in/books/about/Data_Structures_and_Algorithms.html?id=1lCHYj5eV-EC&redir_esc=y
2. https://techdevguide.withgoogle.com/paths/data-structures-and-algorithms/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate elements and attributes of a web page.	1,2,3,4,8
2	Build web pages using HTML and Cascading Style Sheets	1,2,3,4,7,8
3	Develop XML documents and Schemas	1,2,3,4,7,8
4	Design and implement static and dynamic website	1,2,3,4,5,7,8
5	Analyse best technologies for solving web client/server problems	1,2,3,4,5,7,8

SEMESTER – II									
Course Title	**Cloud Fundamentals**								
Course code	**24BCAM1205R**	**Total credits: 4 Total hours: 45 L + 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**8**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To provide an overview of an exciting field of Cloud Computing 2. To introduce tools requires building, deploying, running and managing applications on a cloud platform. 3. To develop the cloud application development skills, such as Node.js, REST architecture, JSON, Cloud Foundry and DevOps services 4. To enable students to have skills that will help them to solve complex real-world problems in decision support.								
CO1	Understand fundamental cloud computing concepts and architectures.								
CO2	Analyze and implement RESTful APIs and data services on cloud platforms.								
CO3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.								
CO4	Develop and deploy applications using Node.js and related frameworks.								
CO5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.								
Unit-No.	**Content**		**Contact Hour**			**Learning Outcome**		**BL**	

I	**Introduction to Cloud Computing and IBM Cloud:** Introduction to cloud computing and its characteristics, Benefits of cloud computing, Models of cloud computing IBM Cloud resources	**8**	**C1**	1,2
II	**DevOps and REST API's with data services on IBM Cloud:** What is DevOps?,Architecture of REST,IBM Watson services Types and capabilities of databases, Apis interaction with Cloudant database	**8**	**C2**	2,3
III	**Application with IBM Cloud services:** Understanding business problems and goals, Functional and non-functional requirements, IBM Cloud App ID, Container orchestration (Kubernetes), Pods, Deployment, and Service	**8**	**C3**	2,3

IV	**Developing Cloud Application with Node.js:** Introduction to JavaScript,Node.js modules, Synchronous and Asynchronous programming, Introduction to Express framework ,Route handling	**8**	**C4**	2,3,4
V	**Advanced Cloud Concepts and State-of-the-Art Architecture:** Introduction to Async patterns in ECMAScript, Building a cluster using IBM Cloud, Deployment of an application to Kubernetes ,State-of-the-art architecture in Kubernetes cluster	**8**	**C5**	4

TEXT BOOKS:

1. Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood.
2. Cloud Computing: A Hands-On Approach by ArshdeepBahga and Vijay Madisetti.
3. Cloud Native Architectures: Design High-availability and Cost-effective Applications for the Cloud by Tom Laszewski, Kamal Arora, and Erik Farr.

REFERENCE BOOKS:

1. Cloud Computing Principles and Paradigms by RajkumarBuyya, James Broberg, and Andrzej Goscinski.
2. Cloud Computing: Principles, Systems and Applications by Nick Antonopoulos and Lee Gillam.
3. Cloud Computing: From Beginning to End by Ray J. Rafaels.

Additional Resources:

1. Cloud Essentials: CompTIA Authorized Courseware for Exam CLO-001 by Kalani Kirk Hausman and Susan L. Cook.
2. Cloud Computing: A Practical Approach by Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter.
3. IBM Cloud Essentials: V3 Certification Study Guide by William Rothwell.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand fundamental cloud computing concepts and architectures.	2,4 and 8
2	Analyze and implement RESTful APIs and data services on cloud platforms.	2,3 and 8
3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.	2,3 and 7
4	Develop and deploy applications using Node.js and related frameworks.	2,5 and 6
5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.	5,6 and 7

SEMESTER – II									
Course Title	**Environmental Studies**								
Course code	**24UBES1001R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45L	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To understand the fundamental concepts and principles of environmental science. 2. To analyse the impact of human activities on the environment. 3. To explore ecological processes and the importance of biodiversity. 4. To examine various environmental issues and their potential solutions. 5. To develop critical thinking skills to evaluate environmental policies and practices.								
CO1	Demonstrate the multidisciplinary nature of environmental studies								
CO2	Learning about natural resource, its importance and environmental impacts of Human activities on natural resource								
CO3	Explain environment and ecosystem, concept of biodiversity.								
CO4	Aware students about problems of environmental pollution, its impact and control measures.								
CO5	Illustrate the human population and the environment								
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**						**c**
I	**Multidisciplinary nature of environmental studies:** Definition, scope and importance Need for public awareness.	**6**	Understand basic ecological concepts and environmental processes.						**1,2**
II	**Natural Resources:** **Renewable and non-renewable resources:** Natural resources and associated problems. Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. Energy resources: Growing energy needs, renewable and non-renewable	**6**	Identify and analyze the impact of human activities on the environment. Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them.						**2,3,4**

	energy sources, use of alternate energy sources. Case studies. Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.			
III	**Ecosystems:** Concept of an ecosystem. Structure and function of an ecosystem. Producers, consumers and decomposers. Energy flow in the ecosystem. Ecological succession. Food chains, food webs and ecological pyramids. Introduction, types, characteristic features, structure and function of the Following ecosystem: - Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)	**6**	Identify and analyze the impact of human activities on the environment. Gain knowledge about environment and ecosystem	**3,4**
IV	**Biodiversity and its conservation** Introduction – Definition: genetic, species and ecosystem diversity. Biogeographical classification of India. Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. Biodiversity at global, National and local levels. India as a mega diversity nation. Hot-sports of biodiversity. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. Endangered and endemic species of India. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.	**6**	Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them Gain knowledge about the conservation of biodiversity and its importance. Aware students about problems of environmental pollution, its impact on human and ecosystem and control measures	**4,5**
V	**The Interconnection Between Environmental Pollution and Social Issues: Impacts and Solutions:** Definition Cause, effects and control measures of:-Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards.	**6**	Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them Gain knowledge about the conservation of biodiversity and its importance. Aware students	**4,5**

	Solid wasteManagement: Causes, effects and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies. Disaster management: floods, earthquake, cyclone and landslides. **Social Issues and the Environment:** From Unsustainable to Sustainable development. Urban problems related to energy. Water conservation, rain water harvesting, watershed management. Resettlement and rehabilitation of people; its problems and concerns. Case Studies. Environmental ethics : Issues and possible solutions. Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies. Waste land reclamation. Consumerism and waste products. Environment Protection Act. Air (Prevention and Control of Pollution) Act.Water (Prevention and control of Pollution) Act. Wildlife Protection Act. Forest Conservation Act. Issues involved in enforcement of environmental legislation. Public awareness.		about problems of environmental pollution, its impact on human and ecosystem and control measures	

TEXT BOOKS:

T1: Harucha E. B, Textbook of Environmental Studies, Orient Blackswan Publishing.
T2: Tiwari V. K A Textbook of Environmental Studies, Himalaya Publishing House
T3: Chatwal G. R. &Sharma H. Environmental Studies, Himalaya Publishing House

REFERENCE BOOKS:

R1:Trivedi R.K., Handbook of Environmental Laws, Rules Guidelines, Compliances and
R2: Stadards, Vol I and II, Enviro Media (R)
R3: Trivedi R. K. and P.K. Goel, Introduction to air pollution, Techno-Science Publication (TB)
R4: Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner. BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email:mapin@icenet.net (R)
R5: Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p Clark R.S., Marine Pollution, Clanderson Press Oxford (TB)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate the multidisciplinary nature of environmental studies	**4**
2	Learning about natural resource, its importance and environmental impacts of Human activities on natural resource	**3,4**
3	Explain environment and ecosystem, concept of biodiversity.	**3,4**
4	Aware students about problems of environmental pollution, its impact and control measures.	**4,5,6,7**
5	Illustrate the human population and the environment	**4,8**

SEMESTER – II									
Course Title	**SOFT SKILL - II**								
Course code	**24UBPD1204R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15L+30P	**1**	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To enable students to learn, understand and practice transformation of sentences, correct usage of all tenses and rectify common grammatical errors. 2. To help students to expand their Vocabulary strength along with learning new words and collocations. 3. To train and guide students to improve and their sharpen their listening skill and to become good listeners, 4. To encourage students to master the art of effective reading skills and help apply the SQ3r while reading the text. 5. To guide students to acquire the skill of time managing and apply the same in their profession lives. To equip individuals with skills and information required for creating an excellent LinkedIn Profile.								
CO1	Enable students to understand grammar to write effectively and speak flawlessly, knowing correct usage of tenses and rectifying grammatical errors.								
CO2	Students will acquire the technique of expanding vocabulary and creating new words and use them in different contexts. Encouraging and helping them to sharpen their listening skills and to become good listeners.								
CO3	Encouraging and helping them to sharpen their listening skills and to become good listeners. Students will be able to develop and showcase their effective reading skills while reading any texts.								
CO4	Helping students to acquire the skill of time managing and apply the same in their profession lives along with the skills and information required for creating an excellent LinkedIn Profile.								
CO5	To make them prepare for various public and private sector exams & placement drives. To enhance the analytical skill and problem-solving skill of the students								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar (flipped classroom)** 1. Interchange of Interrogative and Assertive Sentences, Exclamatory and Assertive Sentences 2. Types of Tenses 3. Common Errors 4. Non Verbal reasoning 5. Counting figures 6. Water images and mirror image 7. Clock images 8. Worksheet1 and Worksheet 2	12	Enrich in vocabulary.	**1,2,3**

II	**Vocabulary Development** 1. One word substitution 2. Homonyms and Homophones 3. Words often confused 4. Idioms and phrases 5. V.Average and Age 6. Average of numbers objects things and quantities 7. Problem on age 8. Solving questions on age and average	12	Identify common errors in English speaking and writing	**3,4**
III	**Listening Skills** 1. What is listening? 2. Types of Listening 3. Understanding Listening Barriers 4. Missing number/Letter 5. Letter Series(advanced) 6. Number series(advanced)	12	Improve listening skills.	**3,4**
IV	**Reading Skills** 1. Techniques of Effective Reading 2. Gathering ideas and information from a text 3. The SQ3R Technique 4. Venn diagram **5.** Introduction to venn diagram	10	Learn effective reading techniques.	**4**
V	**Time-Management Skills** 1. Introduction to Time Management 2. Purpose and Importance of Time Management 3. Basic Tips to maintain time 4. Classification 1) Classification of odd one out. Number, meaningful word 2) Classification based on priority	14	Learn time management techniques.	**2,4**

TEXTBOOKS:

T1: Barrett,Grant.2016.*PerfectEnglishGrammar:TheIndispensibleGuidetoExcellentWritingand Speaking* ,Zephyros Press.

T2: McDowell,GayleLaakmann.2008.*CrackingtheCoding Interview*(Indian Edition)

REFERENCE BOOKS:

R1: Zinsser, William.(2006) On Writing Well: The Classic Guide to Writing Nonfiction Harper Perennial
R2: Taylor J. and Wright, J., IELTS Advantage Reading Skills: A step-by-step guide to a high IELTS reading score, Delta Publishing by Klett.
R3: Murphy, Raymond,.(2012) English Grammar in Use Book with Answers: A Self- Study and Practice Book for Intermediate Learners of English ,Cambridge University Press
R4: Quantitative Aptitude and Logical Reasoning by R.S. Agarwal
R5: Quantitative Aptitude and Logical Reasoning by Arihant

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOME

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enable students to understand grammar to write effectively and speak flawlessly, knowing correct usage of tenses and rectifying grammatical errors.	**5,6**
2	Students will acquire the technique of expanding vocabulary and creating new words and use them in different contexts. Encouraging and helping them to sharpen their listening skills and to become good listeners.	**6,8**
3	Encouraging and helping them to sharpen their listening skills and to become good listeners. Students will be able to develop and showcase their effective reading skills while reading any texts.	**6,8**
4	Helping students to acquire the skill of time managing and apply the same in their profession lives along with the skills and information required for creating an excellent LinkedIn Profile.	**7,8**
5	To make them prepare for various public and private sector exams & placement drives. To enhance the analytical skill and problem-solving skill of the students	**8**

SEMESTER – II									
Course Title	**MINI PROJECT-II**								
Course code	**24BCAM1206R**	**Total credits: 2 Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	0	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II Semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: ● theoretical studies ● computer simulations ● hardware construction; 2. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 3. Deliver a seminar on the general area of work being	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	3, 4, 5

	undertaken and specific contributions to that field; 4. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

TEXT BOOKS:
REFERENCE BOOKS:
OTHER LEARNING RESOURCES:
As given by the Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written in oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – I									
Course Title	**EXTRA-CURRICULAR ACTIVITIES**								
Course code	**24UBEC1201**	**Total credits: 1** **Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			0	**0**	**0**	**0**	**0**	**4**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives (Minimum 3)	It is to develop the social and soft skills and to promote a holistic development of the learners								
CO1	Connect and adapt cultural diversity among communities.								
CO2	Enhance team for working toward a shared vision								
CO3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals								
CO4	Learn to effectively communicate, delegate responsibilities and motivate team members.								
CO5	Develop strong teamwork and collaboration skills by engaging in group activities.								
Course Contents	AdtU encourages activities that are aimed to develop the social and soft skills of students and promote a holistic development of the learners. Keeping in mind the 360 degree learning methodology, the students are engaged in different activities other than their regular classes. Experts are invited to conduct workshops that benefit the students. Technical quizzes are conducted to enhance the knowledge of the students. Technical seminars help them to develop their public speaking abilities.								

SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Connect and adapt cultural diversity among communities.	4,5,6,7,8
2	Enhance team for working toward a shared vision	4,5,6,7,8
3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals	4,5,6,7,8
4	Learn to effectively communicate, delegate responsibilities and motivate team members.	5,6,7,8
5	Develop strong teamwork and collaboration skills by engaging in group activities.	5,6,7,8

SEMESTER – II									
Course Title	**BRIDGE I (ALGORITHMS FOR PROBLEM SOLVING)**								
Course code	**23BCAO125R**	**Total credits: 3 Total hours: 45L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To understand the basics of algorithm and computational knowledge. 2. To understand computational thinking. 3. To apply algorithmic problem-solving techniques to real world problems. 4. To write efficient code for solving computational challenges.								
CO1	Understand computational thinking and its four pillars.								
CO2	Understand and apply algorithms for various problems.								
CO3	Understand and analyze the principle of divide and conquer.								
CO4	Understanding graph theory and representation.								
CO5	Understanding linked list and trees in data structure.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to Computational Thinking: About computational thinking, four pillars of computational thinking.	**8**	Understanding and applying computational thinking for different real-world computer-based problems.	1,3
II	Introduction to algorithm: About algorithm, basic algorithm like search and sort, pseudocode	**10**	Understanding and applying different algorithm and analyzing the algorithm	1,3, 4
III	Divide and Conquer: Principles of divide and conquer, Quick Sort, Merge Sort	**8**	Understanding, applying divide and conquer method, and analyzing quick sort and merge sort.	1,3, 4
IV	Graph Algorithms: Basics of graph theory, Graph representation and traversal. Shortest path algorithm.	**10**	Understanding, applying graph structure, and analyzing shortest path algorithm.	1,3, 4
V	Data Structures: Basic data structures, linked list, trees, minimum spanning trees	**9**	Understanding data structure, and analyzing minimum spanning trees.	1,4

TEXTBOOKS:

T1: "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.

T2: “Computational Thinking: A Beginner's Guide to Problem-Solving and Programming” by Karl Beecher.

T3: "C Programming Absolute Beginner's Guide" by Perry and Miller

REFERENCES:

R1: "Code: The Hidden Language of Computer Hardware and Software" by Charles Petzold.

R2: "Algorithms to Live By: The Computer Science of Human Decisions" by Brian Christian and Tom Griffiths.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand computational thinking and its four pillars.	1,2,3,4,5,6,7, and 8
2	Understand and apply algorithms for various problems.	1,2,3,4,5,6,7, and 8
3	Understand and analyze the principle of divide and conquer.	1,2,3,4,5,6,7, and 8
4	Understanding graph theory and representation.	1,2,3,4,5,6,7, and 8
5	Understanding linked list and trees in data structure.	1,2,3,4,5,6,7, and 8

SEMESTER – II									
Course Title	**BRIDGE II (FUNDAMENTALS OF STATISTICS)**								
Course code	**23BCAO126R**	**Total credits: 3** **Total hours: 45L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To introduce the foundations of probability and statistical methods. 2. To explain the concepts in random variables and several distributions in engineering applications. 3. To learn the concepts of correlation, regression and estimations and their properties and to explain the concept of testing of hypotheses.								
CO1	Understanding the fundamental concepts of probability and statistics.								
CO2	Understanding measures of central tendency and measures of dispersion.								
CO3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.								
CO4	Explain the concept of sampling distributions and estimation.								
CO5	Analyze the concepts of hypothesis testing.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to statistics, population vs sample, collection of data, primary and secondary data, types of variables: dependent, independent, categorical and continuous variables, data visualization,	**10**	Understanding and applying statistical knowledge.	1,3
II	Measures of Central Tendency (Mean, median, mode, Skewness and symmetry) Measures of Dispersion (Range, variance, standard deviation, Interquartile range)	**8**	Understanding and applying different ways of measuring the central tendency.	1,3
III	Introduction to Probability: Probability rules (addition, multiplication, complement) Probability Distributions: Discrete and continuous distributions, Binomial, Poisson, and normal distributions.	**10**	Understanding, applying probability and its method.	1,3

IV	Sampling and Sampling Distributions, Introduction to Random sampling, Confidence Intervals: Basic idea of estimation, Constructing simple confidence intervals.	**8**	Understanding, applying sampling distribution, and analyzing confidence interval.	1,3,4
V	Hypothesis Testing: Introduction to Hypothesis Testing, Formulating hypotheses, Basic understanding of p values, Common Tests.	**9**	Understanding hypothesis testing, and analyzing different hypothesis testing method.	1,4

TEXT BOOKS:

T1: Richard A. Johnson, "Miller & Freund's probability and statistics for engineers", 9th Edition, Pearson, 2017.

T2: S.C. Gupta and V.K. Kapoor, "Fundamentals of Mathematical Statistics", 11th Edition, Sultan Chand & Sons Educational Publications, 2012.

REFERENCE BOOKS:

R1: W. Feller, "An Introduction to Probability Theory and its Applications", 3rd Edition, Wiley, 1968.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

		CO PO Mapping
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understanding the fundamental concepts of probability and statistics.	**1,2,3,4,6,7, and 8**
2	Understanding measures of central tendency and measures of dispersion.	**1,2,3,4,6,7, and 8**
3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.	**1,2,3,4,6,7, and 8**
4	Explain the concept of sampling distributions and estimation.	**1,2,3,4,6,7, and 8**
5	Analyze the concepts of hypothesis testing.	**1,2,3,4,6,7, and 8**

SEMESTER – III									
Course Title	**Database Management Systems**								
Course code	**24BCAM2101R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To master core database concepts and SQL for efficient database design and implementation. 2. To explore advanced database features including NoSQL, data warehousing, and security. 3. To apply theoretical knowledge to real-world database management challenges through hands-on projects.								
CO1	Gain an understanding of fundamental database concepts and architectures.								
CO2	Develop proficiency in SQL and relational database design.								
CO3	Master advanced SQL features and database optimization techniques.								
CO4	Learn to design and manage solutions for unstructured data with NoSQL.								
CO5	Apply database knowledge to a real-world data analytics project.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to Database Systems:** Overview of Database Systems: Definition and importance of databases, Evolution of DBMS Types of Database Systems: Relational, NoSQL, NewSQL databases Data models: Entity-Relationship diagram, Relational model, Database schema vs. data Database System Architecture: Roles of database users and administrators, DBMS architecture and data independence	**8**	Understand the definition, importance, and evolution of database systems; differentiate between relational, NoSQL, and NewSQL databases; comprehend data models including Entity-Relationship diagrams and relational models; and explain the roles of database users and administrators, DBMS architecture, and data independence.	2,4,5
II	**Relational Database Management:** SQL and Data Manipulation: Basic to advanced SQL, including queries, updates, and management Database Design: Normalization (1NF to BCNF), Integrity constraints, Indexing, and access methods Transaction Management: ACID properties, concurrency control, deadlock prevention	**10**	Utilize SQL for data manipulation, design normalized databases with integrity constraints and indexing, and manage transactions ensuring ACID properties, concurrency control, and deadlock prevention.	3,4,5
III	**Advanced Database Features:** Advanced SQL and Stored Procedures: Incorporating triggers, views, and procedures Database Security: Implementing data security measures, encryption, authorization, and authentication Performance Tuning and Optimization: Techniques for query optimization and database tuning	**8**	Apply advanced SQL features, implement database security measures, and optimize database performance through various tuning techniques.	3,4,5
IV	**Unstructured Data Management:** Introduction to Unstructured Data: Understanding the significance and management challenges NoSQL Databases: Types and applications: Document stores, key-value stores, wide-column stores, graph databases Data Modelling for NoSQL: Approaches to schema design, integration in NoSQL environments	**6**	Understand the significance and challenges of managing unstructured data, explore various types and applications of NoSQL databases, and learn approaches to schema design and integration in NoSQL environments.	2,3,4

V	**NoSQL Databases and Project-Based Applications:** MongoDB: Specifics of working with MongoDB, data model, query language, and tools for data manipulation and administration Practical Use Cases of NoSQL Databases: real-world applications and case studies that effectively utilize NoSQL databases to handle scalability, performance, and flexibility Project Design and Implementation Project Presentation and Evaluation	**5**	Understanding of MongoDB and NoSQL databases, analyze real-world applications, design and implement database projects, and effectively present and evaluate their work.	1,2,4,5
Practical Component				
Practical 1	**Consider the bank database given, where the primary keys are underlined. Construct the following SQL queries for this relational database.** a. Find the ID of each customer of the bank who has an account but not a loan. b. Find the ID of each customer who lives on the same street and in the same city as customer '12345'. c. Find the name of each branch that has at least one customer who has an account in the bank and who lives in "Harrison".	**4**	Describe, illustrate, explain, apply and develop DBMS Programming using commands.	1,2,3,4
Practical 2	**Create the below tables with the given attributes and enter some records.** a. Write a query to find the total number of marks obtained by each student. b. Write a query that gives the total marks obtained by Mohan. c. Write a query that displays the maximum marks of every subject. d. Write the SQL query which displays the name of those students who's total of all subjects is greater than 205. e. Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2. Pre-Experiment Questions 1. What is the use of sub Queries? 2. Explain different types of Join. Post Experiment Questions 1. How to alter Primary Key? 2. How to update a table by enforcing constraint?	**4**	Design and Create Database Tables, Write SQL Queries for Data Retrieval, Utilize SQL Functions, Implement Conditional Query Logic, Filter Data Based on Multiple Conditions, Understand and utilize subqueries, Alter table structures, Update data with constraints	2,3
Practical 3	**Create the table as given below, fill in the data.** Question: Which normal form in the table? After creating the table, create a second table (s) to convert the same into 2NF!	**4**	Understand Database Normalization, Design and Create Database Tables, Identify Normal Forms,4Transform Tables to Achieve Higher Normal Forms, Implement Referential Integrity, Enhance Data Integrity and Reduce Redundancy	2,3
Practical 4	**Write the SQL queries using Group by and having clause using following table.** Create the above tables with the given attributes and enter some records. I. Write a query to find the total number of marks obtained by each student.	**4**	Understanding Table Creation and Data Insertion, Querying and Summarizing Data, Using Aggregate Functions, Filtering Grouped Data	1,2,3

	II. Write a query that gives the total marks obtained by Mohan. III. Write a query that displays the maximum marks of every subject. IV. Write the SQL query which displays the name of those students who's total of all subjects is greater than 205. V. Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2.			
Practical 5	**Create a table, fill it with customer details and do the following queries:** a. Find the total revenue for each customer b. List customers who have made more than one order c. Find the date of the most recent order for each customer d. Calculate the average order amount for each customer e. Retrieve the orders placed in January 2023 f. Find the top 3 customers with the highest total revenue g. Calculate the total revenue for each month in 2023 h. List orders with a total amount greater than the average total amount for all orders i. Find customers who have made orders on consecutive days j. Calculate the total revenue for each customer, including orders made by customers who haven't placed any orders	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders, Identifying High-Value Customers	3,4
Practical 6	(Exercise on retrieving records from the table) EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id) (a) Find out the employee id, names, salaries of all the employees (b) List out the employees who works under manager 100 (c) Find the names of the employees who have a salary greater than or equal to 4800 (d) List out the employees whose last name is 'AUSTIN' (e) Find the names of the employees who works in departments 60,70 and 80 (f) Display the unique Manager_Id	**4**	Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders, Identifying High-Value Customers	
Practical 7	(Exercise on updating records in table) Create Client_master table with the following fields (ClientNO, Name, Address, City, State, bal_due) (a) Insert five records (b) Find the names of clients whose bal_due>5000. (c) Change the bal_due of Client NO "C123" to Rs. 5100 (d) Change the name of Client_master to Client12. (e) Display the bal_due heading as "BALANCE"	4	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders	3,4
Practical 8	Rollback and Commit commands Create Teacher table with the following	**4**	Data Insertion, Summarizing customer revenue, Analyging	2,3,4

	fields(Name, DeptNo, Date of joining, DeptName, Location, Salary) (a) Insert five records (b) Give Increment of 25% salary for Mathematics Department . (c) Perform Rollback command (d) Give Increment of 15% salary for Commerce Department (e) Perform commit command		ordering data, Filtering and Retrieving Specific Orders, using roll back and commit command	
Practical 9	(Exercise on order by and group by clauses) Create Sales table with the following fields(Sales No, Salesname, Branch, Salesamount, DOB) (a) Insert five records (b) Calculate total sales amount in each branch (c) Calculate average sales amount in each branch . (d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09 (e) Display the name and DOB of salesman in alphabetical order of the month.	**4**	Creation and Insertion of records, use of group by clouse	2,3,4
Practical 10	Create an Emp table with the following fields: (EmpNo, EmpName, Job,Basic, DA, HRA,PF, GrossPay, NetPay) (Calculate DA as 30% of Basic and HRA as 40% of Basic) (a) Insert Five Records and calculate GrossPay and NetPay. (b) Display the employees whose Basic is lowest in each department . (c) If NetPay is less than <Rs. 10,000 add Rs. 1200 as special allowances . (d) Display the employees whose GrossPay lies between 10,000 & 20,000 (e) Display all the employees who earn maximum salary	**4**	Database Design and Table Creation, Data Manipulation and Retrieval, **Querying Data with Conditions,** Aggregate Functions and Grouping, Updating Data,	2,3,4
Practical 11	Create a table, fill it with customer details and do the following queries: a) Find the total revenue for each customer b) List customers who have made more than one order c) Find the date of the most recent order for each customer d) Calculate the average order amount for each customer e) Retrieve the orders placed in January 2023 f) Find the top 3 customers with the highest total revenue g) Calculate the total revenue for each month in 2023 h) List orders with a total amount greater than the average total amount for all orders i) Find customers who have made orders on consecutive days j) Calculate the total revenue for each customer, including orders made by customers who haven't placed any orders.	**4**	Table Creation and Data Insertion, summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders.	
Practical 12	Employee Database An Enterprise wishes to maintain a database	**4**	Database Design and Table Creation, Data Manipulation	2,3,4,5

	to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas Dept (deptno, dname, loc) Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno) a) Update the employee salary by 15%, whose experience is greater than 10 years. b) Delete the employees, who completed 30 years of service. c) Display the manager who is having maximum number of employees working under him? d) Create a view, which contain employee names and their manager		and Retrieval, **Querying Data with Conditions,** Aggregate Functions and Grouping, Updating Data,	
Practical 13	Using Employee Database perform the following queries a) Determine the names of employee, who earn more than their managers. b) Determine the names of employees, who take highest salary in their departments. c) Determine the employees, who are located at the same place. d) Determine the employees, whose total salary is like the minimum Salary of any department. e) Determine the department which does not contain any employees.	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders.	2,3,4,5
Practical 14	11. Write the SQL queries using Group by and having clause using following table. Create the above tables with the given attributes and enter some records. a) Write a query to find the total number of marks obtained by each student. b) Write a query that gives the total marks obtained by Mohan. c) Write a query that displays the maximum marks of every subject. d) Write the SQL query which displays the name of those students who's total of all subjects is greater than 205. e) Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2.	**4**	Design and Create Database Tables, Write SQL Queries for Data Retrieval, Utilize SQL Functions, Implement group by clause Logic, Filter Data Based on Multiple Conditions, Understand and utilize subqueries, display data	2,3,4,5
Practical 15	Create the following tables : Book(accession-no, title, publisher, year, date-of-purchase, status) Member(member-id, name, number-of-books-issued, max-limit) Book-issue(accession-no, member-id, date-of-issue) (a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following : (i) List all those books which are due from the students to be returned. A book is considered to be due if it has been issued 15 days back and yet not returned. (ii) List all those members who cannot be	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders	2,3,4,5

	issued any more books.			
Practical 16	You are given a dataset containing information about various books in JSON format. Each book has attributes such as title, author, published_year, genres, and ISBN. Write a Python script to: 1. Connect to a MongoDB database. 2. Insert the dataset into a collection called books. 3. Query the database to find all books published after the year 2000 and print their titles and authors.	**2**	Connect to a MongoDB database using Python, insert a JSON dataset into a MongoDB collection, and perform queries to retrieve specific data based on given criteria. This exercise enhances their understanding of database operations and data manipulation using MongoDB.	3,5
Practical 17	Using Redis as a key-value store, write a Python script to perform the following tasks: 1. Connect to a Redis server. 2. Store user session data with keys as session_id and values as JSON objects containing user_id, login_time, and status. 3. Retrieve and print the session data for a given session_id.	**2**	Connect to a Redis server using Python, store and manage user session data in a key-value format, and retrieve session data for a given session ID.	3,5
Practical 18	You are tasked with creating a data model in Cassandra to store user activity logs. Each log entry should contain user_id, activity_type, timestamp, and details. Write a Python script to: 1. Connect to a Cassandra cluster. 2. Create a keyspace called user_activity. 3. Create a table called activity_logs with appropriate columns and data types. 4. Insert sample data into the table. 5. Query the table to retrieve all activity logs for a specific user_id.	**2**	Connect to a Cassandra cluster using Python, create a keyspace and table to store user activity logs, insert sample data, and query the table to retrieve logs for a specific user ID.	3,5
Practical 19	You are tasked with managing a MongoDB database for an e-commerce application. The database contains a collection called products with documents that include fields such as product_id, name, category, price, and stock_quantity. Write a Python script to: 1. Connect to the MongoDB database. 2. Insert the following products into the products collection: {"product_id": 1, "name": "Laptop", "category": "Electronics", "price": 1200, "stock_quantity": 30} {"product_id": 2, "name": "Smartphone", "category": "Electronics", "price": 800, "stock_quantity": 50} {"product_id": 3, "name": "Office Chair", "category": "Furniture", "price": 150, "stock_quantity": 20} 3. Write a query to find all products in the "Electronics" category and print their names and prices.	**2**	Establish a connection to a MongoDB database, perform CRUD (Create, Read) operations, and construct queries to filter and retrieve specific data.	2,3,5
Practical 20	You are building a scalable blog platform using MongoDB to manage blog posts. Each blog post document contains post_id, title, author, content, tags, published_date, and views. Write a Python script to: ● Connect to the MongoDB database. ● Insert the following blog posts into the blog_posts collection:	**2**	Handling large-scale data, performing CRUD operations, and writing complex queries to analyze and retrieve high-traffic data efficiently.	3,4,5

	{"post_id": 101, "title": "Introduction to MongoDB", "author": "Alice", "content": "This is a blog post about MongoDB.", "tags": ["MongoDB", "Database"], "published_date": "2024-01-15", "views": 120} {"post_id": 102, "title": "Scaling Applications with NoSQL", "author": "Bob", "content": "This post discusses scaling applications using NoSQL databases.", "tags": ["NoSQL", "Scalability"], "published_date": "2024-02-10", "views": 200} ● Write a query to find all blog posts with more than 150 views and print their titles and authors.			

Text Books:

T1: Database System Concepts, Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, 7th Edition, 2019

T2: SQL in a Nutshell, Kevin Kline, Brand Hunt, and Daniel Kline, 4th Edition, 2020

T3: NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pramod J. Sadalage and Martin Fowler, 1st Edition, 2012

T4: Designing Data-Intensive Applications, The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, Martin Kleppmann, 1st Edition, 2017

Reference Books:

R1: Fundamentals of Database Systems, RamezElmasri&Shamkant B. Navathe, 8th Edition, 2020

R2: Database Management Systems, Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2002

R3: MongoDB: The Definitive Guide, Kyle Banker and Kristina Chodorow

OTHER LEARNING RESOURCES:

- **https://www.jbiet.edu.in/coursefiles/cse/HO/cse2/DBMSI-III.pdf**
- **https://cs.ulb.ac.be/public/_media/teaching/infoh303/dbmsnotes.pdf**
- **SQLZoo-** https://sqlzoo.net/
- DB-Engines- https://db-engines.com/
- MongoDB - https://www.mongodb.com/online

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the basic concepts of database management system.	2, 3, 7, 8
2	Discover the basic fundamentals of creating a database by using data definition, data manipulation and control languages.	2, 3, 7, 8
3	Develop a database application and retrieve the values with the help of queries using SQL.	2, 3, 7, 8
4	Design, analyze and evaluate the project developed for an application.	2, 3, 7, 8
5	Demonstrate the concepts of database security, distributed database systems and object-oriented databases	2, 3, 7, 8

SEMESTER – III									
Course Title	**Object Oriented Programming Paradigm in JAVA**								
Course code	**24BCAM2102R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Basic Programming**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To enable students to understand the basic object-oriented programming concepts and apply them in problem solving. 2. To illustrate among students the inheritance concepts for reusing the program. 3. To enable students to develop GUI-based applications using AWT, Swing, JavaFX and Event handling.								
CO1	Understand object-oriented programming concepts and implement in java.								
CO2	Demonstrate building blocks of OOPs language, inheritance, package and interfaces, and analyze real-world problems in terms of these.								
CO3	Apply the exception handling methods on programming.								
CO4	Develop interactive as well as GUI-based java applications in project-based learning.								
CO5	Outline the concept of package, interface, multithreading and File handling in java.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Java Overview:** Java Introduction, Platform Independence, JVM & JDK, Data types, Operators, If, else statement, Switch condition, while, do-while, for loop, break and continue statement. Array and String: Single Array & Multidimensional Array, Library Classes-String, String Buffer & Wrapper Class, Command line arguments and Various String Operations.	**12**		
II	**Classes, Objects and Methods:** Class and Object, Object reference, Constructor: Constructor Overloading, Method: Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method, Import statement, Static import, Access control, Nested class, Inner class, Anonymous inner class.	**8**		
III	**Inheritance and Interfaces in Java:** Overview of Inheritance, inheritance in constructor, Inheriting Data members and Methods, Multilevel Inheritance – method overriding Handle multilevel constructors super keyword, Stop Inheritance, final keyword. Creation and Implementation of an interface, Interface reference, instance of operator, Interface inheritance, Dynamic method dispatch, Abstract class, Comparison between Abstract Class and interface, inside of System.out.println – statements.	**10**		
IV	**Exception Handling in Java:** Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. **JAVA File Handling:** Overview of Different Stream (Byte Stream, Character stream), Readers and Writers class, File Class, File Input Stream, File Output	**7**		

	Stream, Input Stream Reader and Output Stream Writer class, File reader and writer class, File Writer, Buffered Reader class.			
V	**Applet, AWT, Swing and JavaFX:** Applet: Applet Fundamental, Applet Architecture, Applet Skeleton, Requesting Repainting , Event Handling: various event handling mechanisms, various classes related to event sources and event listeners, AWT: window fundamentals, creating frames, Adding removing various controls, Layout managers, Introduction To Swing and JavaFX.	**8**		
Practical Component				
Practical 1	Program to print all even numbers between 1 and 50 using for loop.	**2**	Demonstrate the understanding of for loops and conditional statements	2
Practical 2	2. Program to calculate and print factorial of a number 'n' using for loop	**2**	Apply the knowledge of for loops and mathematical concepts	3
Practical 3	3. Program to check two arrays are equal or not.	**2**	Analyze and compare two arrays to determine if they are equal by writing a program that iterates through the arrays and checks for element-wise equality	4
Practical 4	Program to add two matrices.	**2**	Apply the understanding of matrix operations	3
Practical 5	Program to display marks, percentage, grade based on attendance.	**2**	Explain how attendance data impacts marks, percentage, and grades	2
Practical 6	Program to check if a number is odd or even.	**2**	Determine and explain whether a number is odd or even by writing a program that uses conditional statements	2
Practical 7	Program to check if a number is prime or not.	**2**	Determine and explain whether a number is prime by writing a program that tests the number's divisibility by integers other than 1 and itself	2
Practical 8	Program to use switch case to print seasons.	**2**	Apply the knowledge of switch case statements by writing a program that prints the appropriate season based on a given input	3
Practical 9	Program to print Volume of box using Constructor overloading.	**2**	Analyze and differentiate between various constructor implementations by writing a program that uses constructor overloading	4
Practical 10	Program to show function overloading.	**2**	Apply the understanding of function overloading	3
Practical 11	Program to show return by object.	**2**	Apply object-oriented programming concepts by writing a program that returns an object from a function to demonstrate encapsulation and data handling	3
Practical 12	Program to show pass by value.	**2**	Understand and explain the concept of pass by value by writing a program that shows how passing arguments by value affects the function's behaviour	2

Practical 13	Program to show pass by reference.	**2**	Understand and explain the concept of pass by reference by writing a program that shows how passing arguments by reference affects the function's behavior	2
Practical 14	Program to find Factorial using recursive function.	**2**	Apply the understanding of recursion by writing a program that calculates the factorial of a number using a recursive function	3
Practical 15	Program to find Month- season using switch case.	**2**	Apply the knowledge of switch case statements by writing a program that determines and prints the season based on the input month	3
Practical 16	Program to print data of employee.	**2**	Demonstrate their understanding of data structures and object-oriented programming	2
Practical 17	Program to print student data.	**2**	Apply object-oriented programming principles	3
Practical 18	Write a Java program to create a new Box class in Java.	**2**	Synthesize the knowledge of object-oriented programming by designing and implementing a new class in Java	6

TEXT BOOKS:

T1: Herbert Schildt and Dale Skrien, "Java Fundamentals – A comprehensive Introduction", McGraw Hill.

T2: Herbert Schildt, "Java the complete reference", McGraw Hill, Osborne.

T3: T. Budd, "Understanding Object- Oriented Programming with Java", Pearson Education.

REFERENCE BOOKS:

R1: P.J.Dietel and H.M.Dietel , "Java How to program", Prentice Hall.

R2: P.Radha Krishna , "Object Oriented programming through Java", CRC Press.

R3: S.Malhotra and S. Choudhary, "Programming in Java", Oxford University Press.

OTHER LEARNING RESOURCES:

Web References:

1. http://java.sun.com
2. http://www.oracle.com/technetwork/java/index.html)
3. http://java.sun.com/javase
4. http://www.oracle.com/technetwork/java/javase/overview/index.html
5. http://download.oracle.com/javase/7/docs/api/index.html

E-Text Books:

1. http://docs.oracle.com/javase/tutorial/
2. iiti.ac.in/people/~tanimad/JavaTheCompleteReference.pdf
3. https://www.codejava.net/books/4-best-free-java-e-books-for-beginners

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Explain the object-oriented programming concepts and implement in java.	1, 2, 5, 8
2	Demonstrate the building blocks of OOPs language, inheritance, package and interfaces, and analyse real-world problems in terms of these.	2, 3, 4, 5, 7, 8
3	Apply the exception handling methods on programming	7
4	Develop interactive as well as GUI-based java applications in project-based learning.	2, 3, 4, 5, 7, 8
5	Outline the concept of package, interface, multi-threading and File handling in java.	1, 2, 3

SEMESTER – III									
Course Title	**WEB AND MOBILE PROGRAMMING TECHNOLOGIES**								
Course code	**24BCAM2104R**	**Total credits: 3** **Total hours: 45L+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**8**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To create responsive web pages using HTML5 and CSS3. 2. To understand and apply JavaScript in web development. 3. .								
CO1	Understand the fundamentals of web technologies, including HTML, CSS, and JavaScript.								
CO2	Develop dynamic web applications using frontend frameworks like React or Angular.								
CO3	Implement server-side development using Node.js, Express, and database integration.								
CO4	Create responsive and mobile-friendly applications using Bootstrap and media queries.								
CO5	Design and develop mobile applications using platforms like Android Studio or Flutter.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Overview of HTML, CSS:** Overview HTML5, elements and attributes; CSS3 styling, selectors, box model; Responsive Design Principles.	**6**	Gain foundational knowledge of web and mobile development, build responsive websites using HTML5	2,3,5
II	**Introduction to JavaScript:** Basic syntax, data types, variables, operators, control flow, functions, array, object, DOM manipulation; ES6 features, DOM manipulation with JavaScript; Event handling, asynchronous JavaScript.	**10**	Understanding JavaScript, applying JavaScript knowledge to solve different problems.	2,3
III	**Introduction to Frontend Development:** Introduction to react, react components and JSX, state and lifecycle methods, handling events in React. React Hooks, React Router for single-page applications, State management with Context API, Integrating APIs with React.	**10**	Understand the concept of frontend development and JSX. Understanding React.js and applying its knowledge to create webapp with react.	2,3,5
IV	**Introduction to Backend Development:** Introduction to PHP, syntax, data types, variables, operators, control flow, functions, array, object; Form handling and user input, database connectivity. MySQL, connecting PHP to MySQL database, CRUD operations.	**10**		3
V	**PHP Frameworks:** Laravel, setting up Laravel, routing, controllers, views and blade templating; Models, eloquent ORM, authentication, middleware, RESTful API with Laravel. Integrating React frontend with Laravel.	**12**		3,6
Practical Component				
Practical 1	Create a basic HTML page with a title, heading, and paragraph. Explain the significance of each HTML tag used.	**2**	Demonstrate the ability to create a basic HTML document structure with a title, heading, and paragraph using appropriate HTML tags.	3
Practical 2	Create a responsive web page using HTML5 and CSS3. Include a header, main content area, and footer. Design a web page layout using Flexbox. Create a	**2**	Develop and implement a responsive web page layout using HTML5, CSS3 (including Flexbox and media queries), to ensure optimal	5,6

	navigation bar with horizontally aligned items that adjust on smaller screens. Implement media queries to adjust the layout for different screen sizes. Test it on different devices.		viewing experience across various screen sizes.	

Textbooks:

T1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins.
T2. "Eloquent JavaScript: A Modern Introduction to Programming" by MarijnHaverbeke.
T3. "React Up & Running: Building Web Applications" by StoyanStefanov.
T4. "Angular Up and Running: Learning Angular, Step by Step" by ShyamSeshadri.
T5. "FullstackVue: The Complete Guide to Vue.js" by Hassan Djirdeh, Nate Murray, and Ari Lerner.

Reference:

R1. "Learning Node: Moving to the Server-Side" by Shelley Powers.
R2. "Express in Action: Writing, building, and testing Node.js applications" by Evan Hahn.
R3. "Beginning Flutter: A Hands On Guide to App Development" by Marco L. Napoli.
R4. "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws" by DafyddStuttard and Marcus Pinto.
R5. "The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations" by Gene Kim, Jez Humble, Patrick Debois, and John Willis.

Other Materials:

- "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley.
- FreeCodeCamp - Creating a RESTful API with Node.js
- https://www.reactnative.express/
- https://flutterbyexample.com/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the fundamentals of web technologies, including HTML, CSS, and JavaScript.	1,2,4,8
2	Develop dynamic web applications using frontend frameworks like React or Angular.	2,3,4,6
3	Implement server-side development using Node.js, Express, and database integration.	2,3,4,7
4	Create responsive and mobile-friendly applications using Bootstrap and media queries.	3,4,6
5	Design and develop mobile applications using platforms like Android Studio or Flutter.	3,4,6,8

COURSE TITLE	Data Visualization								
COURSE CODE	24BCAM2105R	TOTAL CREDITS 3	L	T	P	S	R	O/F	C
			2	0	2	0	0	0	5
PRE-REQUISITE	Basic Programming knowledge	CO-REQUISITE	NIL						
ANTI-REQUISITE	NIL								
PROGRAMME	BCA								
SEMESTER									

Course Objectives:

1. Understand the principles and importance of data visualization in the context of computer applications.
2. Develop proficiency in creating interactive dashboards using IBM® Cognos Analytics.
3. Master the fundamentals of data visualization using Python's Matplotlib library.
4. Explore advanced visualization techniques for data analysis and interpretation.
5. Apply data visualization techniques to real-world datasets and scenarios.
6. Develop critical thinking and analytical skills through hands-on projects and assignments.

Course Outcomes:

On successful completion of the course, the students will be able to:

CO1: Ability to create interactive dashboards and visualizations using Python Programming & IBM® Cognos Analytics.

CO2: Proficiency in scripting and creating various types of plots using Matplotlib.

CO3: Understanding of advanced visualization techniques for data analysis and interpretation.

CO4: Application of data visualization techniques to real-world datasets and scenarios.

CO5: Utilize interactive data visualization tools like Cognos to create dynamic visualizations.

Course Description:

This course aims to introduce students to the fundamentals of data visualization and analysis using Python. Through a comprehensive curriculum, students will gain proficiency in Python programming, understand the mathematical foundations of data science, explore data visualization techniques, and learn to analyze and interpret data effectively.

Text Books:

1. Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures" by Claus O. Wilke
2. Data Visualization with Python" by Mario Dobler

Reference Books:

1. Matplotlib 3.0 Cookbook" by Srinivasa Rao Poladi
2. Data Visualization in Python" by David Landup

Additional Resources:

- IBM Cognos Analytics
- Matplotlib Documentation

UNIT -NO	CONTENT	CONTACT HRS	LEARNING OUTCOME	KNOWLEDGE LEVELS	REFERENCE / LEARNING MATERIALS
I	**Introduction to Data Visualization :** Basics of data visualization Introduction to Python programming language. Data manipulation with NumPy and Pandas Introduction to Matplotlib for basic plotting.	**8**	**C1,C2**	**Remembering Understanding**	**T1/R1**
II	**Exploratory Data Analysis & Advance Data Visualization:** Exploratory data analysis (EDA), Advance Data visualization with Matplotlib&Seaborn , Customizing plots and visualizations, Interactive visualizations with Plotly.	**10**	**C3,C5**	**Analysing, Understanding**	**T1/R1**

III	**Introduction To Data Analytics Using IBM Cognos:** Introduction to IBM Cognos Analytics, IBM Cognos Analytics in Finance Sectors - Reporting Exploring IBM Cognos Environment and Authoring Templates, Design and Execution of Reports, Object Properties Manipulation, Creating Reports from Dimensionally-Modeled Data Sources Exploration of Personal Data Sources and Data Modules, Report Generation from Personal Data Sources, Understanding and Working with List Reports, Grouping and Formatting in List Reports. Enhancing List Reports with Headers and Footers, Advanced List Column Formatting, Aggregating Data: Options and Techniques, Creating Multi-Fact Queries, Filtering Strategies for Report Focus, Detail and Summary Filters, Formatting and Sorting. Crosstab Reports, Chart Creation with Peer and Nested Columns, Utilizing Different Chart Types.	**8**	**C2**	**Understanding, Analysing**	**T2/R2**

IV	**Advanced Report and Visualization:** Introduction to Visualization, Dashboard Design and Key Data Presentation, Prompt Types and Utilizing Parameters, Navigation Techniques within Reports, Calculations and Run-time Information, Expressions using Functions. Introduction to Active Report Charts and Visualizations, Traditional Charts and Static Decks, Creating Data Decks with Traditional Charts, Optimizing Report Size with Decks, Rapidly Adaptive Visualization Engine (RAVE), Converting and Rendering RAVE Visualizations, Creating Active Reports with RAVE Visualization	**8**	**C4**	**Understanding, Applying**	**T1/R2**
V	**Application of Data Visualization in Real World:** Case studies demonstrating data visualization . Visualizing datasets & Creating dashboards for analysis Real-world applications and projects.	**6**	**C6**	**Applying**	**T2/R2**

SEMESTER – III									
Course Title	**MOOCS III (Introduction to Hardware and Operating Systems)**								
Course code	**23MOSY212R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:	**0**	**0**	**0**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To gain foundational knowledge of computers, including their definition, four core functions, essential components, and benefits of using them. 2. To explore the distinction between computer hardware and peripherals. Learn about various types of hardware components and peripherals, including printers, scanners, input/output devices, external storage devices, and audio/video devices. 3. To explore how internal and external components of a computer connect and interact to receive input, process information, and deliver output. 4. To delve into the essential internal components of a computer system, including those found in both home and business environments. 5. To gain knowledge of workstation procurement, basic Windows operating system configuration, and management techniques.								
CO1	Define computers, explain their core functionalities, and identify key components. Understand advantages of using computers and data representation concepts.								
CO2	Differentiate hardware and peripherals. Gain knowledge of various types and their roles in computer operations.								
CO3	Explain communication between internal components and peripherals using ports, interfaces, and connectors.								
CO4	Identify and explain the functions of key internal components like CPU, memory, storage, and motherboard.								
CO5	Manage basic workstation setup (focusing on Windows), organize files effectively, and learn troubleshooting methods.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Computing Fundamentals:** Welcome to your first module! After completing this module, you'll be able to define what a computer is. You'll learn about the four functions of computing, the parts of a computer associated with each of those functions, and the benefits of computing. Explore how to classify computing devices based on size and usage and describe the evolution of operating systems. Discover how computers communicate using notational (numeric) systems for processing and storing data, displaying onscreen text, hardware addresses, website colors, and web addresses. You'll gain the practical knowledge needed to evaluate computing performance and storage. You'll know the four ways to log into Microsoft Windows, be able to describe how to use the Start menu, Tiles, taskbar, and settings capabilities, and will be able to explain how to manage Windows on the desktop. You'll round out your week with a lab where you'll locate programs and navigate Windows to find important information about the computer and the Windows operating system	**3**	Gain a foundational understanding of computers and their role in the modern world.	2
II	**Computing Devices and Peripherals:** Computer hardware is the set of physical parts that make up the interior of your	**3**	Differentiate between computer hardware and peripherals, and identify their various types and	2,3

	computer. Peripherals are pieces of hardware that are added to the outside of a computer for it to work more efficiently. Both hardware and peripherals are used to take input, store data, display output, and execute commands. This week, you will learn about several types of hardware components and peripherals. You will learn about printers and scanners, input and output devices, external storage devices, and audio and video devices.		functionalities within a computer system.	
III	**Interfaces and Connectors:** A computer's multiple components, both internal and external, are constantly interacting with each other to receive input, perform requested computing functions, and provide output. As technology has evolved and connection methods have standardized, a vast array of external devices can now easily and successfully connect with computers and reliably perform their tasks. This week, you will learn how hardware and peripheral devices connect to form a functioning computer. You will learn about ports, interfaces, and connectors used by graphics, audio, networks, printers, and other peripheral devices. You will also learn about the types of installations these devices use to become operational.	**3**	Explain how internal components and external peripherals connect and communicate with each other to facilitate data flow within a computer system.	2,3
IV	**Internal Computer Components:** Computer hardware, the physical part of a computer, is infrequently changed, while software and data are modified frequently. When you think of the term computer hardware, you probably think of what's inside your personal computer at home or your computer at work. However, computer hardware does not specifically refer to personal computers. Instead, it is all types of computer systems. Computer hardware is located in embedded systems in automobiles, microwave ovens, CD players, DVD players, and many more devices. In this lesson, you'll learn about common internal components of home and business computer systems.	**3**	Identify and explain the functions of key internal computer components like the CPU, memory, storage devices, and motherboard.	2,3
V	**Windows Workstation Setup, Evaluation, and Troubleshooting:** Workstation setup, operating system configuration, and workstation troubleshooting are all in a day's work for IT support. Whether the task is	**5**	Gain the practical skills to perform basic workstation setup (focusing on Windows), manage file organization effectively, and apply basic troubleshooting methods to resolve common issues.	3

	selecting a new computer for a colleague, reconfiguring an operating system for a coworker's accessibility needs, or troubleshooting a file or directory visibility issue, IT support personnel need procedures that help them complete these tasks quickly, confidently, and correctly. This week, you'll gain knowledge of workstation procurement and basic Windows operating system configuration and management. You'll discover techniques that help you optimize file and folder organization and perhaps even help users locate hidden files. Learn timesaving keyboard commands to capture screens using Windows 10 workstations, macOS workstations, and Chromebooks. You will complete your learning with an overview of business continuity principles used for fault tolerance and disaster recovery.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Define computers, explain their core functionalities, and identify key components. Understand advantages of using computers and data representation concepts.	
2	Differentiate hardware and peripherals. Gain knowledge of various types and their roles in computer operations.	
3	Explain communication between internal components and peripherals using ports, interfaces, and connectors.	
4	Identify and explain the functions of key internal components like CPU, memory, storage, and motherboard.	
5	Manage basic workstation setup (focusing on Windows), organize files effectively, and learn troubleshooting methods.	

SEMESTER – III									
Course Title	**Fieldbase Learning**								
Course code	**24UUFL2103R**	**Total credits: 1** **Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**8**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings. 4. To foster collaboration and communication skills through group projects and professional interactions. 5. To cultivate an appreciation for the complexities and challenges of professional practice in the field.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyze and interpret field data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								
Course Contents	The organization will provide the students two field visit for this program. The students will be taken to a specific site for learning purposes. The students will visit a relevant site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners regarding the subject or purpose of the visit. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Apply theoretical concepts to real-world situations.	2, 4, and 8
2	Conduct field-based research and gather data effectively.	2, 3, and 8
3	Analyze and interpret field data to draw meaningful conclusions.	2, 3, and 7
4	Communicate findings clearly and effectively, both orally and in writing.	2, 5, and 6
5	Demonstrate professional behaviour and teamwork skills in field settings.	5, 6, and 7

SEMESTER – III									
Course Title	**BASIC LIFE SAVING SKILLS(BLSS)**								
Course code	**24UULS2102R**	**Total credits: 1** **Total hours: 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives (Minimum 3)	The aim of the course is to provide the learners with basic knowledge and practical skills needed in an emergency fire situation, and to provide appropriate basic management and treatment for injuries								
CO1	Understand the specific skills of Basic life saving skill and knowledge that individuals should acquire and demonstrate after completing a BLS training course.								
CO2	Explain the soft skill attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally.								
CO3	Implement the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies.								
CO4	Understand the principles and purpose of the Triage system in healthcare settings.Describe the different levels of triage and their significance in prioritizing patient care.								
CO5	Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage.Recognize the importance of accurate and timely patient assessment during triage.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Basic Life Support (BLS)** Introduction of BLS Chain of survival ABCs Assessment CPR and Ventilation Technique AED Choking for adult and children	**5**	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives	1,2
II	**Soft skills** Introduction Communications Skills Situational Skills Team Work Other Soft Skills	**4**	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life	1,2
III	**Trauma emergencies** Introduction Priorities of Initial approach in pre-hospital care Scene safety Primary assessment Bleeding control	**10**	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders,	1,2,3

	Helmet removal Care of amputated body part Extrication of victims and safe transfer Cervical spine stabilization Cervical collar application Splinting of broken Limbs		and anyone involved in emergency care.	
IV	**Triage system** Introduction Flow chart approach of Triage Triage of Multiple Casualties in Pre-Hospital setting Triage of Single casualt	**5**	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate, delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage	1,2,3,4
V	**Medical emergencies** Introduction Victim centred approach in medical emergency Management of :- a)seizures b)heart attack c)asthma diabetic emergencies emergency childbirth stroke recovery position	**6**	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction (heart attack), stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.	1,2,3,4

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the specific skills of Basic life-saving skill and knowledge that individuals should acquire and demonstrate after completing a BLS training course.	5,6
2	Explain the soft skill attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally.	5,6
3	Implement the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies.	5,6
4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage and their significance in prioritizing patient care.	5,6
5	Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.	5,6

SEMESTER – III									
Course Title	**ADVANCED ENGLISH LANGUAGE FOR EXCELLENCE**								
Course code		**Total credits: 2** **Total hours: 60P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To enable students to learn, understand and practice different ways of presentation and use of correct grammar. 2. To augment the writing skills in different areas including CV and cover letter writing. 3. To boost productivity and performance at work, which assists in the achievement of professional goals. 4. To evaluate the required attributes in a candidate.								
CO1	Develop better listening skills and enhance writing abilities.								
CO2	Identify both positive and negative nonverbal gestures and signs.								
CO3	Improve adept presentation skills by practicing delivery, refining communication techniques, and ensuring clarity of expression.								
CO4	Enhance writing abilities and develop the ability to recognize awkward expressions, correct grammatical errors, and enhance the overall coherence of your writing								
CO5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Grammar :** I. Use of Prepositions II. Tag questions III. Simple, complex, compound sentences	**10**	Describe, illustrate, and explain Prepositions, types of sentences	1,2
II	**Grammar:** I. Active and Passive Voice II. Direct and Indirect Speech	**15**	Describe, illustrate, and explain Voice, Types of voice, Speech	1,2
III	**Writing Skills :** The Basics of Writing; avoid ambiguity and vagueness I. Paragraph Writing II. Letter Writing III. Resume, CV and Cover Letter	**15**	Describe, illustrate, and explain about Paragraph writing, letter writing, Creation of resume, cover letter.	1,2,3
IV	**Self- Management Skills** SWOT Analysis Self- Regulation Personal Hygiene	**10**	Describe, illustrate, and explain and apply Self management skills	1,2,3,4
V	**Non- Verbal Communication- Sciences of Body Language** i. What is Non- Verbal Communication & Body Language, ii. Elements of Communication, iii. Types of Body Language, iv. Importance and Impact of Body	**10**	Describe, illustrate, and explain and apply Non verbal communication.	1,2,3,4

	Language, v. Types of Communication through Body Language, vi. Body Language Do's and Don'ts, Doubt Clearing Session.			

TEXT BOOKS:

1. What Employers Want : The Work skills Handbook- Karen Holmes, 2011

2. English Grammar in Use, Raymond Murphy 4th edition, CUP

REFERENCES:

1. Professional Communication, 2015, by Dr. Prachi Dr. S. K. Singh

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Develop better listening skills and enhance writing abilities.	5,6,8
2	Identify both positive and negative nonverbal gestures and signs.	5,6,8
3	Improve adept presentation skills by practicing delivery, refining communication techniques, and ensuring clarity of expression.	3,5,6,7,8
4	Enhance writing abilities and develop the ability to recognize awkward expressions, correct grammatical errors, and enhance the overall coherence of your writing	5,6,8
5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills.	2,3,5,6,7,8

SEMESTER – III									
Course Title	**Personal Financial Planning**								
Course code	**24UUFL2103R**	**Total credits: 1** **Total hours: 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To create awareness among students about the need for possessing financial literacy education. 2. To Identification of money as a working asset. 3. To impart the ability to make better financial decisions								
CO1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.								
CO2	Design a diversified investment portfolio that addresses several different investment objectives.								
CO3	Differentiate between open- and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.								
CO4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** i. Meaning, need and importance of Financial Literacy; ii. Different components of Financial Literacy; iii. Prerequisites of financial literacy; iv. Savings – Meaning and Difference between savings and investment; v. Types of Financial Institutions and the services provided - Banking and Non-Banking; vi. Different investment avenues.	**7**	Describe, illustrate, and explain concepts of money, Importance of Financial Literacy heir applications to financial planning.	1,2
II	**Financial Planning:** i. Meaning, need and importance for financial planning, ii. Economic needs, balancing between economic need and resources; iii. Three pillars of investments-risk, return, liquidity; iv. Budgeting and its importance in financial planning; v. Steps involved in Financial Planning Process; vi. Preparation of personal budgets, budget surplus and budget deficit, avenues for savings from surplus, sources for meeting deficit. vii. Informal Society funds and crowd funding	**6**	Describe, illustrate, and explain financial planning process, the life cycle of financial plans, and methods of goal achievement.	1,2
III	**Banks & Post Office - As financial service provider:** i. Meaning and evolution of money, ii. Banks – meaning, types & functions; types of accounts; Formalities to open various accounts.	**6**	Describe, illustrate, and explain Formulation of budget, record keeping system, and tax planning strategy based on current financial goals.	1,2,3

	iii. Different types of Post Office saving schemes: Recurring deposit, savings, term deposit; NSC; KisanVikasPatra; Monthly Income scheme (MIS) Account, iv. ublic Provident Funds (PPF), Senior citizen savings scheme (SCSS), SukanyaSamriddhi Accounts, v. Indian Postal Order; International Money transfer service; Forex Services; vi. Money remittance services; Jansuraksha Scheme.			
IV	**Insurance - As financial service provider:** i. Different types of Risks and their Management, Diversification of risk; ii. Meaning,need and importance of Insurance; Types of Insurance – Life Insurance, Health Insurance, General Insurance, Term Insurance, iii. Pension and retirement policies; iv. Post office life insurance schemes, Postal life insurance and rural postal life insurance.	**6**	Describe, illustrate, and explain and Creation of financial plan that covers Risk management, importance of Insurance.	1,2,3,4
V	**Transformations in Digital Money market:** i. Various functions & innovative services of Banks; Mobile Banking, NEFT, IMPS, RTGS, ii. Money transfer, Different types of cards- Debit & Credit, E-Banking,Unified payment interface(UPI), iii. Credit Scoring - CIBIL, Digital Banking, crypto currency and related transactions, **iv.** Fintech, Block chain; Understanding Digital Payments.		Explain the Digital money market , credit score, CIBIL score.	

Textbooks:

T1. Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems and the Million Dollar Financial Advisor: Powerful Lessons and Proven Strategies from Top Producers by David J. Mullen Jr.
T2. Personal Finance and Planning by Dr. Rajni
T3. Peaceful Personal Finance: A short read on the basics of personal finance and planning Kindle Edition by Hema Singh.
T4. Be your own financial advisor: Financial Planning, Investment Options, Risk Management, Tax Management, Succession Planning Kindle Edition y Sushil Bali.
T5. The dumb things smart people do with their money: Thirteen ways to right your financial wrongs Kindle Edition y Jil Schlesinger.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	1,2,4,5,6,7,8
2	Design a diversified investment portfolio that addresses several different investment objectives.	2,3,4,5,6,7
3	Differentiate between open- and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.	4,5,6,8
4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.	1,2,3,4,5,6,7,8
5	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	1,2,3,5,6,7,8

SEMESTER – IV									
Course Title	**Design and Analysis of Algorithms**								
Course code	**24BCAO2201R**	**Total credits: 3** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Data Structures**	**Co-requisite**	**C /C++ Programming**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives (Minimum 3)	1. Analyze the asymptotic performance of algorithms. 2. Demonstrate a familiarity with major algorithms and data structures. 3. Apply important algorithmic design paradigms and methods of analysis.								
CO1	For a given algorithm, analyze worst-case running time based on asymptotic analysis and justify the correctness of algorithm.								
CO2	Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.								
CO3	Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.								
CO4	Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For given problems of dynamic-programming, develop the dynamic programming algorithms, and analyze it to determine its computational complexity.								
CO5	For a given model engineering problem, model it using graph and write the corresponding algorithm to solve the problems.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.	**8**	Define and Identify Characteristics of Algorithms, Understand and Perform Asymptotic Analysis, Evaluate Algorithm Performance, Solve Recurrence Relations, Use Master's Theorem,	**1,2,3**
II	Fundamental Algorithmic Strategies: Brute-Force, Greedy, Dynamic Programming, Branch and Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics –characteristics and their application domains.	**12**	Understand Brute-Force Techniques, Apply Greedy Algorithms, Implement Dynamic Programming Solutions, Use Branch and Bound Techniques, Solve Problems Using Backtracking, Practical Application of Strategies,	**3,4**
III	Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth 6First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	**8**	Solve Problems Using Backtracking, Understand and Implement DFS and BFS, Apply Shortest Path Algorithms, Calculate Transitive Closure, Find Minimum Spanning Trees, Perform Topological Sorting, Understand and Implement Network Flow Algorithms	**3,4**
IV	Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and	**6**	Understand and Explain Cook's Theorem, Identify Standard NP-complete Problems, Solve and Analyze NP-complete Problems, Apply Reduction	**4**

	Reduction techniques		Techniques,	
V	Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE	**5**	Understand Approximation Algorithms, Apply Approximation Algorithms, Understand Randomized Algorithms, Understand P SPACE	**4**

TEXT BOOKS:

T1: Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms, 4TH Edition, MIT Press/McGraw-Hill.

T2: Fundamentals of Algorithms – E. Horowitz et al.

REFERENCE BOOKS:

R1: Jon Kleinberg and ÉvaTardos, Algorithm Design, 1ST Edition, , Pearson.

R2: Michael T Goodrich and Roberto Tamassia, Wiley Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition,.

R3: Udi Manber, Algorithms -- A Creative Approach, 3RD Edition, Addison-Wesley, Reading, MA.

OTHER LEARNING RESOURCES:

1. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/
2. **https://www.vssut.ac.in/lecture_notes/lecture1428551222.pdf**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	For a given algorithm, analyze worst-case running time based on asymptotic analysis and justify the correctness of algorithm.	1,8
2	Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.	1,8
3	Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.	1, 2, 3, 8
4	Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For given problems of dynamic-programming, develop the dynamic programming algorithms, and analyze it to determine its computational complexity	1, 2, 3
5	For a given model engineering problem, model it using graph and write the corresponding algorithm to solve the problems.	1, 2, 3

SEMESTER – IV									
Course Title	**Basics of Python Programming**								
Course code	**24BCAO2202R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Data Structures**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives (Minimum 3)	1. Understand the programming basics (operations, control structures, data types, etc.) 2. Understand and begin to implement compound data like python lists, tuples etc 3. Read/write to files and learn to use basic libraries in python for data analysis								
CO1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.								
CO2	Express proficiency in the handling of strings and functions.								
CO3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets								
CO4	Identify the commonly used operations involving file systems and regular expressions.								
CO5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Python:** Unique features of Python, Python-2 and Python-3 differences, Install Python and Environment Setup, First Python Program, Python Identifiers, Keywords and Indentation, inserting comments in python, Command line arguments, Getting User Input, Data types, Operators, Precedence and associativity	**6**	Understand Python's Features, Compare Python Versions, Install Python, Manage Virtual Environments, Write a Basic Python Program, Understand Python Syntax, Use Comments Effectively, Use Command Line Arguments, Get User Input, Understand Python Data Types	**1,2,3**
II	**Conditional execution and Iteration:** Boolean expressions, logical operators, conditional and alternative execution, chained and nested conditionals, the while statement, infinite loops and *break*, finishing iterations with *continue*, *for* loops	**6**	Understand Boolean Expressions, Use Logical Operators, Implement Conditional Execution, Use Alternative Execution, Use Chained Conditionals, Implement Nested Conditionals, Understand and Use While Loops, Understand and Use For Loops	**3,4**
III	**Functions:** python built-in functions, defining and calling a function, parameters and arguments, fruitful functions and void functions, anonymous functions	**8**	Understand Built-in Functions, Define Functions, Use Parameters and Arguments, Use Parameters and Arguments, Understand Fruitful Functions	**3,4**
IV	**Data Structures and strings in Python:** Lists – traversing a list, list operations, list slices, list methods Dictionaries – dictionary as a set of counters, looping and dictionaries, Tuples – tuple assignment, dictionaries and tuples, using tuples as	**12**	Understand and Traverse Lists, Perform List Operations, Utilize List Slices, Apply List Methods, Work with Dictionaries, Utilize Dictionaries as Counters, Understand Tuple Assignment, Work with Dictionaries and Tuples,	**4**

	keys in dictionaries Strings – string slices, looping and counting, the *in* operator, string comparison, string methods ,		Utilize String Slices, Loop through Strings and Count, Use the in Operator, Compare Strings, Apply String Methods	
V	**Basic File and Exception Handling in python:** Reading and Writing to Files, Exception Handling **Introduction to basic libraries for data analysis:** Introduction to Matplotlib library, NumPy, Pandas	**8**	Understand the Importance of Exception Handling, Implement Basic Exception Handling, Understand Basic Concepts of Matplotlib, Understand the Role of Numpy in Data Analysis, Use Numpy for Data Manipulation, Understand the Importance of Pandas in Data Analysis	**4**
Practical				
Practical 1	Program to find area and circumference of a circle.	**1**	Creating a python program to find the area and circumference of a circle.	5
Practical 2	Program to convert temperature from degree centigrade to Fahrenheit	**1**	Creating a python program to convert degree to Fahrenheit.	5
Practical 3	Program to calculate Sum of 5 subject and find Percentage	**1**	Creating a python program to calculate the sum of 5 subjects and find the percentage.	5
Practical 4	Program to show swap of two numbers by using third variable	**1**	Creating a python program to swap two numbers using a third variable.	5
Practical 5	Program to show swap of two numbers by without using third variable	**1**	Creating a python program to swap two numbers without using the third variable.	5
Practical 6	Program to reverse a given number	**1**	Creating a python program to reverse a given number.	5
Practical 7	Program to find greatest among 3 Numbers	**1**	Creating a python program to find the greatest number from three inputs.	5
Practical 8	Program to find whether the given number is even or odd	**1**	Creating a python program to find the given number is odd or not.	5
Practical 9	Program to use switch statement	**1**	Creating a python program to make use of the switch case statement.	3, 5
Practical 10	Program to display first 10 natural number and their sum	**1**	Creating a python program for displaying the natural numbers and its sum.	5
Practical 11	Program to find Fibonacci series up to a range	**1**	Creating a python program to print the Fibonacci series.	5
Practical 12	Program to count Number of digits	**1**	Creating a python program to count the number of digits.	5
Practical 13	Program to find factorial of a number	**1**	Creating a python program to find the factorial of a given number.	5
Practical 14	Program to check whether the given number is palindrome or not	**1**	Creating a python program to check a palindrome number	5

Practical 15	Program to find whether the given number is prime or not	**1**	Creating a python program to check for a prime number	5
Practical 16	Program to show dynamic list implementation	**1**	Creating a python program for dynamic lists.	5
Practical 17	Program to find dynamic array implementation	**1**	Creating a python program for dynamic array	5
Practical 18	Program to display matrix	**1**	Creating a python program to display a matrix	5
Practical 19	Program to show the sum of 10 elements of array and show their average	**1**	Creating a python program to calculate the sum of 10 elements of an array and show their average.	5
Practical 20	Program to show the sum of 10 elements of list and show their average	**1**	Creating a python program to calculate the sum of 10 elements of a list and show their average.	5
Practical 21	Program to find the maximum and minimum number in an array	**1**	Creating a python program to find the maximum and minimum number in an array	5
Practical 22	Program to find the maximum and minimum number in a list	**2**	Creating a python program to find the maximum and minimum number in a list	5
Practical 23	Program to create a list and perform the following operation ● Insert a new value in 4th index ● Delete the value from the 2nd index ● Replace the existing value of 1st position ● Insert the value at the end of the list ● Sort the list in ascending order ● Print the reverse of the list	**2**	Creating a python program insert, delete, and replace the value in the list.	5
Practical 24	Program to implement queue using list	**2**	Creating a python program to implement a queue.	5
Practical 25	Program to implement stack using list	**2**	Creating a python program to implement a stack	5
Practical 26	Program to show the sum of two matrices	**2**	Creating a python program to display the addition of two matrices.	5
Practical 27	Program to calculate the square of the elements of a list by using list comprehension method and print the final list	**2**	Creating a python program to calculate the square of the element of a list.	5
Practical 28	Program to explain tuple conversion function	**2**	Creating a python program understand the tuple conversion function.	5
Practical 29	Program to perform the following operation ● Creation of a tuple ● Updation of a tuple ● Deletion of a tuple Slicing of a tuple	**2**	Creating a python program for a different operation on a tuple.	5

TEXT BOOKS:

T1: Allen B. Downy, O'reilly, Think Python

T2: Charles Severance, Python for Everybody: Exploring Data Using Python 3

REFERENCE BOOKS:

R1: Jakes Vander Plas O' Reilly, Python Data Science Handbook

R2: Eric Matthes, "Python Crash Course, A Hands – on Project Based Introduction to Programming", 2nd Edition, No Starch Press, 2019

OTHER LEARNING RESOURCES:

1. https://www.w3schools.com/python/python_intro.asp
2. https://www.tutorialspoint.com/python/index.htm

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.	1, 2, 3
2	Express proficiency in the handling of strings and functions.	1, 2, 3, 4
3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets	1, 2, 3
4	Identify the commonly used operations involving file systems and regular expressions.	1, 2, 3
5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python	1, 2, 3

SEMESTER – IV									
Course Title	**Operating Systems**								
Course code	**24BCAM2204R**	**Total credits: 3** **Total hours: 45T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To learn the mechanisms of OS to handle processes and threads and their communication and the mechanisms involved in memory management in contemporary OS. 2. To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols 3. To know the components and management aspects of concurrency management.								
CO1	Explain the basic concepts of Operating Systems and related concepts.								
CO2	Summarize the concepts of processes and threads, process scheduling including Throughput, Turnaround Time, Waiting Time, Response Time.								
CO3	Identify the concept for optimally allocating memory to processes by increasing memory utilization and improving the access time.								
CO4	Demonstrate and implement the concepts of deadlocks and related concepts								
CO5	Implement various techniques of memory and file management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**7**	Understand the Concept of Operating Systems, Trace the Generations of Operating Systems, Identify Types of Operating Systems, Comprehend OS Services, Utilize System Calls, Understand OS Structures, Conceptualize the Virtual Machine	
II	**Processes:** Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. **Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, **Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.	**8**	Understand the Definition of a Process, Comprehend Process Relationships, Identify Different States of a Process, Understand Process State Transitions, Explore the Process Control Block (PCB), Understand Context Switching, Define Threads, Identify Various States of Threads, Understand the Benefits of Threads, Conceptualize Multithreading, Understand the Foundation and Objectives of Process Scheduling, Identify Types of Schedulers, Understand Scheduling Criteria, Explore Scheduling Algorithms, Comprehend Multiprocessor Scheduling	
III	**Inter-process Communication:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution **Deadlocks:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery	**8**	Understand the Concept of Inter-Process Communication, Comprehend the Critical Section Problem, Identify Race Conditions, Understand Mutual Exclusion, Explore Hardware Solutions for Mutual Exclusion, Understand the Definition of Deadlock, Identify Necessary and Sufficient Conditions for Deadlock, Understand Deadlock	

			Prevention, Explore Deadlock Avoidance, Comprehend Deadlock Detection and Recovery	
IV	**Memory Management:** Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging. **Virtual Memory**: Basics of Virtual Memory – Hardware and control, l structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).	**8**	Understand the Basic Concept of Memory Management, Comprehend Logical and Physical Address Mapping, Explore Memory Allocation Techniques, Identify Internal and External Fragmentation, Understand Compaction, Explore Paging, Understand the Basics of Virtual Memory, Comprehend Hardware and Control Structures for Virtual Memory, Explore the Concept of Locality of Reference, Understand Page Faults, Comprehend the Working Set Model, Understand Dirty Pages and Dirty Bits, Explore Demand Paging, Understand Page Replacement Algorithms,	
V	**I/O Hardware:** I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure.. **File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management. **Disk Management:** Disk structure, Disk scheduling, Disk reliability, Disk formatting, Boot-block, Bad blocks.		Understand the Different Types of I/O Devices, Comprehend Device Controllers, Understand Direct Memory Access (DMA), Comprehend the Goals of Interrupt Handlers, Understand Device Drivers, Explore Device-Independent I/O Software, Understand Secondary-Storage Structure, Understand the Concept of a File, Explore File Access Methods, Comprehend File Operations, Understand Directory Structure, Explore File System Structure, Understand File Allocation Methods, Comprehend Free-Space Management, Understand Disk Structure, Explore Disk Scheduling, Comprehend Disk Reliability, Understand Disk Formatting, Explore the Boot Block and Bad Blocks,	

TEXT BOOKS:

T1: Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.

T2: Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

REFERENCE BOOKS:

R1: Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing.

R2: Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley

R3: Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India

R4: Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Explain the basic concepts of Operating Systems and related concepts.	2, 3, 5,6,8
2	Summarize the concepts of processes and threads, process scheduling including Throughput, Turnaround Time, Waiting Time, Response Time.	1, 2, 3, 5, 6, 8
3	Identify the concept for optimally allocating memory to processes by increasing memory utilization and improving the access time.	2, 3, 5, 6, 8
4	Demonstrate and implement the concepts of deadlocks and related concepts	1,2
5	Implement various techniques of memory and file management.	1,2

SEMESTER – IV									
Course Title	**Predictive Analysis**								
Course code	**24BCAM2205R**	**Total credits: 3** **Total hours: 30L+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. Understand model building and evaluation techniques. 2. Use predictive analytics for business decision-making. 3. Gain hands-on experience with tools like IBM Watson and Python								
CO1	Understand and critically apply the concepts and methods of Predictive analytics.								
CO2	Understand and apply IBM SPSS Modeller in Data Mining, what kinds of data can be mined, what kinds of patterns can be mined.								
CO3	Applying and analysing how to use functions, deal with missing values, use advanced field operations, handle sequence data and improve efficiency.								
CO4	Get exposed to the testing of hypothesis and solving assumptions								
CO5	Understanding of IBM Watson Studio with Machine Learning Model.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**ANALYTICS OVERVIEW :**What is Predictive Analysis, how predictive model works, why predictive modelling, what are the models in Predictive Analysis. How Predictive analytics: Transforming data into future insights, Analytics trends: Past, Present & Future, Towards a Predictive enterprise.	**8**	Learners will understand how predictive models forecast future outcomes using historical data and apply analytics to solve business problems, while also exploring trends and transforming organizations into predictive enterprises.	1, 2
II	**Statistical Analysis :** Define Statistics and its types, Measures of Central Value; Mean, Median and Mode, Measures of Dispersion: Absolute and Relative measures of dispersion - Range, Quartile Deviation, Mean Deviation, Standard Deviation, Coefficient of Variation, Moments, Skewness, Kurtosis.	**8**	Upon completing the unit on statistical analysis, learners will be able to calculate and interpret	1, 2

			measures of central tendency and dispersion, understand data distribution through moments, skewness, and kurtosis, and apply these statistical techniques to analyze and make informed decisions from data.	
III	**DATA MINING**: What is a Data Mining application? Strategy for data mining: CRISP-DM, Steps of Data Mining, Stages and tasks in CRISP-DM, Life Cycle of a Data Mining Project, Skills Needed for Data Mining. Identify nodes and streams, The framework of a Data – mining project, Brief the unit of analysis, Explain the type of dialog box.	**8**	On completing the Data Mining unit, learners will be able to apply the CRISP-DM methodology, manage data mining projects, and effectively analyze and interpret data using various tools and techniques.	1,2
IV	**UNIT OF ANALYSIS :**Concepts of Unit of analysis (Distinct, Aggregate, SetToFlag), Integrate data, CLEM Expression, Role of Relationship between two fields, Identifying the modeling objective.	**8**	Upon completing this unit, learners will be able to identify units of analysis, integrate data, apply CLEM expressions, and define modeling objectives using field relationships.	1, 2, 3
V	**PREDICTIVE ANALYTICS WITH IBM WATSON STUDIO :** Understanding of IBM Cloud, IBM Watson Studio, Watson studio Components, Creating a Machine Learning Model, Data preparation, Watson Machine learning, Data Refinery, Watson Studio Neural Network Modeler, IBM Watson Studio jobs, Use case with AutoAI.	**8**	learners will be able to use IBM Cloud and Watson Studio tools to create machine learning models, prepare data, build neural	1, 2, 3

			network models, and apply AutoAI for real-world use cases.	
Practical				
Practical 1	Introduction to SPSS, Sorting File, Split File, Compute File, Recode File and Select Cases	**2**	Know the SPSS software which is used for statistical analysis, data management, and visualization in various fields like social sciences, health, and market research.	1
Practical 2	Chi- Square Test (Parametric and Non-Parametric Test)	**2**	The Chi-Square Test assesses associations between categorical variables, with the test for independence checking relationships and the goodness-of-fit test comparing observed vs. expected frequencies.	1, 2
Practical 3	Exploratory Factor Analysis	**2**	Exploratory Factor Analysis (EFA) identifies underlying relationships among variables by grouping them into factors, helping to simplify complex data structures.	2, 3

Practical 4	Cluster Analysis	**2**	Identify patterns and segment data into distinct groups.	2
Practical 5	Logistic Regression	**2**	Logistic Regression is a statistical method used to model the relationship between a dependent binary variable and one or more independent variables, predicting the probability of an event occurring.	3
Practical 6	Discriminant Analysis	**4**	Discriminant Analysis is a statistical technique used to classify cases into predefined groups based on predictor variables, aiming to find the best combination of variables that differentiates the groups.	3
Practical 7	Confirmatory Factor Analysis	**4**	Confirmatory Factor Analysis (CFA) is a statistical method used to test whether a hypothesized factor structure fits the observed data, validating the relationship between	3

			measured variables and underlying factors.	
Practical 8	Time Series	**4**	Time Series analysis involves analyzing data points collected or recorded at specific time intervals to identify trends, patterns, and forecast future values.	3
Practical 9	MANOVA	**4**	MANOVA (Multivariate Analysis of Variance) is a statistical test used to examine the effect of independent variables on multiple dependent variables simultaneously, assessing whether group differences exist across several outcomes.	3, 4
Practical 10	Decision Tree Analysis	**4**	Decision Tree Analysis is a predictive modeling technique that maps out possible decisions, outcomes, and their probabilities, helping to classify or	4

			predict data based on input features through a tree-like structure.	

TEXT BOOKS:

T1 : Eric Siegel, Predictive Analytics: The Power to Predict, 1st Edition, Wiley, 2013.

T2 :Anasse Bari, Mohamed Chaouchi, and Tommy Jung, Predictive Analytics for Dummies, 1st Edition, Wiley, 2016.

REFERENCE BOOKS:

R1 : Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. 1st edition. O'Reilly Media, 2013.

R2 : Linda Miner, Gary Miner, and DursunDelen, Practical Predictive Analytics and Decisioning Systems for Medicine, First Edition, Morgan Kaufmann, 2013

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and critically apply the concepts and methods of Predictive analytics.	**1, 2, 3, 4, 5, 12**
2	Understand and apply IBM SPSS Modeller in Data Mining, what kinds of data can be mined, what kinds of patterns can be mined.	**1, 2, 3, 4, 5**
3	Applying and analysing how to use functions, deal with missing values, use advanced field operations, handle sequence data and improve efficiency.	**1, 2, 3, 5**
4	Get exposed to the testing of hypothesis and solving assumptions	**1, 2, 3, 5**
5	Understanding of IBM Watson Studio with Machine Learning Model.	**1, 2, 5, 7, 8**

SEMESTER – IV									
Course Title	**Computer Networks**								
Course code	**24BCAO2203R**	**Total credits: 3** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Programming Skills**	**Co-requisite**	**LINUX**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives (Minimum 3)	1. Understand the fundamentals of data communications principles of media access, switching, routing and flow control. 2. Understand the basics of network protocol design and analysis. 3. Be familiar with the TCP/IP protocol suite and with application layer protocols. 4. Be able to write network-capable programs using the socket libraries. 5. Understand the structure of client-server systems and be able to build client-server programs.								
CO1	Explain the basics of data communication, networking, internet, physical layer techniques and circuit switching.								
CO2	Explain the different data link layer techniques and protocols including flow and error control.								
CO3	Discuss network layer protocols along with routing issues.								
CO4	Summarize transport and application layer operations and protocols along with QoS services								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to networks, internet, protocols and standards, the OSI model, layers in OSI model, TCP/IP suite, Addressing, Analog and digital signals. Physical Layer: digital transmission, multiplexing, transmission media, circuit switched networks, Datagram networks, virtual circuit networks, switch and Telephone network.	**8**	Understand the Concept of Computer Networks, Explain the Internet and its Components, Describe the OSI Model, Understand the TCP/IP Protocol Suite, Understand Signals in Networking, Define switching in networking	**1,2,3**
II	Data link layer: Introduction, Block coding, cyclic codes, checksum, framing, flow and error control, Noiseless channels, noisy channels, HDLC, point to point protocols	**8**	Introduction to Data Link Layer, Describe block coding and its use in error detection and correction, Describe block coding, Explain cyclic redundancy check (CRC), Define flow control and its importance	**3,4**
III	Network Layer: Logical addressing, internetworking, tunnelling, address mapping, ICMP, IGMP, forwarding, uni-cast routing protocols, multicast routing protocols.	**8**	Define logical addressing, Describe the concept of internetworking, Define tunnelling, Explain address mapping techniques	**3,4**
IV	Transport Layer: Process to process delivery, UDP and TCP protocols, SCTP, data traffic, congestion, congestion control, QoS, integrated services, differentiated services, QoS in switched networks.	**9**	Explain address mapping techniques, Define ICMP and its role in reporting errors, Explain IGMP and its role, Define forwarding, Describe unicast routing protocols	**4**
V	Application Layer – Domain name space, DNS in internet, electronic mail, FTP, WWW, HTTP, SNMP, multi-media, network security	**6**	Define domain name space, Describe the role of DNS, Define electronic mail, Define FTP and its role in transferring files, Describe the World Wide Web, Define HTTP and its role	**4**

TEXT BOOKS:,

T1: Behrouz A. Forouzan, Data Communications and Networking, Fourth Edition TMH,2006
T2: Andrew S Tanenbaum, Computer Networks, 4th Edition, Pearson Education.

REFERENCE BOOKS:

R1: S.Keshav, An Engineering Approach to Computer Networks, 2nd Edition, Pearson Education
R2: W.A.Shay, Understanding communications and Networks, Cengage Learning 3rd Edition
R3: Nader F. Mir, Computer and Communication Networks, Pearson Education
R4: James F.Kurose,K.W.Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 3rd Edition, Pearson Education

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Explain the basics of data communication, networking, internet, physical layer techniques and circuit switching.	1, 2, 3, 8
2	Explain the different data link layer techniques and protocols including flow and error control.	1, 2, 3, 8
3	Discuss network layer protocols along with routing issues.	1, 2, 3, 8
4	Summarize transport and application layer operations and protocols along with QoS services.	1, 2, 3, 8

SEMESTER – IV									
Course Title	**BASIC ACCLIMATIZING SKILLS (BAS)**								
Course code	**24UULS2201R**	**Total credits: 1** **Total hours: 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	0	2	0	0	0	1
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**All the Under Graduate Programmes**								
Semester	**Fall/I or Winter/II Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To impart knowledge of the fundamentals of Hospitality industry and its applications. 2. Students will be able to familiarize with the cooking equipments& Utensils. 3. Students will be able to handle different modes of reservations.								
CO1	Students will have basic knowledge of cooking methods.								
CO2	Students will gain the knowledge of organizing & Cleaning of Rooms.								
CO3	Students will be able to gain the travel management concept.								
CO4	Students will be able to acquire the knowledge of basic households amenities for day- to-day use.								
CO5	Understand the importance of time management and organization in acclimatizing to new academic demands.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Accommodation Management:** Telephone handling technique, Organizing of Rooms, Cleaning agents, Cleaning equipments and uses, Bed making Process.	**7**	Describe, illustrate, and explain cell organization and functions, microscopy, and structural differences.	1,2
II	**Fundamentals of Cooking**: Definition of cookery –Aim & Objectives of cooking, Use of basic Cooking equipments, Personal Hygiene and Safety, Use of Fire & Fuels	**10**	Describe, illustrate, and explain membrane structure, function; cell organization, and the proteins involved in transportation.	1,2
III	**Methods of Cooking:** Different Cuts. Use of Herbs and Spices. Basic Food and Beverage Preparation. Regional food Habits	**10**	Describe, illustrate, and explain chromosomal structure and types.	1,2
IV	**Forms & Format's**: C –form Reservation form Registration form Passport Application form Legal Rent Agreement	**8**	Describe, illustrate, and explain the mechanism of cell-to-cell communication	1,2
Practical	1. Staining and microscopic observation of various stages of Mitosis of given sample(s). 2. Staining and microscopic observation of various stages in Meiosis of given sample(s).	**30**	Describe, illustrate and explain and apply staining techniques and carry out microscopic examination.	1,2,3,4

TEXT BOOKS:

T1 Arora K (2011). Theory of cookery, Frank brothers & company (pub) pvtltd-New Delhi.

T2 Bruce H. Axler, Carol A. Litrides (2010) Food and Beverage Service Volume 1 ofWiley Professional Restauranteur, Guides.

T3 Mohammed Zulfikar (2010) - Introductions to Tourism and Hotel Industry Introduction to Tourism and Hotel Industry. Vikas Publishing.

T4 Sudhir Andrews (2013) Food and Beverage Service: A Training Manual, Tata McGraw Hill, 2013.

REFERENCE BOOKS:

R1: Cooper GM. The Cell: A Molecular Approach. 2nd edition. Sunderland (MA): Sinauer Associates; 2000.

R2: Ambrose and Dorothy. Cell Biology. 2nd Edition. MEasty, ELBS Publications; 1970.
R3: Sharp, Lester W. Fundamentals of Cytology. 1st edition. Mc Graw Hill Company; 1943.
OTHER LEARNING RESOURCES:
https://www.ncbi.nlm.nih.gov/books/NBK9839/?term=cell%20Biolpgy

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Able to explain the fundamentals and advances of cytology including structure and functions of cell and cell organelles.	1,3 & 4
2	Able to explain the cell cycle and cell division.	1,2
3	Learn and develop skills for operating microscope, preparing slides by various staining techniques	7.9,10
4	Apply knowledge of cellular processes to explain how cells operate and interact within living organisms.	5,7
5	Demonstrate a comprehensive understanding of cell structure and function.	5,8

SEMESTER – IV									
Course Title	**Computational Systems and Digital World**								
Course code	**24UCDL2201R**	**Total credits: 1** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite		**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall / II semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To understand the fundamentals of computer systems and Internet search along with advanced features of Ms Office. 2. To learn data management, statistical analysis and visualization. 3. To use social media and e-commerce portals, Digital Payment systems, and other utility software.								
CO1	Basic understanding of computer systems and internet search.								
CO2	Solve data analysis, management and visualization issues using Ms Office products.								
CO3	Efficiently and ethically use social media and e-commerce sites.								
CO4	Introduction to various utility software used in research and information management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Fundamentals of Computer Systems, Office Automation and Internet Search:** Components of a computer and their functions; Office automation using Ms Word, Ms Excel, and Ms PowerPoint; Data management, statistical data analysis and data visualization with Ms Excel; Use of functions, graphs, and charts in Ms Excel.	**4**	Understand the components of computers and its functions. Apply the knowledge of different Ms Office for different real-world problem.	2,3
II	**Internet and Cyber World:** Introduction to computer networks, internet and world wide web, websites and web portals; Creation and use of email accounts; Web browsing, web searching, different aspects of web searching – search keyword, conditions and combinations; Study different search engines like Google, Microsoft Bing, Yahoo, Yandex, DuckDuckGo, Ask.com, etc; Cyber Crimes, cyber laws and IT Act 2000, India.	**4**	Understanding the basics of Computer network, internet and WWW. Creating email accounts for different purpose. Understanding the search engine and study the different search engines.	2, 3, 5
III	**Introduction to Social-Media and E-Commerce:** Relevance of Social-Media in present scenario. Posting different types of contents in Social-Media; Creating accounts and using some popular social media portals and Apps like WhatsApp, Facebook, etc. Social Media etiquettes and crimes; Definition of e-commerce, e-commerce versus traditional commerce; Cast studies of popular e-commerce portals like Amazon; E-commerce etiquettes and crimes.	**4**	Understand what social-media is, and what is the impact of social media on different field. Creating social-media account of different platforms. Understanding e-commerce and case studying on popular e-commerce portals.	2,3,5
IV	**Digital payments and Digital Transactions:** Introduction to digital payment systems; Creating accounts and using Digital Payment Systems	**4**	Understand the process if digital payment and transactions system. Creating an account for	2, 3, 5

	like Credit Cards, Debit Cards, Net banking, UPI; Digital payments etiquettes and crimes.		digital payment.	
V	**Basic Accounting and Utility Software:** Introduction to Basic accounting concepts; Introduction to an Accounting Software like GnuCash or Tally; Introduction to Technical Document writing using LaTex; Introduction to Data Visualization software – Sigma, Google Charts, Tableau.	**4**	Equips you with basic accounting principles, accounting software usage, technical document writing with LaTex, and data visualization tools for practical application.	2,3,5
Practical				
Practical 1	Create a new document with Ms Word, save it with the name "*Student_Name*" in the Documents folder. Formatting text, paragraph alignment.	**1**	Create, save, and format documents in Ms Word, including text formatting and paragraph alignment.	5
Practical 2	Creating tables, inserting shapes and images. Creating profile with description and wrapping the images.	**1**	Create tables, insert and format images and shapes in Ms Word documents. Design a profile with a description and incorporate image wrapping for a visually appealing layout.	5
Practical 3	Using mail merge, creating documents with existing templates, enable track changes.	**1**	Utilize mail merge in Ms Word to create personalized documents from existing templates. Enable track changes functionality for collaborative document editing in Ms Word.	5
Practical 4	Create a simple document with title, author, and body text. Learn basic text formatting options like bold, italic, and underline.	**1**	Create a basic LaTex document with proper structure. Gain control over text appearance and presentation in your document.	5
Practical 5	Structure your document with sections and subsections. Learn how to write mathematical equations in LaTex.	**1**	Organize your document for better readability and navigation. Represent mathematical expressions clearly in your document.	5
Practical 6	Creating a presentation with Ms PowerPoint, adding and formatting text	**1**	Create presentations in Ms PowerPoint by adding and formatting text content.	5
Practical 7	Applying slide layouts, using transitions, inserting and formatting shapes.	**1**	Apply various slide layouts and transitions for visually engaging presentations in Ms PowerPoint. Insert and format shapes effectively in Ms PowerPoint presentations.	5
Practical 8	Creating and managing animations, using slide master, embedding and linking media.	**1**	Create and manage animations in Ms PowerPoint presentations to enhance visual impact. Utilize the slide master feature in Ms PowerPoint to maintain consistent formatting across slides.	5

Practical 9	Creating new excel file, entering data, formatting cells, using basic formulas.	**1**	Create and manage spreadsheets in Ms Excel by entering and formatting data. Utilize basic formulas in Ms Excel for calculations.	3, 5
Practical 10	Creating and formatting table, using functions like VLOOKUP, IF, CONCATENATE, etc.	**1**	Create and format tables in Ms Excel for organized data presentation. Utilize advanced formulas like VLOOKUP, IF, and CONCATENATE in Ms Excel for complex calculations and data manipulation.	5
Practical 11	Creating and inserting graphs using different kind of data.	**1**	Create and insert various charts and graphs to visually represent data in Ms Excel.	5
Practical 12	Use of advanced formulas, data validation, pivot tables, macros, and automation.	**1**	Utilize advanced formulas in Ms Excel, including data validation for error prevention. Create pivot tables in Ms Excel for data summarization and analysis. Gain an introduction to macros and automation for repetitive tasks in Ms Excel.	5
Practical 13	Exploration on search engines, how it works and so on.	**1**	Understand the basic principles of how search engines work and explore techniques for effective online searching.	2
Practical 14	Creating a social media account, e-commerce websites, etc. Upload images, videos, exploring the e-commerce websites and learning about etiquette and crimes.	**1**	Create social media accounts and navigate e-commerce websites. Understand online etiquette for responsible interactions on social media platforms. Be aware of potential crimes associated with social media and e-commerce platforms.	5
Practical 15	Creating an account in any platform for digital payments and learning about digital payment etiquettes and crimes.	**1**	Create accounts for digital payments. Understand safe practices for online transactions and be aware of potential digital payment crimes.	5
Practical 16	Learn how to create basic tables in LaTex.	**1**	Present tabular data in a well-organized manner.	5

TEXT BOOKS:

T1: Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems & Applications. 3rd ed. New Delhi: BPB Publications.
T2: Goel, A, 2010. Computer Fundamentals, Pearson India.

REFERENCE BOOKS:

R1: Balaguruswamy, E. 2009 Fundamentals of Computers, Tata McGraw-Hill Education.
R2: Balaguruswamy, 2014. E. Fund Of Comp & Programming (Updated Ed Sem. I, Au) Tata McGraw-

Hill Education.
R3: Lawson, C. 2022. Introduction to Social Media, Oklahoma State University.

OTHER LEARNING RESOURCES:

- https://www.w3schools.com
- https://edu.gcfglobal.org
- https://www.tutorialspoint.com
- https://www.javatpoint.com
- Latest updates available in WWW.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Basic understanding of computer systems and internet search.	1,2,4,8
2	Solve data analysis, management and visualization issues using Ms Office products.	2,3,4,7
3	Efficiently and ethically use social media and e-commerce sites.	5,6,7
4	Introduction to various utility software used in research and information management.	1,4,8

SEMESTER – IV									
Course Title	**MINI PROJECT-I**								
Course code	24BCAM2207R	**Total credits: 2 Total hours:60P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	0	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives (Minimum 3)	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: ● theoretical studies ● computer simulations ● hardware construction; 2. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 3. Deliver a seminar on the general area of work being undertaken and specific contributions to that field; 4. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	3, 4, 5

TEXT BOOKS:
REFERENCE BOOKS:
OTHER LEARNING RESOURCES:
As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – V									
Course Title	**ADVANCE APPLICATION DEVELOPMENT**								
Course code	**24BCAM3101R**	**Total credits: 4** **Total hours: 40L+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter / V semester of the third year of the program**								
Course Objectives (Minimum 3)	1. To understand the MERN stack for application development. 2. To develop full-stack proficiency for front-end and back-end development. 3. To create RESTful APIs to handle requests and responses.								
CO1	Understanding the concept of MERN stack.								
CO2	Create RESTful APIs using Express.js.								
CO3	Implementing authorization and authentication using JWT.								
CO4	Creating UI and responsive design using React.js								
CO5	Implementing authorization and authentication in an application.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to MERN Stack:** Introduction to Full-Stack Development - Overview of MERN Stack components (MongoDB, Express.js, React.js, Node.js) - Setting up development environment with Node.js and package managers (npm/yarn)	**8**	This unit lays the groundwork for the MERN stack by establishing a foundational understanding of full-stack development, exploring the core components (MongoDB, Express.js, React.js, Node.js), and setting up the development environment.	1,2
II	**Databases and MongoDB and JavaScript Deep Dive:** Introduction to database concepts (relational vs. NoSQL) - MongoDB fundamentals (data models, documents, collections) - Connecting to MongoDB from Node.js applications - Mongoose ODM (Object Data Modeling) for MongoDB; Advanced JavaScript concepts (ES6+, closures, modules), Asynchronous programming (promises, async/await), Introduction to DOM manipulation and event handling	**12**	This unit dives deeper into data management, exploring relational vs. NoSQL concepts, the fundamentals of MongoDB (data models, documents, collections), and connecting to it from Node.js applications using Mongoose ODM. Additionally, students will solidify their grasp of advanced JavaScript features (ES6+, closures, modules) and asynchronous programming (promises, async/await) while gaining an introduction to DOM manipulation.	2,3
III	**Building User Interfaces with React.js:** Introduction to React components (functional, class-based) - JSX syntax and component lifecycle - State management with React hooks (useState, useEffect); Introduction to	**12**	This unit focuses on building interactive user interfaces with React.js. Students will learn to create reusable components (functional and class-	2,3

	routing and user authentication		based), work with JSX syntax and the component lifecycle, manage application state effectively using hooks (useState, useEffect), and gain a basic understanding of routing and user authentication in React applications.	
V	**Backend Development with Node.js & Express.js:** Introduction to Node.js and server-side programming concepts - Building RESTful APIs with Express.js - Connecting to MongoDB databases using Mongoose ODM; Implementing user authentication and authorization.	**12**	This unit delves into server-side development using Node.js and Express.js. Students will learn to build RESTful APIs, connect Node.js applications to MongoDB databases with Mongoose ODM, and implement user authentication and authorization mechanisms to secure their backend applications.	3,4
V	**Deployment and Project Development:** Testing strategies for web applications (unit testing, integration testing); Deployment considerations and options (cloud platforms); Introduction to CI/CD pipelines.	**10**	Explore testing strategies (unit testing, integration testing), evaluate deployment considerations (cloud platforms), gain an introduction to CI/CD pipelines, and culminate the course by developing a full-stack web application using the MERN stack, demonstrating their ability to apply the learned concepts to real-world development.	3,5

Practical

Practical	Content	Contact Hour	Learning Outcome	KL
Practical 1	Install Node.js and npm/yarn on your machine. Create a simple "Hello World" application using Node.js. Initialize a new Node.js project using npminit or yarn init. Install a few packages (e.g., lodash) and demonstrate their usage in your project.	**2**	Learn to install Node.js and npm/yarn, and initialize a Node.js project.	3
Practical 2	Set up a basic folder structure for a MERN stack application. Create a server.js file to serve as the entry point for your Node.js server. Set up a simple Express server with a few routes (/, /about, /contact).	**2**	Understand the basic folder structure for a MERN stack application and set up an Express server.	3
Practical 3	Create a RESTful API using Express.js with endpoints for GET, POST, PUT, and DELETE operations.	**2**	Develop a RESTful API with Express.js, handling various HTTP operations	5

	Use middleware to handle errors and log requests.		and middleware.	
Practical 4	Install MongoDB and set up a local database. Create a collection and insert a few documents using the MongoDB shell.	**2**	Install and set up MongoDB, and perform basic operations using the MongoDB shell.	3
Practical 5	Write a few ES6+ features like arrow functions, template literals, and destructuring.	**2**	Apply modern JavaScript features such as arrow functions, template literals, and destructuring.	3
Practical 6	Create a Node.js application that connects to a MongoDB database. Perform basic CRUD operations using Mongoose ODM.	**2**	Create a Node.js application that connects to MongoDB and performs CRUD operations using Mongoose.	3
Practical 7	Write functions using promises and async/await for asynchronous operations. Manipulate the DOM using vanilla JavaScript to create a simple interactive page.	**2**	Write asynchronous functions with promises and async/await, and manipulate the DOM with JavaScript.	3
Practical 8	Define complex data models with relationships (e.g., user and posts). Implement advanced queries and data validation using Mongoose.	**2**	Define complex data models and implement advanced queries with Mongoose.	5
Practical 9	Create a small project demonstrating closures, modules, and event handling.	**2**	Demonstrate JavaScript concepts like closures and modules through a small project.	5
Practical 10	Set up a new React project using Create React App. Create functional and class-based components. Write components using JSX. Implement component lifecycle methods in class-based components.	**2**	Set up a React project, create components, and manage component lifecycle methods.	3
Practical 11	Use React hooks like useState and useEffect to manage state and side effects. Create a simple form with state management.	**2**	Manage state and side effects in React using hooks like useState and useEffect.	3
Practical 12	Implement routing in your React application using react-router-dom.	**2**	Implement routing in a React application using react-router-dom.	3
Practical 13	Manage complex state using context API or state management libraries like Redux. Implement user authentication and conditional rendering based on user state.	**2**	Manage complex state and implement user authentication in a React application.	3
Practical 14	Design and implement reusable components with proper props and state management.	**2**	Design and implement reusable components with proper props and state management.	5
Practical 15	Set up a basic Express server with routes for different HTTP methods. Create middleware for request logging.	**2**	Set up an Express server with various HTTP method routes and create middleware for logging.	3
Practical 16	Develop a RESTful API with endpoints for creating, reading, updating, and deleting resources.	**2**	Develop a RESTful API connected to MongoDB and implement user	5

	Connect the API to a MongoDB database using Mongoose. Implement user authentication using JWT (JSON Web Tokens).		authentication using JWT.	
Practical 17	Implement user authorization and role-based access control. Create and handle complex query parameters and request validation.	**2**	Implement user authorization and role-based access control in an API.	5
Practical 18	Write unit and integration tests for your API using testing libraries like Mocha or Jest.	**2**	Write unit and integration tests for APIs using testing libraries like Mocha or Jest.	6
Practical 19	Develop a full-fledged MERN stack application with user authentication, data validation, and complex state management. Implement advanced features like real-time updates using WebSockets or data visualization.	**4**	Develop a comprehensive MERN stack application with advanced features like real-time updates and data visualization.	5

TEXT BOOKS:

T1: "Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node" by Vasan Subramanian

T2: "Learning React: A Hands-On Guide to Building Web Applications Using React and Redux" by KirupaChinnathambi

REFERENCE BOOKS:

R1: "Node.js Design Patterns" by Mario Casciaro

R2: "Express.js Guide: The Comprehensive BookonExpress.js" by AzatMardan

R3: "MongoDB: The Definitive Guide" by Shannon Bradshaw, Kristina Chodorow, and Michael Dirolf.

OTHER LEARNING RESOURCES:

- https://frontendmasters.com/
- https://www.codecademy.com/catalog
- https://v2.scrimba.com/
- https://learn.mongodb.com/
- https://expressjs.com/
- https://nodejs.org/en/download/package-manager
- https://www.freecodecamp.org/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understanding the concept of MERN stack.	1,2,3
2	Create RESTful APIs using Express.js.	1,2,3,5
3	Implementing authorization and authentication using JWT.	2,3,4,5,6
4	Creating UI and responsive design using React.js	1,2,3,5,7
5	Implementing authorization and authentication in an application.	2,3,4,5,6,8

<table>
<tr><td colspan="9">SEMESTER – V</td></tr>
<tr><td>Course Title</td><td colspan="8">PROJECT-I</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BCAM3204R</td><td rowspan="2">Total credits: 6
Total hours:60P</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>4</td><td>12</td><td>6</td><td>0</td><td>6</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="6">Nil</td></tr>
<tr><td>Programme</td><td colspan="8">Bachelor of Computer Application</td></tr>
<tr><td>Semester</td><td colspan="8">Fall/ V semester of the third year of the program</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="8">1. To develop the knowledge, skills and attitudes of a professional Computer Application professional.
2. To become confident in designing Software solutions to complex software problems utilising a systems approach.
3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.</td></tr>
<tr><td>CO1</td><td colspan="8">Demonstrate a sound technical knowledge of their selected project topic.</td></tr>
<tr><td>CO2</td><td colspan="8">Undertake problem identification, formulation and solution.</td></tr>
<tr><td>CO3</td><td colspan="8">Design Software solutions to complex problems utilising a systems approach.</td></tr>
<tr><td>CO4</td><td colspan="8">Communicate with the community at large in written an oral forms.</td></tr>
<tr><td>CO5</td><td colspan="8">Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="2">Learning Outcome</td><td>KL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
o theoretical studies
o computer simulations
o hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the</td><td>90</td><td colspan="2">Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.</td><td>3, 4, 5</td></tr>
</table>

	general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

TEXT BOOKS:
REFERENCE BOOKS:
OTHER LEARNING RESOURCES:
As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – V									
Course Title	**UI/UX DESIGN**								
Course code	**24BCAM3102R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter / I semester of third year**								
Course Objectives (Minimum 3)	1. To introduce students to industry-standard UI/UX design tools such as Figma and Visily, providing hands-on experience in creating wireframes, prototypes, and interactive designs. 2. To introduce the fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, enabling them to create aesthetically pleasing and user-friendly interfaces. 3. To develop skills in creating interactive prototypes and user flows, utilizing advanced features of Figma and Visily to simulate real-world user interactions and test design concepts. 4. To guide in creating and implementing design systems, ensuring consistency and reusability of UI components across various projects, and promoting efficient team collaboration. 5. To equip with the knowledge and skills to effectively integrate UI/UX designs into development workflows, including exporting assets, collaborating with developers, and managing content within CMS or e-commerce platforms.								
CO1	Demonstrate proficiency in using Figma and Visily for designing wireframes, prototypes, and interactive UI components, leveraging these tools to create user-centered designs.								
CO2	Understand and apply fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, to create visually appealing and functional interfaces.								
CO3	Develop interactive prototypes and user flows, using advanced features of Figma and Visily to simulate real-world user interactions and test design hypotheses.								
CO4	Create and implement comprehensive design systems, ensuring consistency and reusability of UI components across multiple projects, and facilitating effective collaboration within design teams.								
CO5	Integrate UI/UX designs with development workflows, exporting assets and collaborating with developers to ensure seamless transition from design to implementation, and managing content within CMS or E-commerce platforms								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to UI/UX Design:** Definition of UI and UX, Importance of UI/UX Design, Key Differences between UI and UX; User Research Methods, Creating User Personas, User Journey Mapping; Introduction to Design Thinking, Phases of Design Thinking, Empathy in Design; Types of Prototypes, Tools for Prototyping, Low-fidelity vs. High-fidelity Prototypes.	**8**	Understanding the concept of UI/UX design, differentiate between UI and UX, understanding	1, 4
II	**UI Design Fundamentals:** Introduction to Figma, Sketch, Visily, and Adobe XD; Color Theory, Typography, Layout and Composition; Buttons, Icons, Forms, and Input Fields; Responsive Design, Mobile-First Design, Grid Systems.	**10**	Understand the theory of color, typography with different tools of designing. Apply buttons, icons, forms, and inputs. Creating responsive design.	2, 3, 5
III	**UX Design Fundamentals:** User Research Techniques - Surveys, Interviews, Focus Groups, Usability Testing; Information Architecture - Site Mapping, Card Sorting, Navigation Design; Interaction Design - Principles of Interaction Design, Designing User Flows, Micro-interactions; Wireframing	**10**	Applying research techniques through surveys, interviews, focus groups. Designing navigation, sorting card.	3, 4, 5

	and Mockups - Importance of Wireframing, Tools for Wireframing, Creating High-fidelity Mockups			
IV	**Advance UI/UX Design Concepts:** Accessibility in design - Principles of Accessible Design, WCAG Guidelines, Tools for Checking Accessibility; Design Systems - What is a Design System, Benefits of Design Systems, Creating a Design System; Advanced Prototyping Techniques - Interactive Prototypes, Animation and Transitions, Tools for Advanced Prototyping; Usability Testing and Evaluation - : Planning a Usability Test, Conducting Usability Tests, Analyzing Results.	**12**	Understanding the accessibility design, using tools for checking the accessibility, creating a design system, and interactive prototypes, animations and transitions, analyzing the results.	2,3, 4,5
V	**UI/UX Designing and Portfolio Development:** Project Planning and Scope - Defining Project Goals, creating a Project Plan, Setting Milestones; Design Execution - Iterative Design Process, Collaboration and Feedback, Design Handoff; Building a Portfolio - Importance of a Portfolio, Elements of a Strong Portfolio, Showcasing Projects; Final Presentation.	**12**	Creating a project by applying all the techniques and tools.	3, 5
Practical				
Practical 1	Create a simple wireframe for a landing page using basic shapes and text in Figma. Explore Basic Tools and Interface Components in Figma	**2**	Gain familiarity with Figma's interface and basic tools. Understand how to use shapes and text to create a simple wireframe.	1,5
Practical 2	Design a Simple Wireframe for a Landing Page. Plan and sketch a wireframe layout for a landing page.	**2**	Practice planning and sketching wireframe layouts. Learn to structure content effectively for a landing page.	5
Practical 3	Create multiple layers by adding different shapes and text. Rename layers and organize them using frames and groups. Use the layers panel to reorder and hide/show layers.	**2**	Understand how to manage layers and organize them using frames and groups in Figma. Learn efficient workflow practices.	2, 5
Practical 4	Use Figma's layout grids and frames to structure the wireframe. Add placeholders for images, text blocks, and buttons.	**2**	Master the use of layout grids and frames to structure wireframes effectively. Practice adding placeholders for various content elements.	3
Practical 5	Define a color palette using Figma's color picker or hex codes. Apply colors to different elements (background, text, shapes) in your wireframe. Use the color styles panel to create reusable color swatches	**2**	Develop skills in defining and applying cohesive color palettes. Learn to use Figma's color styles for efficient design consistency.	5
Practical 6	Design a component (e.g., card, button) using Figma's vector tools. Convert the component into an Auto Layout frame. Resize the frame to see how elements adjust automatically.	**2**	Create and manipulate components using Figma's Auto Layout feature. Understand how to design responsive UI elements.	3
Practical 7	Create a new frame sized for a mobile screen (e.g., iPhone X). Design UI components using adaptive variants for different screen sizes. Preview the design in Figma's device preview mode.	**2**	Practice designing UI components for mobile screens. Learn to preview designs and adjust for different device sizes in Figma.	3
Practical 8	Design multiple screens for a simple app or website flow. Use Figma's prototyping feature	**2**	Gain proficiency in prototyping with Figma. Learn to link screens, define	4

	to link screens together. Define interactions (e.g., click, hover) to simulate user navigation		interactions, and simulate user navigation.	
Practical 9	Design custom icons or illustrations using Figma's vector tools. Experiment with stroke styles, fills, and path operations. Use these assets to enhance your wireframes or UI designs.	**2**	Develop skills in designing custom icons and illustrations using Figma's vector tools. Understand stroke styles and path operations.	5
Practical 10	Define a design system for consistent UI elements (buttons, forms, typography). Create component libraries and style guides using Figma's design tokens. Share the design system with your team for reuse across projects.	**2**	Learn to create and manage design systems in Figma. Practice using design tokens and component libraries for consistent UI elements.	6
Practical 11	Set up design handoff using Figma plugins or integrations (e.g., Zeplin, Avocode). Export CSS, XML, or JSON code snippets for developers.	**2**	Explore Figma plugins for design handoff. Learn to export CSS, XML, or JSON code snippets for developers.	6
Practical 12	Creating sitemaps and user flows.	**2**	Practice creating sitemaps and user flows. Understand how to visually represent website or app structures and user interactions.	5
Practical 13	Designing user flows, Interactive wireframes, User Flow Design in Figma	**2**	Develop skills in designing interactive wireframes and user flows using Figma's prototyping features.	4
Practical 14	Use Figma's prototyping and animation features to create and simulate micro-interactions within a mobile app interface.	**2**	Learn to create and simulate micro-interactions and animations within a mobile app interface using Figma's prototyping tools.	5
Practical 15	Create two variants of the same web page in Figma, focusing on different design elements (e.g., button placement, color schemes) for A/B testing.	**2**	Practice creating different variants of web pages for A/B testing. Understand the impact of design elements on user behavior.	6
Practical 16	Use Figma's vector tools to create and customize basic shapes (rectangles, circles, lines) and combine them to form UI elements.	**2**	Master Figma's vector tools to create and customize basic UI elements like rectangles, circles, and lines.	5
Practical 17	Set up a grid system in Figma and use guides to ensure consistent spacing and alignment across your design.	**2**	Develop skills in setting up grid systems and using guides for consistent spacing and alignment in Figma designs.	3
Practical 18	Use Figma's text tools to add, style, and organize text elements, and experiment with different fonts and typographic hierarchies.	**2**	Practice adding, styling, and organizing text elements. Experiment with fonts and typographic hierarchy in Figma.	3
Practical 19	Import and position images and video placeholders, and adjust their properties to create an engaging design.	**2**	Learn to import, position, and adjust properties of images and video placeholders in Figma designs.	3
Practical 20	Create a Visily Account and Start Your First Project, add basic shapes and text elements to the canvas.	**2**	Gain familiarity with Visily's interface and basic tools. Learn to add and customize UI elements using drag-and-drop.	1,5
Practical 21	Explore Basic Tools and Interface Components in Visily: Use Visily's drag-and-drop tools to add and customize UI elements. Arrange and group elements for organization.	**2**	Practice designing form layouts and implementing input validation rules using Visily's form elements.	5
Practical 22	Create Interactive Forms with Input Fields and Validation. Design a form layout using Visily's form elements (text inputs, checkboxes).	**2**	Learn to plan and structure wireframes using Visily's layout grids and containers. Incorporate	3

	Implement input validation rules (e.g., required fields, email format).		placeholder content effectively.	
Practical 23	Plan a wireframe layout for a website homepage (header, hero section, features). Use Visily's layout grids and containers to structure the wireframe. Add placeholder content such as images and text blocks.	**2**	Develop skills in defining color schemes and applying colors to various elements across pages using Visily.	3
Practical 24	Define a color scheme using Visily's color picker or palette options. Apply colors to different elements (background, text, buttons) across pages.	**2**	Practice creating navigation structures and interactive components like dropdown menus in Visily. Apply hover effects for interactive navigation.	3
Practical 25	Create a navigation bar or menu structure in Visily. Add dropdown menus using Visily's interactive components. Apply hover effects to menu items for interactive navigation.	**2**	Master Visily's grid and flexbox features to create responsive grid layouts for webpages or app screens.	3
Practical 26	Create a responsive grid layout in Visily for a webpage or app screen. Use Visily's grid and flexbox features to align elements across different screen sizes.	**2**	Develop proficiency in designing multiple pages and linking them together using interactive elements in Visily.	4
Practical 27	Design multiple pages for a website prototype in Visily. Link pages together using interactive elements (e.g., buttons, text links). Define transition effects and animations for a seamless user experience.	**2**	Learn to design and convert templates into reusable components in Visily for efficient project workflow.	5
Practical 28	Create a responsive grid layout in Visily for a webpage or app screen. Use Visily's grid and flexbox features to align elements across different screen sizes.	**2**	Practice integrating Visily with Content Management Systems (CMS) or E-commerce platforms for efficient content management.	3
Practical 29	Design reusable templates (e.g., headers, footers) in Visily. Convert templates into custom components for easy reuse across projects.	**2**	Design and develop reusable templates in Visily, converting them into custom components for consistent and efficient use across multiple projects.	5
Practical 30	Integrate Visily with CMS or E-commerce Platforms for Content Management	**2**	Integrate Visily designs with CMS or E-commerce platforms, enhancing their ability to manage and deploy content effectively in real-world applications.	3

TEXT BOOKS:

T1: "Don't Make Me Think, Revisited: A Common-Sense Approach to Web Usability" by Steve Krug."

T2: "The Design of Everyday Things" by Don Norman."

T3: "Interaction Design: Beyond Human-Computer Interaction" by Yvonne Rogers, Helen Sharp, and Jenny Preece"

REFERENCE BOOKS:

R1: "Sketching User Experiences: Getting the Design Right and the Right Design" by Bill Buxton"

R2: "Designing Interfaces: Patterns for Effective Interaction Design" by Jenifer Tidwell"

OTHER LEARNING RESOURCES:

https://www.interaction-design.org/courses/user-experience-the-beginner-s-guide

https://uxplanet.org/

https://material.io/design/Software/Hardware Tools to be used in

"Figma for UI Design" (Online Course Material)

MOOCS:

https://www.coursera.org/learn/ux-design-fundamentals

https://www.coursera.org/learn/introtoux-principles-and-processes?specialization=michiganux https://www.coursera.org/learn/visual-elements-user-interface-design https://www.udemy.com/course/ui-ux-web-design-using-adobe-xd/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Demonstrate proficiency in using Figma and Visily for designing wireframes, prototypes, and interactive UI components, leveraging these tools to create user-centered designs.	1,2,3,4
2	Understand and apply fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, to create visually appealing and functional interfaces.	1,3,5
3	Develop interactive prototypes and user flows, using advanced features of Figma and Visily to simulate real-world user interactions and test design hypotheses.	2,3,4
4	Create and implement comprehensive design systems, ensuring consistency and reusability of UI components across multiple projects, and facilitating effective collaboration within design teams.	3,5,6,7
5	Integrate UI/UX designs with development workflows, exporting assets and collaborating with developers to ensure seamless transition from design to implementation, and managing content within CMS or E-commerce platforms	3,4,6,8

SEMESTER – V									
COURSE TITLE	**Artificial Intelligence and Deep learning**								
COURSE CODE	**24BCAM3203R**	**TOTAL CREDITS**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**4**	**0**	**0**	**0**	**5**
PRE-REQUISITE	**Basic Python, Data collection**	**CO-REQUISITE**	**OLAP, ETL**						
ANTI-REQUISITE	**NIL**								
PROGRAMME	**BCA**								
SEMESTER									

Course Objectives:

1. Explore the role of AI in modern society and its potential impact on various industries.
2. Learn the fundamentals of deep learning, including neural network architecture, activation functions, and optimization algorithms.
3. Explore cutting-edge architectures like convolutional neural networks (CNNs) and their applications in image processing. apply IBM cloud Service for NLP.

Course Outcomes:

On successful completion of the course, the students will be able to:

CO1: Recall fundamental concepts and principles of artificial intelligence and deep learning.

CO2: Analyze deep learning models using metrics and evaluation techniques.

CO3: Develop chatbot using IBM Watson service.

CO4: Enhance problem-solving abilities with IBM Cloud service.

CO5: Apply EDA to real-world data.

Course Description:

This course provides a comprehensive AI for students with a foundational understanding of key concepts such as machine learning and deep learning, complemented by practical applications using IBM Watson services and IBM Natural Language Processing (NLP) tools. Through hands-on experience with these industry-leading technologies, learners develop proficiency in leveraging AI techniques to solve real-world problems effectively. Additionally, the course emphasizes critical thinking skills necessary to navigate the ethical considerations inherent in AI technology, ensuring responsible innovation and preparing students for success in today's rapidly evolving AI landscape.

Text Books:

1. Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
2. "Deep Learning for Computer Vision" by RajalingappaaShanmugamani

3. Hands-On Artificial Intelligence for IBM Watson: Explore the Power of AI to Innovate and Transform Business Processes" by Dr. Harish Garg
4. "Grokking Deep Learning" by Andrew W. Trask

Reference Books:

1. "Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence" by Jon Krohn, Grant Beyleveld, and AglaéBassens
2. Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth
3. Hands-On Artificial Intelligence for IBM Watson: Explore the Power of AI to Innovate and Transform Business Processes" by Dr. Harish Garg

Additional Resources:

- Python - https://docs.python.org/3/
- AI with IBM - https://developer.ibm.com/technologies/artificial-intelligence/
- IBM Cloud - https://cloud.ibm.com/docs

UNIT-NO	CONTENT	CONTACT HRS	LEARNING OUTCOME	KNOWLEDGE LEVELS	REFERENCE/ LEARNING MATERIALS
I	**Introduction to AI:** Overview of Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) History and Evolution of AI Basic Concepts and Terminology	**8**	**C1**	**Remembering Understanding**	**T1/R1**
II	**Advanced Python and Data Visualization:** Introduction to NumPy: Arrays, Indexing, and Operations, Data Manipulation with Pandas: DataFrames, Indexing, Filtering, and Grouping, Data Visualization with Matplotlib: Plotting, Subplots, Customization	**8**	**C2**	**Analysing, Applying**	**T2/R1/R2**

III	**Introduction to Deep Learning and ML Algorithms:** Introduction to Deep Learning: Neural Networks, Layers, and Activation Functions, Deep Learning Algorithm: ANN, CNN, RNN with Tensorflow and Keras, Supervised Learning Algorithms: Linear Regression, Logistic Regression, Decision Trees Unsupervised Learning Algorithms: K-Means Clustering, Model Evaluation and Hyperparameter Tuning	**8**	**C3**	**Remembering, Analysing**	**T2/R2**
IV	**IBM Natural Language Processing (NLP):** Introduction to NLP: learn the basics of natural language processing, and IBM's NLP services, such as Watson Natural Language Understanding (NLU) and Watson Discovery, for extracting insights from unstructured text data and also use Postman or Nodered for deploying NLP Pipeline.	**8**	**C4**	**Understanding, Applying**	**T3/R2/R3s**

V	**IBM Watson Service:** Introduction to IBM Watson: explore Watson Studio, IBM's integrated environment for data scientists, developers, and domain experts, to build and deploy ML models. Creating Bots using Watson Assistant: learn the basics of Watson Assistant, including intents, entities, dialog flows, and integration capabilities.	**8**	**C5**	**Applying**	**T4/R3**

SEMESTER – V									
Course Title	**SUMMER INTERNSHIP**								
Course code	**24BCAM3207R**	**Total credits: 1** **Total hours: 40**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**8**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives (Minimum 3)	1. Apply academic knowledge and develop industry-specific skills in a professional setting. 2. Gain insight into career pathways and enhance problem-solving, communication, and teamwork skills. 3. Build a professional network and understand workplace dynamics.								
CO1	Demonstrate the application of theoretical knowledge in a practical environment.								
CO2	Identify and develop key professional skills relevant to the field of study.								
CO3	Reflect on personal and professional growth throughout the internship.								
CO4	Produce professional documents, such as a resume, cover letter, and portfolio.								
CO5	Present the internship experience and learning outcomes effectively.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Orientation and Preparation:** Introduction to the internship, setting goals, understanding workplace expectations, and professional behaviour.	**8**	Demonstrate the application of theoretical knowledge in a practical environment.	1, 2
II	**Skills Development:** Developing industry-specific skills, working on assigned projects, and regular feedback sessions.	**8**	Identify and develop key professional skills relevant to the field of study.	2, 3
III	**Professional Growth:** Understanding workplace dynamics, building a professional network, and learning from mentors and colleagues.	**8**	Reflect on their personal and professional growth throughout the internship.	2, 3
IV	**Professional Documentation**: Creating and refining professional documents such as resumes, cover letters, and portfolios.	**8**	Produce professional documents, such as a resume, cover letter, and portfolio.	3, 4
V	**Presentation and Reflection:** Presenting the internship experience, discussing learning outcomes, and reflecting on personal growth and future career plans.	**8**	Present their internship experience and learning outcomes effectively.	3, 4, 5

TEXT BOOKS:
REFERENCE BOOKS:
OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Demonstrate the application of theoretical knowledge in a practical environment.	1, 2
2	Identify and develop key professional skills relevant to their field of study.	2, 3, 7
3	Reflect on their personal and professional growth throughout the internship.	8
4	Produce professional documents, such as a resume, cover letter, and portfolio.	6
5	Present their internship experience and learning outcomes effectively.	6

SEMESTER – VI									
Course Title	**COMPUTER VISION**								
Course code	**24BCAM3201R**	**Total credits: 3** **Total hours: 30L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**1**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Programming Skills, Mathematics	**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application**								
Semester	**VI**								
Course Objectives (Minimum 3)	1. Introduce the fundamentals of computer vision and its applications. 2. Equip students with the skills to acquire, pre-process, and analyse image and video data. 3. Explore various techniques for image feature extraction, segmentation, and object recognition. 4. Understand the principles of motion analysis and tracking. 5. Gain practical experience through programming assignments and projects.								
CO1	Explain the essential concepts of computer vision, including its applications, challenges, and limitations, drawing comparisons with human vision.								
CO2	Classify the different image feature extraction techniques (edge detection, corner/blob detection, texture analysis, shape descriptors) to extract relevant information from images.								
CO3	Develop various image segmentation algorithms (thresholding, region-based, edge-based, morphological) to partition images into meaningful regions for further analysis.								
CO4	Justify supervised and unsupervised learning methods (k-NN, SVM, neural networks, K-means, PCA) for object recognition and classification tasks in images.								
CO5	Examine the diverse applications of computer vision in various domains like image retrieval, medical imaging, robotics, and security systems, highlighting its potential impact.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Computer Vision:** Definition, history, and applications of computer vision, the human visual system vs. computer vision Images in the Spatial Domain: Pixels and Windows	**6**	**Describe and explain Computer vision and Spatial Domain of Images**	**1, 2**
II	Image Formation: 2D transformations, 3D transformations, 3D rotations, 3D to 2D projections, Photometric image formation, The digital camera, Sampling and aliasing, Compression.	**8**	**Describe, Illustrate and Analyse the Image Formation and their different form**	**1,2,4**
III	Image processing: Point operators, Linear filtering, Fourier transforms, Pyramids and wavelets, Parametric transformations, Mesh-based warping, Application: Feature-based morphing	**8**	**Explain and Applying Image processing like Fourier transforms, Parametric transform etc**	**2, 3**
IV	Feature detection and matching: Points and patches, Performance-driven animation, Edge detection, Edge linking, Vanishing points Image **Segmentation,Snakes** Image classification, Application: Visual similarity search	**8**	**Describe and Applying Feature detection and matching in different images**	**2, 3**
V	Applications of Computer Vision: Image and video retrieval: searching for specific images or videos in a database, Medical imaging: diagnosis and analysis of medical images, Surveillance and security: object tracking and anomaly detection	**6**	**Describe and Applying Various Computer Vision techniques like Searching, Medical images diagnosis, etc**	**2, 3**
Practical	1. **Image Loading and Display:** Learn how to read and display images using OpenCV. 2. **Image Grayscale Conversion:**	**30**	**Describe, illustrate and explain and apply different imaging techniques and carry out practical in lab.**	**1,2,3,4**

	Understand the process of converting color images to grayscale. 3. **Image Thresholding:** Explore thresholding techniques for image binarization. 4. **Image Smoothing and Blurring:** Implement various filters for smoothing and blurring images. 5. **Edge Detection:** Detect edges in images using Canny edge detection and other methods. 6. **Image Morphological Operations:** Apply morphological operations like erosion and dilation for image processing. 7. **Image Geometric Transformations:** Perform scaling, rotation, and other geometric transformations on images. 8. **Contour Detection and Object Recognition:** Find contours in images and use them for object recognition. 9. **Histogram Analysis and Equalization:** Analyze image histograms and perform histogram equalization for image enhancement. 10. **Template Matching:** Find occurrences of a template image within a larger image. 11. **Feature Detection and Keypoints:** Detect keypoints like corners and Harris corners in images. 12. **Feature Matching and Homography:** Match features between images and estimate homography for image registration. 13. **Image Pixel Manipulation and Display with OpenCV**. 14. **Python program to draw various shapes.** 15. **Python program to draw a simple bullseye with the cv2.circle function.** 16. **Program to perform various transformation in an image**. 17. **Python program that finds**			

	and draws contours of objects in an image. 18. Detecting Blue Objects. 19. **Program for detecting face.** 20. Program for Face and Smile Detection. 21. Image segmentation based on color 22. Image Color Conversion: Convert the color space of the image (e.g., RGB to grayscale, HSV, LAB, etc.) and analyze the channels separately.			

Text Books:

1. Szeliski, Richard. *Computer Vision: Algorithms and Applications*. Springer Nature, 2022.
2. Davies, E. Roy. *Computer and Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, 2012.
3. Gonzalez, Rafael C. *Digital Image Processing*. Pearson Education India, 2009.

Reference Books:

1. Forsyth, David A., and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
2. Brownlee, Jason. *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, And Data Transforms In Python*. Machine Learning Mastery, 2020.

Additional Resources:

- OpenCV Tutorials and Documentation: https://opencv.org/
- Python libraries for computer vision: Scikit-image, TensorFlow, PyTorch
- https://nptel.ac.in/courses/117105079
- https://nptel.ac.in/courses/106105216/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Explain the essential concepts of computer vision, including its applications, challenges, and limitations, drawing comparisons with human vision.	1, 2, 7 and 8
2	Classify the different image feature extraction techniques (edge detection, corner/blob detection, texture analysis, shape descriptors) to extract relevant information from images.	1, 2, 7 and 8
3	Develop various image segmentation algorithms (thresholding, region-based, edge-based, morphological) to partition images into meaningful regions for further analysis.	1, 2, 7 and 8
4	Justify supervised and unsupervised learning methods (k-NN, SVM, neural networks, K-means, PCA) for object recognition and classification tasks in images.	1, 2, 7 and 8
5	Examine the diverse applications of computer vision in various domains like image retrieval, medical imaging, robotics, and security systems, highlighting its potential impact.	1, 2, 7 and 8

SEMESTER – VI									
Course Title	**Introduction to Cybersecurity**								
Course code	**24BCAM3202R**	**Total credits: 4 Total hours: 45L**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite		**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives (Minimum 3)	1. To understand the foundational concepts of cyberspace, including the architecture of the internet, web technologies, and the regulatory frameworks governing cyberspace.. 2. To identify and classify various types of cybercrimes, comprehend cybercriminals' methodologies, and analyze the legal aspects of cybercrimes under the IT Act 2000 and its amendments. 3. To explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption algorithms, and demonstrate knowledge of Public Key Infrastructure (PKI) and digital certificates. 4. To describe the components and security challenges of e-commerce, including digital payments methods like UPI, e-wallets, and Aadhar-enabled payments, while adhering to RBI guidelines and Payment Settlement Act provisions. 5. To implement cybersecurity risk management practices, including risk assessment methodologies, security policies, incident response planning, and the application of security tools and technologies for safeguarding digital devices and networks.								
CO1	Understand the foundational concepts of cyberspace, including the architecture of the internet, communication protocols, and governance structures.								
CO2	Identify and classify various types of cybercrimes, their modus operandi, and the legal frameworks governing cybercrimes in India.								
CO3	Explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption, and understand the role of PKI and digital certificates.								
CO4	Analyze the security challenges and best practices in e-commerce, including threats, digital payment mechanisms, and regulatory guidelines.								
CO5	Implement cybersecurity risk management strategies, including risk assessment methodologies, incident response planning, and compliance with security policies and procedures.								
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**						**KL**
I	**Introduction to Cybersecurity:** Defining Cyberspace and Overview of Computer and Web-technology, Architecture of cyberspace, Communication and web technology, Internet, World wide web, Advent of internet, Internet infrastructure for data transfer and governance, Internet society, Regulation of cyberspace, Concept of cyber security, Issues and challenges of cyber security.		After completion of this module, students would be able to understand the concept of Cyber security and issues and challenges associated with it.						1,2,3
II	**Cybercrime and Cyber law:** Classification of cybercrimes, Common cybercrimes-cybercrime targeting computers and mobiles, cybercrime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Cybercriminals modus-operandi , Reporting of cybercrimes, Remedial and mitigation		Students, at the end of this module, should be able to understand the cybercrimes, their nature, legal remedies and as to how report the crimes through available platforms and procedures.						2,3,4

	measures, Legal perspective of cybercrime, IT Act 2000 and its amendments, Cybercrime and offences, Organisations dealing with Cybercrime and Cyber security in India, Case studies.			
III	**Cryptography and Encryption:** Fundamentals of cryptography and encryption algorithms Symmetric vs. asymmetric encryption Public key infrastructure (PKI) and digital certificates.		Explain network security principles, assess vulnerabilities, and implement measures to secure networks against external and internal threats.	2,3,4
IV	**E-Commerce and Digital Payments:** Definition of E-Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, E-Commerce security best practices, Introduction to digital payments, Components of digital payment and stake holders, Modes of digital payments- Banking Cards, Unified Payment Interface (UPI), e-Wallets, Unstructured Supplementary Service Data (USSD), Aadhar enabled payments, Digital payments related common frauds and preventive measures. RBI guidelines on digital payments and customer protection in unauthorised banking transactions. Relevant provisions of Payament Settlement Act,2007,		After the completion of this module, students would be able to understand the basic concepts related to E-Commerce and digital payments. They will become familiar with various digital payment modes and related cyber security aspects, RBI guidelines and preventive measures against digital payment frauds.	2,3,4
V	**Cybersecurity Risk Management, Digital Devices Security, Tools and Technologies for Cyber Security:** Risk assessment methodologies and frameworks Security policies, procedures, and compliance Incident response and disaster recovery planning; End Point device and Mobile phone security, Password policy, Security patch management, Data backup, Downloading and management of third-party software, Device security policy, Cyber Security best practices, Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus, Wi-Fi security, Configuration of basic security policy and permissions.		Discuss cybersecurity governance frameworks, analyze regulatory requirements, and develop policies to align with organizational goals and ensure compliance. Understand the basic security aspects related to Computer and Mobiles. They will be able to use basic tools and technologies to protect their devices.	1,2,3

Text Books:

T1: Introduction to Computer Security" by Michael T. Goodrich and Roberto Tamassia
T2: "Cybersecurity: A Practical Guide to the Law of Cyber Risk" by Andrew Serwin

T3: Network Security Essentials: Applications and Standards" by William Stallings

Reference Books:
R1: "Cryptography and Network Security: Principles and Practice" by William Stallings
R2:"Security Engineering: A Guide to Building Dependable Distributed Systems" by Ross Anderson
R3: "Principles of Computer Security: CompTIA Security+ and Beyond" by Wm. Arthur Conklin, Greg White, Chuck Cothren, Roger L. Davis, Dwayne Williams

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the foundational concepts of cyberspace, including the architecture of the internet, communication protocols, and governance structures.	1,2,3,4,5,6,7, and 8
2	Identify and classify various types of cybercrimes, their modus operandi, and the legal frameworks governing cybercrimes in India.	1,2,3,4,5,6,7, and 8
3	Explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption, and understand the role of PKI and digital certificates.	1,2,3,4,5,6,7, and 8
4	Analyze the security challenges and best practices in e-commerce, including threats, digital payment mechanisms, and regulatory guidelines.	1,2,3,4,5,6,7, and 8
5	Implement cybersecurity risk management strategies, including risk assessment methodologies, incident response planning, and compliance with security policies and procedures.	1,2,3,4,5,6,7, and 8

SEMESTER – VI									
Course Title	**Project II**								
Course code	**24BCAM3205R**	**Total credits: 6**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:	**0**	**0**	**0**	**0**	**36**	**0**	**6**
Pre-requisite		**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives (Minimum 3)	1. To develop the practical skill 2. To enhance Critical Thinking and Problem-Solving Abilities 3. To Improve Communication and Collaboration Skills								
CO1	Demonstrate a sound technical knowledge of the selected project topic.								
CO2	Illustrate problem identification, formulation, and solution.								
CO3	Plan solutions to complex problems utilizing a systems approach.								
CO4	Design the prototype of the solution and analyze the prototype using testing methods.								
CO5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Project Introduction and Planning** Overview of project goals and objectives. Importance of project planning and management. Project Proposal Development: Identifying a project topic or problem Conducting preliminary research Writing a project proposal Setting timelines and milestones Resource allocation (materials, tools, etc.) Risk assessment and management strategies		**Describe, illustrate and explain problem**	**2,3**
II	Literature Review and Research Methodology Identifying and reviewing relevant literature Summarizing and synthesizing existing research Highlighting gaps and formulating research questions Choosing appropriate research methods (qualitative, quantitative, or mixed methods) Designing research instruments (surveys, interviews, experiments, etc.) Ethical considerations in research		**Describe, illustrate and Literature Review**	**2,3,4**
III	Data Collection and Analysis Gathering primary data (surveys, experiments, interviews, observations) Ensuring data accuracy and reliability Storing and organizing data		**Describe, illustrate and explain data collection and analysis**	**2,3,4**

	Using statistical tools and software for data analysis Interpreting data and drawing conclusions Visualizing data (charts, graphs, tables)			
IV	Project Implementation and Monitoring Project Execution: Implementing the project plan Monitoring progress and making adjustments as needed Documentation and Record Keeping: Keeping detailed records of project activities and outcomes Troubleshooting and problem-solving during implementation Interim Reporting: Preparing and presenting progress reports Receiving and incorporating feedback		**Describe, illustrate and explain project implementation and execution**	**6**
V	Project Presentation and Evaluation Final Reporting: Writing the final project report (structure, content, formatting) Ensuring clarity, coherence, and completeness Project Presentation: Preparing visual aids (slides, posters, etc.) Practicing presentation skills (public speaking, answering questions) Evaluation and Reflection: Assessing project outcomes against objectives Reflecting on the project process and personal learning Identifying areas for future improvement and potential follow-up projects		**Describe, illustrate and explain testing and documentation**	**2,3,5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Demonstrate a sound technical knowledge of selected project topic.	1, 2 and 4
2	Illustrate problem identification, formulation and solution.	1, and 2
3	Plan solutions to complex problems utilizing a systems approach.	1, 2, 3, 5 and 7
4	Design the prototype of the solution and analyze the prototype using testing methods.	1, 2, 3, 5, 6, 7, and 8
5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.	1, 2, 3, 6 and 8

SEMESTER – VI									
Course Title	**PE I: FUNDAMENTALS OF AI /ML**								
Course code	**24BCAM3203**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45 L	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter/ I semester of the second year of the program**								
Course Objectives (Minimum 3)	1. Learn the basic principles of machine learning and how they can be applied to different scenarios. 2. Understand the differences between supervised and unsupervised learning algorithms. 3. Understand the fundamental concepts and terminology of Artificial Intelligence (AI)								
CO1	Explain machine learning concepts, applications, challenges, and basic data descriptions								
CO2	Apply and evaluate clustering techniques using various methods.								
CO3	Differentiate prediction and classification, apply algorithms, and evaluate performance.								
CO4	Implement and compare various search strategies and algorithms.								
CO5	Use logic and probabilistic models for knowledge representation and reasoning.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Machine Learning:** What is machine learning, applications of machine learning, issues and challenges faced in machine learning, types of data attributes – nominal, binary, ordinal, numeric, discrete and continuous, basic statistical descriptions of data – measures of central tendency and dispersion, major tasks in data pre-processing	**8**	Define machine learning, list its applications, identify common issues, distinguish between types of data attributes, describe basic statistical data measures, and outline key data pre-processing tasks.	1, 2, 4
II	**Unsupervised Learning - Data Clustering:** what is cluster analysis, need for cluster analysis, data matrix, dissimilarity matrix, proximity and dissimilarity measures for different data attribute types, partitioning based clustering methods, density-based clustering methods and hierarchical clustering methods, measuring cluster quality	**8**	Define cluster analysis, explain its necessity, describe data and dissimilarity matrices, discuss proximity and dissimilarity measures, compare partitioning, density-based, and hierarchical clustering methods, and evaluate cluster quality.	1, 2, 3, 4
III	**Supervised Learning – Prediction:** What is prediction, application areas of prediction, Simple linear regression, Multiple linear regression, Predictor Error measures **Classification**: What is classification, differences between classification and prediction, applications of classification, Some classification algorithms, Confusion matrix and metrics for evaluating classifier performance	**8**	Define prediction and classification, identify their applications, explain the differences between them, describe linear regression models and prediction error measures, list classification algorithms, and evaluate classifier performance using confusion matrices and metrics.	1, 2, 3, 4
IV	**Introduction to Search:** Searching for solutions, Uniformed search strategies, Informed search strategies, Local search algorithms and optimistic problems, Adversarial Search, Search for games, Alpha – Beta pruning.	**8**	Understand various search strategies, including uninformed and informed methods, explore local search algorithms and adversarial search techniques, and apply alpha-beta pruning in game search scenarios.	2, 3, 4
V	**Knowledge Representation & Reasoning:** Propositional logic, Theory of first order logic, Inference in First order logic,	**8**	Master propositional and first-order logic, implement inference mechanisms like forward and backward	3, 4, 5

	Forward & Backward chaining, Resolution, Probabilistic reasoning, Utility theory, Hidden Markov Models (HMM), Bayesian Networks.		chaining and resolution, understand probabilistic reasoning including Hidden Markov Models and Bayesian Networks, and apply concepts of utility theory.	

TEXT BOOKS:

T1: Jiawei Han and MichelineKamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, India

T2: Russell, S., &Norvig, P. Artificial intelligence: a modern approach. Third Edition. Pearson new international edition. 2014

REFERENCE BOOKS:

R1: Han, Manilla and Smyth, Principles of Data Mining, PHI, India

R2: Tan, Steinbach, Vipin Kumar, Introduction to Data Mining, Pearson Education Inc.

R3: E. Rich and K. Knight, Artificial Intelligence, Tata McGrawHill.

R4: N. J. Nilsson, Principles of Artificial Intelligence, Narosa

R5: D. W. Patterson, Introduction to Artificial Intelligence and Expert Systems, Prentice Hall of India.

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Explain machine learning concepts, applications, challenges, and basic data descriptions	1, 2, 3
2	Apply and evaluate clustering techniques using various methods.	2, 3, 4
3	Differentiate prediction and classification, apply algorithms, and evaluate performance.	2, 3, 6
4	Implement and compare various search strategies and algorithms.	2, 3, 7
5	Use logic and probabilistic models for knowledge representation and reasoning.	3, 4, 5

SEMESTER – VI									
Course Title	**PE I: BUSINESS INTELLIGENCE**								
Course code	**24BCAM3204R**	**TOTAL CREDITS: 3 TOTAL HOURS: 40**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter/ I semester of the third year of the program**								
Course Objectives (Minimum 3)	1. To learn how managers utilise business analytics to create, address, and assist management decision-making for company challenges. 2. To become accustomed to the procedures required to create, present, and analyse business data. 3. Get knowledge on how to use Excel and its add-ons to solve business challenges								
CO1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.								
CO2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.								
CO3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.								
CO4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods								
CO5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Overview of Business Analytics**: Introduction to Analytics, The Challenge of Decision Making, What Is Business Intelligence, The Business Intelligence Value Proposition, The Combination of Business and Technology	**8**	Define and understand the fundamentals of business analytics, the decision-making challenges, the concept and value of business intelligence, and the synergistic relationship between business and technology.	1,2
II	**Introduction to Data Visualization:** Visualization/ Data Issues, Organization/sources of data, Importance of data quality, Dealing with missing or incomplete data, Data Classification.	**8**	Master the principles of data visualization, including data organization, the significance of data quality, strategies for handling missing data, and data classification techniques.	2, 3
III	**Descriptive Analytics**: Data Visualization and Analytics- Charts (Bars-Pie-Line-Scatter-Map-Bubble-Box & Whisker-Tree map - Heat map-Circle and Area) -Worksheet, Dashboard and Story Board creation	**8**	Create and interpret diverse charts and visualizations such as bar, pie, line, scatter, map, bubble, box & whisker, tree map, heat map, and area charts, and develop comprehensive worksheets, dashboards, and storyboards.	3, 6
IV	**Predictive Analytics**: Linear Regression, Multi-linear Regression and Time Series Forecasting. **Prescriptive Analytics**: Linear optimization, Integer optimization, Non-linear programming, Optimization of Network models and Monte Carlo Simulation	**8**	Apply linear and multi-linear regression models and time series forecasting for predictive analytics, and utilize optimization techniques	3, 4
V	**Applications:** Practical demonstration using R or Python or SPSS by industry experts on Emerging trends in business analytics and intelligence	**8**	Demonstrate proficiency in using software tools like R, Python, or SPSS to implement advanced analytics techniques	3, 5, 6

TEXT BOOKS:

T1: Sharda R, Delen D, Turban E, Aronson J, Liang T. P, (2014), Business Intelligence and Analytics: Systems for Decision Support, 10th edition, Pearson Education.
T2: Powell S. G, Barker K. R, (2014), Management Science: The Art of Modeling With Spreadsheets, (W/Cd), 4th edition, John Wiley & Sons.

REFERENCE BOOKS:

R1: Linoff G. S, Berry M. J, (2011), Data mining techniques: for marketing, sales, and customer relationship management, 3rd edition, John Wiley & Sons.
R2. Frank B, Green B, Harris T, Van De Vanter K, (2010), Business Intelligence Strategy: A Practical Guide for Achieving BI Excellence, MC Press.
R3. Hair, J. F, Black W. C, Babin B. J, Anderson R. E, Tatham R. L, (2009), Multivariate data analysis, 7th edition, Pearson education.

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.	2, 3, 4
2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.	2, 3
3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.	2, 3, 6
4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods	2, 3, 7
5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.	2, 3, 7, 8

SEMESTER – VI									
Course Title	**PE II: DATA ANALYTICS USING PYTHON**								
Course code	**24BCAM3205 R/**	**TOTAL CREDITS: 4 TOTAL HOURS: 45L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives (Minimum 3)	1. Equip students with essential skills in algebra and statistical analysis fundamental to data science. 2. Teach students to effectively manipulate data using Python libraries like NumPy and Pandas and create comprehensive visualizations. 3. Enable students to implement unsupervised learning methods and engage with the latest data science advancements through practical applications and case studies.								
CO1	Understand and apply concepts of functions, exponentials, logarithms, polynomials, alternate coordinate systems, and statistical distributions								
CO2	Demonstrate proficiency in using NumPy for various operations and broadcasting, and manipulate data frames in Pandas, including loading data in different formats								
CO3	Create and interpret various types of data visualizations such as histograms, box plots, bar plots, pie charts, and line charts								
CO4	Implement and evaluate unsupervised learning techniques in Python, including K-Means clustering, hierarchical clustering, and Principal Component Analysis (PCA)								
CO5	Analyze the latest improvements and applications in data science through case studies and exploratory data analysis (EDA) in interdisciplinary research areas.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Intermediate Algebra& Linear Algebra: Functions, Exponentials and Logarithm's, Polynomial's, Alternate Coordinate systems, Binomial Distribution, Poisson distribution and Normal distribution its properties, Assumption of ANOVA, Measures of Central Tendency in Data.	**8**	Demonstrate understanding and application of intermediate algebra and linear algebra concepts	1,2
II	**Introduction about NumPy:** Different NumPy Operations, Broadcasting with NumPy, Introduction about Pandas, Reading or Loading data into Data frame, Pandas Data Frame Manipulations, Data Loading /Reading in different formats (CSV, Excel, Json, HTML)	**8**	Master the use of NumPy for performing array operations, broadcasting techniques, and Pandas for data manipulation, including reading, loading, and transforming data within data frames across various formats	2, 3
III	**Introduction to data Visualizations:** Principles Behind Data Visualizations, Histograms Visualize, Box plots-Visualize, the Distribution of Continuous Numerical Variables (Bar Plots Pie Chart Line Chart). Data Visualization using R- Line Plots and Regression.	**8**	Develop the ability to create and interpret various data visualizations, including histograms, box plots, bar plots, pie charts, line charts, and utilize R for generating line plots and conducting regression analysis to effectively communicate data insights.	3, 4
IV	**Unsupervised Learning in Python:** K- Means Theory/ Implementation, Quantifying K-Means Clustering Performance, Hierarchical Clustering Theory, Principal Component Analysis (PCA) theory / Implementation. Selection criteria for	**8**	Apply unsupervised learning techniques in Python, including K-Means clustering and hierarchical clustering, evaluate their performance, implement Principal Component	3, 4

	number of clusters choosing.		Analysis (PCA), and determine the appropriate number of clusters based on selection criteria.	
V	The improvements and most recent developments in the course's topics as well as their most recent applications in the field Data Science. Case Study and EDA in the interdisciplinary research areas of Data Science.	**8**	Analyze recent advancements and applications in data science, applying knowledge through case studies and exploratory data analysis (EDA) in various interdisciplinary research areas to understand current trends and innovations.	4, 5

TEXT BOOKS:

T1: Think Python, Allen B. Downy, O'reilly

T2: Python for Everybody: Exploring Data Using Python 3 Book by Charles Severance

REFERENCE BOOKS:

R1: Python Data Science Handbook, Jakes Vander Plas O' Reilly

R2: Eric Matthes, Python Crash Course, A Hands – on Project Based Introduction to Programming, 2nd Edition, No Starch Press, 2019

OTHER LEARNING RESOURCES:

https://www.w3schools.com/python/python_intro.asp

https://www.tutorialspoint.com/python/index.htm

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand and apply concepts of functions, exponentials, logarithms, polynomials, alternate coordinate systems, and statistical distributions	1, 2, 3
2	Demonstrate proficiency in using NumPy for various operations and broadcasting, and manipulate data frames in Pandas, including loading data in different formats	3, 6, 7
3	Create and interpret various types of data visualizations such as histograms, box plots, bar plots, pie charts, and line charts	3, 6, 8
4	Implement and evaluate unsupervised learning techniques in Python, including K-Means clustering, hierarchical clustering, and Principal Component Analysis (PCA)	2, 3, 7
5	Analyze the latest improvements and applications in data science through case studies and exploratory data analysis (EDA) in interdisciplinary research areas.	3, 7, 8

ASSAM DOWN TOWN UNIVERSITY

Curriculum and Syllabus

Bachelor of Computer Application

OUTCOME BASED EDUCATION FRAMEWORK

CHOICE BASED CREDIT SYSTEM

Version: 2.2

FACULTY OF COMPUTER TECHNOLOGY

July, 2024

PREAMBLE

Assam down town University is a premier higher educational institution which offers Bachelor, Master and Ph.D. degree programmes across various faculties. These programmes, collectively embodies the vision and mission of the university. In keeping with the vision of evolutionary changes taking place in the educational landscape of the country, the university has restructured the course curriculum as per the guidelines of National Education Policy 2020. This document contains outline of teaching and learning framework and complete detailing of the courses. This document is a guidebook for the students to choose desired courses for completing the programme and to be eligible for the degree. This volume also includes the prescribed literature, study materials, texts and reference books under different courses as guidance for the students to follow.

Recommended by the Board of Studies (BOS) meeting of the Faculty of Computer Technology held on dated 15/07/2024 and approved by the 51st Academic Council (AC) meeting held on dated 26/07/2024.

Chairperson, Board of Studies

Member Secretary, Academic Council

Vision

To become a Globally Recognized University from North Eastern Region of India, dedicated to the Holistic Development of Students and Making Society Better

Missions

1. Creation of curricula that address the local, regional, national, and international needs of graduates, providing them with diverse and well-rounded education.

2. Build a diverse student body from various socio-economic backgrounds, provide exceptional value-based education, and foster holistic personal development, strong academic careers, and confidence.

3. Achieve high placement success by offering students skill-based, innovative education and strong industry connections.

4. Become the premier destination of young people, desirous of becoming future professional leaders through multi disciplinary learning and serving society better.

5. Create a highly inspiring intellectual environment for exceptional learners, empowering them to aspire to join internationally acclaimed institutions and contribute to global efforts in addressing critical issues, such as sustainable development, Climate mitigation and fostering conflict-free global society.

6. To be renowned for creating new knowledge through high quality inter disciplinary research for betterment of society.

7. Become a key hub for the growth and excellence of AdtU's stake holders including educators, researchers and innovators

8. Adapt to the evolving needs and changing realities of our students and community by incorporating national and global perspectives, while ensuring our actions are in harmony with our foundational values and objectives of serving the community.

Programme Details

Programme Overview

Understanding the fundamentals of computer programming languages and databases is of utmost importance in the field of computer science and information technology. The Bachelor of Computer Application (BCA) Programme has been designed in sync with the latest industry demands. This programme enriches the students with the necessary skills to build a successful career in the Information Technology sector. Through this programme, we intend to create a skilled workforce to take up future challenges in the industry. It aims at educating the students as expert programmers and computer professionals for the future.

I. Specific Features of the Curriculum

The Bachelor of Computer Application (BCA) programme is meticulously designed to align with current industry demands. It focuses on imparting a strong foundation in programming languages and database management. Students gain practical skills essential for the IT sector, preparing them for successful careers. The curriculum emphasizes hands-on experience, ensuring students are ready to tackle real-world challenges. By integrating the latest technological trends, the programme ensures relevance in a rapidly evolving field. It aims to produce expert programmers and computer professionals. Graduates are equipped to meet future industry challenges. Continuous updates keep the curriculum effective. The programme builds a skilled workforce for the future.

II. Eligibility Criteria:

Students should be secured 45 % in 10+2 in Arts/Science/Commerce from AHSEC or any other equivalent Examination recognized as such by the University with preferably Mathematics as one of the subjects in HS (Pass Marks).

Note: Candidate having (10+2) without mathematics/computer Science has to undertake additional bridge courses.

III. Program Educational Objectives (PEO):

PEO1: To produce graduates who have a strong foundation of knowledge and skills in the field of Computer Applications.

PEO2: To produce graduates who can provide solutions to challenging problems in their profession by applying Computer Science theory and practices.

PEO3: To produce graduates who are employable in industries/public sector/Govt. organizations or work as an entrepreneur, as well as can provide leadership and are effective in a multidisciplinary environment

IV. Program Specific Outcomes (PSO):

PSO 1: Advance the graduates with the contemporary trends in industrial/Computer Application environments and also will be capable of innovating novel solutions to prevailing problems by applying software engineering techniques and strategies.

PSO 2: Develop a holistic comprehension of Computer Science and management principles required for the application of sustainable technologies for societal development, and also will be able to communicate effectively in oral, written, visual, and graphic modes as a member and leader in a team, to manage projects in multidisciplinary environments.

PSO 3: Prepare the graduates for the state, national, and international competitive examinations with focused and updated syllabi.

V. Program Outcome (PO):

PO1 Basic Mathematical Knowledge: Apply knowledge of Mathematics & Statistics to the solution of ICT problems.

PO2 Problem Analysis, Design/Development of Solutions: Analysis the IT problem and then design/develop the solutions using Software Engineering principles that meet specified needs with appropriate consideration for cultural, societal, and environmental considerations.

PO3 Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern ICT tools including prediction and modelling with an understanding of the limitations.

PO4 Environment and Sustainability: Understand the impact of professional IT solutions in societal and environmental contexts and demonstrate knowledge of and need for sustainable development.

PO5 Ethics: Apply ethical principles and commit to professional ethics, responsibilities, and norms.

PO6 Communication: Communicate effectively with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO7 Project Management and Finance: Demonstrate knowledge and understanding of Software Engineering and Project management principles and apply these to one's work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO8 Life-long Learning: Recognize the need for and have the preparation and ability to Engage in independent and life-long learning in the broadest context of technological Change.

Total Credits to be Earned: 141

VI. Career Prospects:

After completing a degree in one of the most desirable professions in the current times, Computer Application graduates have a wide array of options at their disposal. There is no one single career option but several after Computer Application -

a. Higher Studies in Technical Field: From there one could go into the research field or complete PhD and work as a Faculty. In order to pursue M. Tech. in India, one will need to clear the GATE Examination, which will open a lot of opportunities to study Master's as well as work in PSUs. From here you can work in Government owned companies like DRDO, BARC, CSIR, ISRO, AAI, ONGC etc.
b. Technical Job: Graduates can work at a good company in various fields like Data science and Analysis, Machine Learning Engineer, Software Developer, Blockchain Developer and Engineer, Computer Network Architect, Database Administration and Management, Security Analyst, Game Development, Digital Marketing, etc.
c. Government Services: Along with pursuing technical government jobs after appearing for GATE, engineering graduates could also work in non technical fields by applying and studying for various government exams like UPSC, SSC, Banking exams, Defence or IAS. These are very rewarding jobs as they are well paid, and very secure.

Entrepreneurship: If someone is filled with ideas with a need to create a change in the society, and like being one's own boss, one can even commence own start-up. For an entrepreneur, "opportunity" equates to something that solves a problem and addresses a need in a way that the entrepreneurs can profit from or further a cause.

EVALUATION METHODS

The student performance shall be evaluated through In-semester (Sessional) and semester-end examinations. A weightage of 40% or as prescribed by the programme shall be added to the score of the end semester examination.

A. INTERNAL ASSESSMENT:

The teacher who offers the course shall be responsible for internal assessment by conducting in-semester (sessional) examination and evaluating the performance of the students pursuing that course. The components for internal assessment are illustrated in the table given below.

SN	Components/ Examinations	Marks Allotted
1.	In-Sem Exam – I (ISE-I) (Written Examination)*	30
2.	In-Sem Exam – II (ISE-II) (Written Examination)*	30
3.	Assignment	10
4.	Presentation (SP)	10
5.	Quiz	5
6.	Class Performance based score*	5

are compulsory

Note: *Total Internal assessment should be out of 40*

INSTRUCTION

1. If a student fails to appear in the any of the component without any valid reason he/she shall be marked zero in that component. However, the course teacher at his discretion may arrange for the missed test on an alternate date for the absentee students after determining ground with genuine/valid reasons for the absent.
2. The report of evaluation of an activity towards the in-semester (sessional) component of a course shall be duly notified by the concerned course teacher within a week of completion.
3. The program coordinators should upload the in-semester marks to the ERP and forward acknowledgement of all the courses of the program to the Controller of Examinations before the start of the End-semester examination.

B. SEMESTER END EXAMINATION:

Time table for end semester examination is published at least 25 days prior to the start of Examination.

I. Pre-Examination:

Eligibility Criteria for a student to appear in University Examinations:

The student shall only be allowed to appear in a University Examination, if:

i) He/ She is a registered student of the University;
ii) He/ She is of good conduct and character;
iii) He/ She has completed the prescribed Programme of study with minimum percentage of attendance as laid down in the Regulations of the Programme concerned.

Under special cases, a student may be allowed to appear for an examination without being registered in the University but the result of the said student will be kept on hold till the registration of the concerned student is completed.

II. Admit Card:

Admit card for the examination may be downloaded through ERP where the system will generate a Unique ID Cards through online.

The University shall have the right to cancel admission for examination of any candidate on valid grounds.

III. Pattern of Question Papers:

The question paper shall follow the principles of Bloom's Taxonomy.

Table

S. N.	Level	Questions /verbs for test
1	Remember	List, Define, tell, describe, recite, recall, identify, show who, when, where, etc.
2	Understand	Describe, explain, contrast, summarize, differentiate, discuss etc.
3	Apply	Predict, apply, solve, illustrate, determine, examine, modify
4	Analyze	Classify, outline, categorize, analyze, diagrams, illustrate, infer, etc.
5	Evaluate	Assess, summarize, choose, evaluate, recommend, justify, compare etc.
6	Create	Design, Formulate, Modify, Develop, integrate, etc.

Note: No course is to be evaluated on the basis of **all 6 knowledge levels.**

The format of the question paper across all the program follow a unique pattern and the total marks is 60

Table 1: Question paper pattern for End semester examination

Sl no	Question pattern	Total marks
1	MCQs (10 Questions)	10
2	2 Marks questions (10 Questions)	20
3	4 Marks questions (5 Questions)	20
4	10 Marks questions (1 Question)	10

IV. Examination Duration:

Each paper of 60 marks shall ordinarily be of two hours duration.

V. Practical Examinations, Viva-Voice etc.:

i) Practical examination shall be conducted in the presence of one external expert and one or more internal examiners.

ii) Viva-Voice, Oral examinations of the Project report, Dissertation etc. shall be undertaken by a Board of Examiners constituted by the respective Dean of Program with the advice of Supervisor(s).

VI. Procedure of Expulsion:

If any candidate is found to be using any unfair-means during the examination, the

invigilator may cease his/her answer sheet and report it directly to the Officer-in-Charge. The Office-in-Charge of the center may take appropriate decisions as per the rules and procedure of the examination. The Officer-in-Charge may allow the students to write the exam with a new answer sheet or may expel the student from appearing on the paper depending on the nature of unfair-means. In case of Computer based test, the students may be directed to write an apology letter and sign in the prescribed expulsion form. The student may not be allowed to write that examination.

VII. Instruction to the Students:

(i) The students shall not bring to the Examination Hall, any electronic gadget used as a means of communication or record except electronic calculators, if required.

(ii) The students shall not receive any book or printed or handwritten or photocopy (Xerox) or blank-paper from any other person while he/she is in the examination-room or in laboratory or in any other place to which he/she is allowed to have access during the course of examination.

(iii) The students shall not communicate with any other candidate in the examination room or with any other person in and outside the examination-room.

(iv) The students shall not see, read or copy anything written by any other candidate, nor shall he/she knowingly or negligently permit any other candidate to see, read or copy anything written by him/her or conveyed by him/her.

(v) The students shall not write anything on the Question Paper or in other paper or materials during the examination, or pass any kind of paper to any other candidate in the examination-room, or to any person outside the room.

(vi) The students shall not disclose his/her identity to the examiner by writing his/her name or putting any sign / symbol in any part of his answer-script.

(vii) The students shall not use any abusive language or write any objectionable remark or make any appeal to examiner by writing in any part of his answer-script.

(viii) The students shall not detach any page from the answer-script or insert any authorized or unauthorized loose sheet into it. He /she shall also not insert any other answer-script / loose sheet by removing the pins of the origin answer-scripts and re-fixing it.

(ix) The students shall not resort to any disorderly conduct inside the examination-room or misbehave with the invigilator or any other examination official.

VIII. Provision for an Amanuensis (writer):

(i) A candidate may be provided with an Amanuensis (writer) to write down on dictation on his / her behalf on ground of his / her physical disability to write down by himself / herself due to accident or any other reason. The amanuensis may be provided till he / she recovers from the physical disability. The physical disability to write down by himself / herself must be supported by Medical Certificate from a competent Medical Officer.

(ii) The qualifications of the amanuensis so provided must not be equal or higher than that of the candidate. This is also to be supported by Certificate from the Faculty of Study where the Amanuensis is provided.

(iii) Such candidates are to be accommodated in a separate room under the supervision of an invigilator so that the fellow candidates are not disturbed in the process.

C. Credit Point:

It is the product of grade point and number of credits for a course, thus, CP = GP x CR

i. Credit:

A unit by which the course work is measured. It determines the number of hours of instructions required per week. 'Credit' refers to the weightage given to a course, usually in terms of the number of instructional hours per week assigned to it. Credits assigned for a single course always pay attention to how many hours it would take for an average learner to complete a single course successfully.

ii. Grade Point:

Grade Point is a numerical weight allotted to each Grade Letter on a 10-point scale.

iii. Letter Grade:

Letter Grade is an index of the performance of students in a said paper of a particular course. Grades are denoted by letters O, A+, A, B+, B, C, P, F and Abs. Student obtaining Grade F / Grade Abs shall be considered failed/ absent and, will be required to appear in the subsequent ESE. The UGC recommends a 10-point grading system with the following (Table: 1) Letter Grades:

(i) A Letter Grade shall signify the level of qualitative/quantitative academic achievement of a student in a Course, while the Grade Point shall indicate the numerical weight of the Letter Grade on a 10-point scale.

(ii) There shall be 08 (eight) Letter Grades bearing specific Grade Points as listed in Table 1, where the Letter Grades 'O' to 'P' shall indicate successful completion of a course.

(iii) Apart from the 08 (eight) regular Letter Grades listed in Table 1, there shall be 03 (three) additional Letter Grades, which shall be awarded if a Course is withdrawn or spanned over the next Semester or remains incomplete as stated in Table 2.

Table 2: Letter Grades and Grade Points

Letter Grade	Grade Points	Description
O	10	Outstanding
A+	9	Excellent
A	8	Very Good
B+	7	Good
B	6	Above Average
C	5	Average
P	4	Pass
F	0	Fail
Abs	0	Absent
UFM	0	Unfair Means

iv. Grade Point Average:

a. SGPA (Semester Grade Point Average)

The SGPA of a student in a Semester shall be the weighted average of the Grade Points

secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered in that Semester, irrespective of whether he/she could or could not complete the Courses. More specifically, the calculation of SGPA shall take into account the Courses graded with Letter Grades 'O' to 'F' as given in Table 1.

$$\text{SGPA} = \frac{\sum_{i=1}^{n} C_i G_i}{\sum_{i=1}^{n} C_i} \qquad (1.1)$$

The SGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.1) up to two decimal places, where n is the total number of Credit Courses registered by the student in that Semester, Gi is the Grade Point secured in the i^{th} registered Course and Ci is the Credit (weight) of that Course.

b. CGPA (Cumulative Grade Point Average)

(i) The CGPA of a student in a Semester of a Programme shall be the accumulated weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered and successfully completed so far starting from the enrollment in the Programme. In other words, taking into account all the Courses graded with 'O' to 'P' as given in Table 1.1, generally the CGPA of a student shall be calculated starting from the first Semester of his/her enrolled Programme, while the CGPA of a lateral-entry student shall be calculated starting from the Semester of his/her enrollment.

(ii) The CGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.2) up to two decimal places, where N is the total number of Credit Courses registered and successfully completed so far by the student, Gi is the Grade Point secured in the ith completed Course and Ci is the Credit (weight)of that Course.

$$\text{CGPA} = \frac{\sum_{i=1}^{N} C_i G_i}{\sum_{i=1}^{N} C_i} \qquad (1.2)$$

(iii) The CGPA shall be convertible into equivalent percentage of marks using Equation Conversion of CGPA to percentage marks: = CGPA*10

D. Post-Examination

i. Transcript or Grade Card or Certificate:

A marking certificate shall be issued to all the registered students after every Semester. The Semester mark sheet will display the course details (code, title, number of credits, grade secured) along with total credit earned in that Semester.

ii. Grievance Readdress Mechanism:

Students with any dissatisfaction or grievance regarding the marks awarded in any of the Papers / Courses may appeal to the Controller of Examinations for remedial action such as Re-evaluation within 10 days of the declaration of result.

(i) A student has options to appeal for re-evaluation of his /her answer script to the Controller of Examination.

(ii) Application for re-evaluation / re-scrutiny of answer scripts shall be made in the definite proforma available with the Examination Office through the head of the respective departments within 10 days of declaration of the results of the respective examinations.

(iii) The Controller of Examination may appoint an examiner for re-evaluation and will consider and recognize the evaluation done by a University appointed examiner.

(iv) There shall be no provision for re-evaluation of the Practical Papers, Project Work, and Dissertation etc. However, the students fail in practical examination or viva voce and wish to appear again may apply to be evaluated can do so with the next schedule.

(v) After screening the application for re-evaluation, the CoE may send the answer scripts of the student to the examiners appointed by the CoE with the approval of Vice Chancellor.

(vi) The marks/grades achieved by the students after the re-evaluation shall be final and binding.

(vii) Fresh Marks – sheets / Grade Card shall be issued only if the candidate secures pass marks / passing grade in the re-evaluated paper.

(viii) Revaluation of answer scripts shall be deemed to be an additional facility provided to the students with a view to improving upon their results at the preceding examination result for any reason whatsoever shall not confer any right upon them for admission to next higher class which matters always be regulated in accordance with the relevant rules or regulations framed by the University.

(ix) If as a result of revaluation of the candidate attracts the provision of condonation of deficiency, the same may be applied to his/her only for fresh attempt.

INSTRUCTION TO TEACHERS AND STUDENTS

(Teaching and Learning Methods)

In all the courses the teacher has to select topics for teacher-method which should not be less than 20 percent. The approach will be direct class room teaching through series of lectures delivering concepts using ITC facilities, white or black board. Notes may also be circulated to the students however; the students are to be involved in preparation of the notes. The teacher will be responsible in selecting the best note for circulation. The teacher- centric methodology has recently fallen out of favor because this strategy for teaching is seen to favor passive students.

1. **Student- centric / Constructivist Approach:**

The topics of the courses may be selected at the start of the class and assigned one topic to each of the student for studying by themselves, prepare presentations, notes etc., and present at respective class time after consultation and discussion with the course teachers. The teacher facilitate the learning of the students by guiding and providing input and explaining concepts. 60 percent of the course contents may be selected for this purpose. To avoid behavior problems, teachers must lay a lot of groundwork in student- centric classrooms. Typically, it involves instilling a sense of responsibility in students. In addition, students must learn internal motivation.

a. **Project-Based Learning:** The teacher may select 5 percent of topics for the purpose and may conduct visit to the laboratory for experiments or field and survey. The selection of the topic may be done considering the available facility for the purpose. However, in the final semester of each of the programme the student has to undergo a project-Based learning at least 4 months duration. This approach will help the student to think critically, evaluate, analyze, make decisions, collaborate, and more.

b. Inquiry-Based Learning: The teacher/ students are supposed to list at least five questions in each contact hour and student solve these question or search for answer which becomes the home work for the students "question-driven" learning approach. The teacher may look for the correctness of the solution or the best possible answer and discuss in the successive class. This will help in the preparation for various competitive examination and develop a habit for search for solutions.

c. Flipped Classroom: About 10 percent of the course content has to be completed by this method. In this approach the students are asked to watch video or lecture prepared by the teacher or any video available (relevant to the course). A set of questions may be given to the students for searching answers by the students. The idea is that students should have more time in-classroom focusing on achieving these higher levels of thinking and learning. The Flipped classroom is also an acronym. The letters FLIP represent the four pillars included in this type of learning: Flexible environment, Learning culture shift, Intentional content, and Professional educator. As you can see, the second pillar refers to a culture shift from the traditional approach where students are more passive to an approach where students are active participants. As a result, this approach is also a student- centric teaching method.

d. Cooperative Learning: The remaining five percent has to be completed by cooperative learning approach. In this approach the students are allotted with problems. During the library hours the student along with the teacher visits library search probable solution for the

assigned problem. The same has to be done in group so that the students discuss among themselves for the appropriate answers. Essentially, cooperative learning believes that social interactions can improve learning. In addition, the approach recreates real-world work situations in which collaboration and cooperation are required.

The percentage categorization for the completion of a theory course

Teacher- centric or Direct Classroom Teaching: Delivery by series of lectures	20%
Student- centric Approach, Student present and deliver lectures in presence of teacher and supervised by teacher	60%
Student visit fields or perform experiments or teacher perform demonstration	05%
Flipped Classroom approach	10%
Cooperative learning approach	05%

Inquiry based approach has to be followed in all of the classes

Teacher has to distribute the topics to be considered for teaching by the above-mentioned approaches and prepare lesson plan for execution and maintain a file.

Breakdown of Credits

Sl. No	Category		Total number of Credits
1	University Core(UC)	Skill Enhancement Course (SEC)	4
		Ability Enhancement Course(AEC)	-
		Field Training	3
		Discipline Specific Elective (DSE)	-
		Value Added Course (VAC)	-
2	University Elective (UE)	Multidisciplinary Course (MDC)	11
		Value Added Course (VAC)	3
3	Program Core(PC)	Discipline Specific Core(DSC)	83
		Field Training	-
		Research /Industry Internship	16
		Summer Internship	3
4	Program Elective (PE)	Discipline Specific Elective (DSE)	11
		Value Added Course (VAC)	4
5	Faculty Core(FC)	Skill Enhancement Course (SEC)	-
		Ability Enhancement Course(AEC)	-
Total			**140**

Breakdown by categories of courses

Sl no	Category	Credits	%
1	Science	7	5
2	Engineering	132	94.29
3	Commerce and Management	1	0.71
Total		**140**	**100%**

SEMESTER WISE COURSE DISTRIBUTION

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO1101R	DIGITAL ELECTRONICS	DSC (Minor)	3	3	0	0	0	0	0	40	60	0	100
2	24BCAO1102R	PROGRAMMING FOR PROBLEM SOLVING	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAO1105R	MINI PROJECT I	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
4	24BCAO1104R	FUNDAMENTALS OF COMPUTER APPLICATIONS	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
5	24UBPD1102R	ELEMENTARY ENGLISH	AEC	2	0	0	4	0	0	0	0	0	100	100
6	24BCAO1103R	MATHEMATICS FOR COMPUTER APPLICATION	MDC	3	3	0	0	0	0	0	40	60	0	100
7	24MOSY1101R 24MOSY1102R 24MOSY1103R	1. FOUNDATION OF DIGITAL MARKETING AND E-COMMERCE 2. INTERNATIONAL LEADERSHIP AND ORGANIZATIONAL BEHAVIOUR 3. INTRODUCTION TO GIT AND GITHUB	VAC	2	0	0	0	0	0	0	0	0	100	100
8	24UBCC1101	CO-CURRICULAR ACTIVITIES	Co-Curricular	1	0	0	0	4	0	0	0	0	100	100
Credit Without Brdge Course				**20**	**11**	**0**	**8**	**12**	**0**	**0**	**160**	**240**	**600**	**1000**
9	24BCAO1106R	ALGORITHMS FOR PROBLEM SOLVING	Bridge Course	2	1	0	2	0	0	0	40	60	100	200
10	24BCAO1107R	FUNDAMENTALS OF STATISTICS	Bridge Course	2	1	0	2	0	0	0	40	60	100	200
Credit with Brdge Course				**24**	**13**	**0**	**12**	**12**	**0**	**0**	**240**	**360**	**800**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO1201R	DATA STRUCTURE USING C	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAO1202R	COMPUTER ORGANIZATION AND ARCHITECTURE	DSC (Minor)	2	2	0	0	0	0	0	40	60	0	100
3	24BCAO1203R	INTRODUCTION TO WEB TECHNOLOGY	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAO1205R	MINI PROJECT II	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
5	24UBES1001R	ENVIRONMENTAL STUDIES	MDC	2	2	0	0	0	0	0	40	60	0	100
6	24UBPD1203R	ADVANCED IMPLICIT ENGLISH	AEC	2	0	0	4	0	0	0	0	0	100	100
7	24BCAO1204R	FUNDAMENTALS OF MANAGEMENT	VAC	2	1	0	2	0	0	0	40	60	100	200
8	24UBEC1201	EXTRA CURRICULAR	Extra-Curricular	1	0	0	0	0	0	4	0	0	100	100
Credit Without Brdge Course				**19**	**11**	**0**	**10**	**8**	**0**	**4**	**200**	**300**	**600**	**1100**
9	24BCAO1206R	BASICS OF GRAPH THEORY	Bridge Course	2	2	0	0	0	0	0	40	60	0	100
10	24BCAO1207R	MATHEMATICS FOR PROGRAMMING DESIGN	Bridge Course	2	2	0	0	0	0	0	40	60	0	100
Credit with Brdge Course				**23**	**15**	**0**	**10**	**8**	**0**	**4**	**280**	**420**	**600**	**1300**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO1201R	DATA STRUCTURE USING C	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAO1202R	COMPUTER ORGANIZATION AND ARCHITECTURE	DSC (Minor)	2	2	0	0	0	0	0	40	60	0	100
3	24BCAO1203R	INTRODUCTION TO WEB TECHNOLOGY	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAO1205R	MINI PROJECT II	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
5	24UBES1001R	ENVIRONMENTAL STUDIES	MDC	2	2	0	0	0	0	0	40	60	0	100
6	24UBPD1203R	ADVANCED IMPLICIT ENGLISH	AEC	2	0	0	4	0	0	0	0	0	100	100
7	24BCAO1204R	FUNDAMENTALS OF MANAGEMENT	VAC	2	1	0	2	0	0	0	40	60	100	200
8	24UBEC1201	EXTRA CURRICULAR	Extra-Curricular	1	0	0	0	0	0	4	0	0	100	100
Credit Without Brdge Course				**19**	**11**	**0**	**10**	**8**	**0**	**4**	**200**	**300**	**600**	**1100**

9	24BCAO1206R	BASICS OF GRAPH THEORY	Bridge Course	2	2	0	0	0	0	0	40	60	0	100
10	24BCAO1207R	MATHEMATICS FOR PROGRAMMING DESIGN	Bridge Course	2	2	0	0	0	0	0	40	60	0	100
Credit with Brdge Course				**23**	**15**	**0**	**10**	**8**	**0**	**4**	**280**	**420**	**600**	**1300**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO2101R	DATABASE MANAGEMENT SYSTEMS	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAO2102R	OBJECT ORIENTED PROGRAMMING PARADIGM IN JAVA	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAO2103R	MINI PROJECT III	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
4	24BCAO2104R	WEB AND MOBILE PROGRAMMING TECHNOLOGIES	DSC (Minor)	3	2	0	2	0	0	0	0	0	100	100
5	24BCAO2105R/ 24BCAO2106R	PE I	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
6	24UULS2002R	BLSS	MDC	1	0	0	2	0	0	0	0	0	100	100
7	24UUFL2002R	FL	MDC	1	0	0	2	0	0	0	0	0	100	100
8	24MOSY2101R 24MOSY2102R 24MOSY2103R	1. REACT BASICS 2. SPRING FRAMEWORK SPECIALIZATION 3. JAVASCRIPT BASICS	MDC	1	0	0	0	0	0	0	0	0	100	100
9	24UBPD2103R	PDP	AEC	2	0	0	4	0	0	0	0	0	100	100
10		LOGICAL RESONING FOR COMPUTER SCIENCE	SEC	3	2	0	2	0	0	0	0	0	100	100
11	24BCAO2107R	FIELD BASED LEARNING	Field Training	1	0	0	0	0	0	8	0	0	100	100
				25	**12**	**0**	**18**	**8**	**0**	**8**	**120**	**180**	**1100**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO2201R	DESIGN AND ANALYSIS OF ALGORITHMS	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100
2	24BCAO2202R	BASICS OF PYTHON PROGRAMMING	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAO2203R	COMPUTER NETWORKS	DSC (Major)	3	2	0	2	0	0	0	40	60	100	200
4	24BCAO2204R	OPERATING SYSTEM	DSC (Major)	3	2	0	2	0	0	0	40	60	100	200
5	24BCAO2205R/ 24BCAO2206R	PE II	DSC (Major)	3	2	0	2	0	0	0	40	60	0	100
6	24UULS2201R	BAS	VAC	1	0	0	2	0	0	0	0	0	100	100
7	24UCDL2201R	COMPUTATIONAL SYSTEMS AND DIGITAL WORLD	VAC	1	0	0	0	4	0	0	0	0	100	100
8	24UBPD2203R	PDP	AEC	2	0	0	4	0	0	0	0	0	100	100
9	24BCAO2207R	QUANTITIVE APITUTUTE FOR COMPUTER SCIENCE	SEC	2	1	0	2	0	0	0	40	60	0	100
10	24BCAO2208R	Mini Project I	DSC (Minor)	2	0	0	0	8	0	0	0	0	100	100
				22	**13**	**0**	**16**	**4**	**0**	**0**	**240**	**360**	**600**	**1200**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO3101R	ADVANCE APPLICATION DEVELOPMENT	DSC (Major)	4	2	0	2	0	0	0	40	60	100	200
2	24BCAO3102R	CLOUD COMPUTING AND VIRTUALIZATION	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAO3104R	UI/UX DESIGN	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAO3105R/ 24BCAO3106R	PE III	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	24BCAO3107R	PROJECT - I	Project / Research	2	0	0	0	4	6	0	0	0	100	100
6	24BCAO3108R	DOMAIN TRAINING	SEC	2	1	0	2	0	0	0	40	60	100	200
7	24BCAO3109R	DESIGN THINKING AND ENTERPREUSHIP	SEC	2	1	0	2	0	0	0	40	60	100	200
8	24BCAO3110R	SUMMER INTERNSHIP	Summer Internship	4	0	0	0	16	0	0	0	0	100	100
				26	**13**	**0**	**12**	**20**	**6**	**0**	**240**	**360**	**800**	**1400**

Sl. No	Course Code	Course Title	Category I	Credit	L	T	P	S	R	O/F	Max In-Sem Marks [Theory]	Max End-Sem Marks [Theory]	Max End Sem Practical Marks	TOTAL Marks
1	24BCAO3201R	COMPUTER VISION	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	24BCAO3202R	INTRODUCTION TO CYBER SECURITY	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	24BCAO3203R / 24BCAO3204R	PE IV	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	24BCAO3205R / 24BCAO3206R	PE V	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	24BCAO3207R	PROJECT - II	Project / Research	6	0	0	0	8	24	0	0	0	100	100
				22	**12**	**0**	**8**	**8**	**24**	**0**	**160**	**240**	**500**	**900**

***IA: Internal Assessment, SEE: Semester End Examination,**
PE: Practical Examination

Detailed Syllabus of 1st Semester

SEMESTER – I										
Course Title	**Digital Electronics**									
Course code	**24BCAO1101R**	**Total credits: 3** **Total hours: 45T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
			3	**0**	**0**	**0**	**0**	**0**	**3**	
Pre- requisite	**Nil**	**Co-requisite**	**Nil**							
Programme	**Bachelor of Computer Application**									
Semester	**Fall/ I semester of the first year of the program**									
Course Objectives	1. To familiarize with the different number systems, logic gates, and combinational and sequential circuits utilized in the different digital circuits and systems. 2. The course will help in design and analysis of the digital circuit and system. 3. To impart how to design Digital Circuits.									
CO1	Interpret different number systems, binary codes and Boolean algebra to minimize logic expressions									
CO2	Develop K-maps to minimize and optimize logic functions up to 5 variables									
CO3	Infer the knowledge about various logic gates and logic families and analyze basic circuits of these families									
CO4	Design and implement Combinational and Sequential logic circuits.									
CO5	Describe and compare various memory systems, shift registers and analog to digital and digital to analog conversion circuits									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Fundamentals of Digital Systems and logic families:** Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Boolean algebra, examples of IC gates, number systems-binary, signed binary, octal hexadecimal number, binary arithmetic, one's and two's complements arithmetic, codes, error detecting and correcting codes, characteristics of digital ICs, digital logic families, TTL, Schottky TTL and CMOS logic, interfacing CMOS and TTL, Tri-state logic	**10**	Remembering and understanding the different logic gates and applying them for different circuit.	1,2,3
II	**Combinational Digital Circuits:** Standard representation for logic functions, K-map representation, simplification of logic functions using K-map, minimization of logical Faculty of Engineering& Technology| Assam down town University functions. Don't care conditions, Multiplexer, DeMultiplexer/Decoders, Adders, Subtractors, BCD arithmetic, carry lookahead adder, serial adder, ALU, elementary ALU design, popular MSI chips, digital comparator, parity checker/generator, code converters, priority encoders, decoders/drivers for display devices, QM method of function realization	**12**	Applying and analyzing different digital circuits.	3,4
III	**Sequential circuits and systems:** A 1-bit memory, the circuit properties of Bistable latch, the clocked SR flip flop, J- K-T and D types flip flops, applications of flip flops, shift registers, applications of shift registers, serial to parallel converter, parallel to serial converter, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, counters design using flip flops, special counter IC's, asynchronous sequential counters, applications of counters.	**10**	Applying and analyzing different sequential circuits	3,4

IV	**A/D and D/A Converters:** Digital to analog converters: weighted resistor/converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter ICs, sample and hold circuit, analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D converter, A/D converter using voltage to frequency and voltage to time conversion, specifications of A/D converters, example of A/D converter ICs	**7**	Describe, Differentiate, and analyzing A/D and D/A converters	2,4
V	**Semiconductor memories and Programmable logic devices:** Memory organization and operation, expanding memory size, classification and characteristics of memories, sequential memory, read only memory (ROM), read and write memory(RAM), content addressable memory (CAM), charge de coupled device memory (CCD), commonly used memory chips, ROM as a PLD, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).	**8**	Understanding, analyzing and explaining different memories and logic.	1,2,4

TEXT BOOKS:

T1: Modern Digital Electronics, R. P. Jain, McGraw Hill Education, 2009.

REFERENCE BOOKS:

R1: Digital logic and Computer design, M. M. Mano, Pearson Education India, 2016.

R2: Fundamentals of Digital Circuits, A. Kumar, Prentice Hall India, 2016.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Interpret different number systems, binary codes and Boolean algebra to minimize logic expressions	1 and 3
2	Develop K-maps to minimize and optimize logic functions up to 5 variables	2,3, and 4
3	Infer the knowledge about various logic gates and logic families and analyze basic circuits of these families	1,2, and 7
4	Design and implement Combinational and Sequential logic circuits.	1,2,4, and 7
5	Describe and compare various memory systems, shift registers and analog to digital and digital to analog conversion circuits	1,2,5, and 7

SEMESTER – I									
Course Title	**PROGRAMMING FOR PROBLEM SOLVING**								
Course code	**24BCAO1102R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives	1. To formulate simple algorithms for arithmetic and logical problems. 2. To test and execute the programs and correct syntax and logical errors. 3. To solve real-time problems using programming								
CO1	Explain the basic terminology used in computer programming to write, compile and debug programs in C programming language.								
CO2	Examine the syntax and semantics and be fluent in the use of various Operators of C Programming.								
CO3	Demonstrate the concept of Searching and Sorting in programming.								
CO4	Develop programs to describe the applications of derived data types such as arrays and strings etc.								
CO5	Illustarte the dynamics of memory by the use of pointers and Structures								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to C Programming: Introduction, History, Features, Advantages and Disadvantages, Structure of C program, Compiling Process, C Pre-processor and Header Files, Library Function, Character Set, Comments, Tokens and its types. Data types, Escape Sequences, Pre-processors Directives.	**7**	Describe, illustrate, and explain computer, algorithms, flowchart	1,2
II	Operators and Expressions Arithmetic Operator, Relational Operator, Logical Operator, Assignment Operator, Increment/decrement Operator, Conditional Operator, Bitwise Operator, Comma Operator, Sizeof Operator, Operator Precedence and Associativity.	**10**	Describe, illustrate, and explain different operators	1,2
III	Function: Functions (including using built in libraries), Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference Recursion: Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quick sort or Merge sort.	**10**	Describe, illustrate, and explain function, recursive function.	1,2
IV	Structure: Structures, Defining structures and Array of Structures Pointers: Idea of pointers, Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation)	**8**	Describe, illustrate, and explain structure, linked-list	1,2
V	File handling (only if time is available, otherwise should be done as part of the lab)	**10**	Describe, illustrate, and explain file handling	1,2
Practical				
Practical 1	C program to display "hello world".	1	Creating a first program to display hello world in C	3, 5
Practical 2	Create a simple program where we will Print Integer in C where variable value is pre-defined.	1	Creating and displaying a value of a variable in C	3, 5

Practical 3	Program where we will Print the Integer in C where the value of the variable is entered by the user.	1	Creating and displaying the value variable from user input.	3, 5
Practical 4	Input integer, float and character values using one scanf() statement in C.	1	Creating and displaying different types of data types.	3, 5
Practical 5	C Program to Find the Size of int, float, double and char.	1	Creating and finding the size of different data types.	3, 5
Practical 6	Perform addition, subtraction, multiplication and division of two numbers.	1	Creating and displaying the different types of calculation program using C	3, 5
Practical 7	C Program to Calculate Percentage of 5 Subjects.	1	Creating and calculating the percentage of five subject in C	3, 5
Practical 8	C Program to Calculate Simple interest.	1	Creating and calculating the simple interest.	3, 5
Practical 9	C Program to Calculate Compound interest.	1	Creating and finding the compound interest.	3, 5
Practical 10	C Program to Find Area and Perimeter of Rectangle.	1	Creating and finding the area and perimeter of the rectangle.	3, 5
Practical 11	Radius and Area of Circle program in C.	1	Creating and finding the radius and area of a circle.	3, 5
Practical 12	Swap numbers using temporary variable.	1	Creating a program to swap two numbers with the help of a temporary variable.	3, 5
Practical 13	Swap numbers without using temporary variable.	1	Creating a program to swap two numbers without using the temporary variable.	3, 5
Practical 14	C example to print "hello world" using if statement and without using semicolon.	1	Creating a C program for implementing the *if* statement.	3, 5
Practical 15	C Program to find the largest number among three numbers using the if statement.	1	Creating a C program to find the largest number among the three numbers.	3, 5
Practical 16	C Program to find the largest number among three numbers using the if else ladder.	1	Creating a C program to find the largest number among the three numbers using if-else statement	3, 5
Practical 17	C Program to find the largest number among three numbers using nested if-else statement.	1	Creating a C program to find the largest number	3, 5

			among the three numbers using the nested if-else.	
Practical 18	C Program to check whether a number is even or odd.	1	Creating a C program to check whether the given number is odd or even.	3, 5
Practical 19	C Program to check whether a character is a Vowel or a Consonant.	1	Creating a program to check whether a character is a vowel or consonant.	3, 5
Practical 20	C program to check whether a number is Positive or Negative or Zero.	1	Creating a C program to check if the given number is positive or negative.	3, 5
Practical 21	C program to find the factorial of a number.	1	Creating a program to find the factorial of a number.	3, 5
Practical 22	C example to print "hello world" using switch statement and without using semicolon.	1	Creating a C program to display hello world using switch statement.	3, 5
Practical 23	C Program to Find Largest of Two Numbers using Switch Case.	1	Creating and finding the largest of two numbers using switch case statement.	3, 5
Practical 24	C Program to Find Largest of Two Numbers using Conditional Operator.	1	Creating a program to find the largest of two using conditional operators.	3, 5
Practical 25	C Program to print first 10 natural numbers using a for loop	1	Creating a C program to display the first 10 natural numbers using a for loop.	3, 5
Practical 26	C Program to calculate the sum of first n natural numbers using a for loop	1	Creating a C program to display the first n natural numbers using a for loop.	3, 5
Practical 27	C program to print all natural numbers in reverse in a given range using a for loop.	1	Creating a program to print all natural numbers in reverse	3, 5
Practical 28	C Program to print even and odd natural numbers using a for loop	1	Creating a program to print even and odd numbers using a for loop.	3, 5
Practical 29	C Program to check whether a number is a prime number or not.	1	Creating and displaying whether a given number is a prime number or not.	3, 5
Practical 30	Print prime numbers between 1 to n in c language using a for loop.	1	Creating a C program to check the prime numbers between 1-10 using a for loop.	3, 5
Practical 31	Write a c program to print Fibonacci series using a for loop.	1	Creating a C program to print Fibonacci series	3, 5

			using a for loop.	
Practical 32	C example to print "hello world" using a while loop and without using a semicolon.	1	Creating a C program using a while loop.	3, 5
Practical 33	C Program to find the sum of the first and last digit using a while loop.	1	Creating a C program to demonstrate while loop for finding the sum of the first and last digit.	3, 5
Practical 34	C program to reverse a number using a while loop.	1	Creating a program for reversing a number using a while loop.	3, 5
Practical 35	C Program to print the first 10 natural numbers using a while loop.	1	Creating a program to print the first 10 natural numbers using a while loop.	3, 5
Practical 36	C program to print all natural numbers in reverse from n to 1 using a while loop.	1	Creating a C program to print all natural numbers in reverse using a while loop.	3, 5
Practical 37	Write a c program to print Fibonacci series using while loop	1	Creating a program to display the Fibonacci series using a while loop.	3, 5
Practical 38	C example to print "hello world" using do while loop.	1	Creating a program to display the do while loop.	3, 5
Practical 39	C Program to print first n natural numbers using do while loop.	1	Creating a program to display the first n natural number using a do while loop.	3, 5
Practical 40	C program to print all the numbers from 20 to 1 in reverse order on the screen using do while loop.	1	Displaying the natural numbers in reverse using a do while loop.	3, 5
Practical 41	C program to print all the even numbers from 10 to 20 on the screen using do while loop	1	Displaying the even numbers from 10-20 using a do while loop.	3, 5
Practical 42	C Program to print multiplication table using do...while loop.	1	Displaying the multiplication table using a do while loop.	3, 5
Practical 43	C program to input a 4-digit number and find the sum of its digits using do while loop.	1	Displaying the sum of 4 digit numbers and finding its sum using a do while loop.	3, 5

TEXT BOOKS:

T1: E. Balaguruswamy, Programming in ANSI C, Tata McGraw-Hill.
T2: Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, PrenticeHall of India.

REFERENCE BOOKS:

R1: Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the basic terminology used in computer programming to write, compile and debug programs in C programming language.	1,2 & 8
2	Examine the syntax and semantics and be fluent in the use of various Operators of C Programming.	1,2 & 8
3	Demonstrate the concept of Searching and Sorting in programming.	1,2 &8
4	Develop programs to describe the applications of derived data types such as arrays and strings etc.	1,2 ,4, 5,6,7 & 8
5	Illustrate the dynamics of memory by the use of pointers and Structures	1,2 ,4

SEMESTER – I									
Course Title	**MATHEMATICS – I**								
Course code	**24BCAO1103R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives	1. To make understand to evaluate definite and improper integrals. Apart from these some special functions like Beta and Gamma function must be introduced. 2. To provide the application of differential and integral calculus. 3. To make understand the convergence and divergence of sequence and series.								
CO1	Illustrate the foundation concepts in Mathematics like Matrices, Set, relations and functions								
CO2	Outline Differential Integration and Differentiation and with this background.								
CO3	Learn about Multivariable Calculus for Limit, continuity and partial derivatives								
CO4	Decribe matrices like Inverse and rank of a matrix, rank-nullity theorem								
CO5	Apply the foundation concepts in programming and analytical subjects.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	***Calculus:*** Evaluates and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions.	**8**	Describe, illustrate, and explain Integrals and its applications	1,2
II	***Calculus:*** Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; indeterminate forms and L'Hospital's rule; Maxima and minima..	**12**	Describe, illustrate, and explain various theorems of calculus	1,2
III	***Sequences and series:*** Convergence of sequence and series, tests for convergence; Power series, Taylor's series, series for exponential, trigonometric and logarithm functions; Fourier series: Half range sine and cosine series, Parseval's theorem.	**8**	Describe, illustrate, and explain numbers series	1,2
IV	***Multivariable Calculus (Differentiation):*** Limit, continuity and partial derivatives, directional derivatives, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers; Gradient, curl and divergence.	**8**	Describe, illustrate, and explain multivariable calculus	1,2
V	***Matrices:*** Inverse and rank of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Determinants; Eigenvalues and eigenvectors; Diagonalization of matrices; Cayley-Hamilton Theorem, and Orthogonal transformation.	**9**	Describe, illustrate, and explain types of matrices	1,2

TEXT BOOKS:

T1:G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
T2:Erwinkreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
T3:Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.

REFERENCE BOOKS:

R1: N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
R2: B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate the foundation concepts in Mathematics like Matrices, Set, relations and functions	1 & 6
2	Outline Differential Integration and Differentiation and with this background.	1 & 6
3	Learn about Multivariable Calculus for Limit, continuity and partial derivatives	1 & 6
4	Describe matrices like Inverse and rank of a matrix, rank-nullity theorem	1 & 6
5	Apply the foundation concepts in programming and analytical subjects.	1 & 6

SEMESTER – I									
Course Title	**Fundamentals of Computer Applications**								
Course code	**24BCAO1104R**	**Total credits: 4** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives	quire the basic knowledge of Computer Systems and their uses. ow about Application Software and Programming environments fy and protect from computer viruses and online threats.								
CO1	Explain the working of a computer and its various components.								
CO2	Discuss the use of Software and programming in a computer system.								
CO3	Outline the basic concepts of Computer Networks and Internet Protocols.								
CO4	Develop Proficiency in Identifying different types of computer viruses, worms, and malware to enhance threat awareness.								
CO5	Apply the role of various Software packages for Office Automation								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Basics of Computer** What is Computer and its History, Generation of Computers, Classification of Computers; Components of a Digital Computer: CPU, Memory, I/O devices; Storage Devices (RAMs, ROMs, Hard Disk etc.), Backup System and it is use.	**8**	Understand and explain the different types of computers, working of computer, and identify the various components.	1, 2
II	**Software and Languages** What is Computer Software and its need, Types of Computer Software's; Computer languages, Generation of Computer Languages, Classification of Computer Languages.	**8**	Understand the basics of software and programming concepts, identify different types of programming language.	1, 2
III	**Networks** LAN, MAN, WAN, Internet, Intranet, Browsers, Basic Internet Protocols (http, TELNET, FTP).	**8**	Understand the working of an Internet and Network. Identify and apply the different types of networks.	1, 2, 3
IV	**Computer Viruses and Security**. Type of Virus, Worms, Malware, Firewalls, Phishing, Anti-Virus.	**8**	Understand and identify different types of viruses and worms, firewall and phishing.	1, 2
V	**Software packages for Office Automation** (Any available standard Office Suite may be referred) Word Processing software, Spreadsheet software, Presentation software, Database software, Drawing and Graphics manipulation software.	**12**	Describe the different types of MS Office package. Apply and creating different files using MS Office Software.	3, 5
Practical Components				
Practical 1	Identify the components of a computer	1	Identifying and understanding the components of a computer.	2
Practical 2	Showing and explain the working process of the computer's components.	1	Remembering and understanding the working process of computer.	1,2
Practical 3	A computer is not powering on. List and demonstrate the steps you would take to	1	Applying the knowledge of computers to fix some	3

	diagnose and fix the issue.		issues related to computers.	
Practical 4	Access the BIOS/UEFI on a computer and describe how to change the boot order to prioritize booting from a USB drive.	1	Understanding and applying the knowledge on BIOS.	2,3
Practical 5	Demonstrate how to install an operating system (e.g., Windows, Linux) on a computer. Include partitioning the hard drive and configuring system settings.	1	Applying the knowledge of installation to install software's.	3
Practical 6	Connect and configure a printer or a scanner to a computer. Install the necessary drivers and software, and demonstrate printing or scanning a document.	1	Applying the knowledge to configure a printer.	3
Practical 7	Install a software application (e.g., a web browser or office suite) on a computer and configure its basic settings.	1	Applying the configuration knowledge for software installation.	1,3
Practical 8	Write a simple program in a language of your choice (e.g., C, C++) that takes user input and performs a basic calculation or task.	1	Applying and creating a first program in any language.	3,5
Practical 9	Demonstrate how to use version control software (e.g., Git) to clone a repository, make changes, commit those changes, and push them to a remote repository.	1	Understanding and applying the Git knowledge to clone repositories, etc.	2,3
Practical 10	Use an Integrated Development Environment (IDE) to write, debug, and run a simple program. Explain the benefits of using an IDE.	1	Using different IDEs for creating, solving and running simple programs.	3
Practical 11	Set up a small local area network (LAN) with at least two computers. Ensure that they can communicate with each other by sharing files.	1	Identifying and applying the knowledge on different types of networking.	2,3
Practical 12	Explain the difference between static and dynamic IP addressing. Configure a computer with a static IP address.	1	Explaining and differentiating between static and dynamic IP addressing.	2,4
Practical 13	Use the ping and traceroute (or tracert on Windows) commands to diagnose network connectivity issues between two devices on a network.	1	Using ping and traceroute knowledge to diagnose networks.	3
Practical 14	Install an antivirus program on a computer. Perform a full system scan and demonstrate how to handle detected threats.	1	Understanding and demonstrating how to handle threats.	2,3
Practical 15	Explain safe browsing practices to avoid malware. Show how to configure a web browser with extensions and settings to enhance security.	1	Understanding and explaining safe browser practices.	2,3
Practical 16	Demonstrate how to create strong passwords and manage them using a password manager. Explain the importance of using different passwords for different accounts	1	Explaining and creating strong passwords using different characters and different passwords for different accounts.	2,5
Practical 17	Configure a software firewall on a computer to block all incoming connections except for a specific application (e.g., a web server).	1	Configuring a software firewall to block any connections from malicious users.	3,5
Practical 18	Make a Resume. It must include the features mentioned below— · Tables · Alignment	1	Applying and creating a resume with MS Word.	3,5

	· Bullets · At least 3 Font Styles and 3 Font Sizes One picture			
Practical 19	Create time table in MS Word.	1	Applying and creating a time table in MS Word.	3,5
Practical 20	Create University ID card in MS Word.	1	Applying different styles to create a University ID card.	3,5
Practical 21	Use MS-Word to create Project Reports or Thesis.	1	Applying and creating reports in MS Word.	3,5
Practical 22	Make a presentation of 3 slides in MS PowerPoint. The first slide should have a title and subtitle, the second slide an image with a caption, and the third slide a bulleted list.	1	Applying and building presentation with MS PowerPoint	3,5
Practical 23	Create time table in MS Excel	1	Creating time table in MS Excel	5
Practical 24	Prepare a stationary order for the month of March in MS Excel.	1	Applying the knowledge of MS Excel to create stationary order details.	3,5
Practical 25	Prepare a student marks distribution table in MS Excel	1	Creating a marks records using MS Excel	5
Practical 26	VLOOKUP and INDEX function in MS Excel	1	Using formulas like VLOOKUP and INDEX in MS Excel.	3,5
Practical 27	Joining two strings in MS Excel	1	Using excel formulas to join two strings.	3,5
Practical 28	Conditional Formatting in MS Excel	1	Applying conditional formatting for different data cells.	3

TEXT BOOKS:

T1: Introduction of Computer Sc. ITL ESL, Pearson Education India.
T2: Computer Fundamentals. Rajaraman, V.

REFERENCE BOOKS:

R1: Computer Fundamentals: Concepts, Systems & Applications Priti Sinha, Pradeep K., Sinha, BPB Publications

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the working of a computer and its various components.	3,8
2	Discuss the use of Software and programming in a computer system.	1,2,3,8
3	Outline the basic concepts of Computer Networks and Internet Protocols.	2,8
4	Develop Proficiency in Identifying different types of computer viruses, worms, and malware to enhance threat awareness.	2,3,8
5	Apply the role of various Software packages for Office Automation	2,3,8

SEMESTER – I									
Course Title	**ELEMENTARY ENGLISH**								
Course code	**24UBPD1102R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	Bachelor of Business Administration/Bachelor of Hotel Management and Catering Technology/Bachelor of Business Administration (I)/Bachelor of Business Administration (Industry Integrated)/Bachelor of Social Work/Bachelor of Arts in Sociology/Bachelor of Arts in Psychology/Bachelor of Trauma, Emergency and Disaster Management/Bachelor of Science in Biotechnology/Bachelor of Science in Microbiology/Bachelor of Science in Food Nutrition and Dietetics/Bachelor of Computer Application/ Bachelor of Computer Application (P)/Bachelor of Science in Information Technology (P)/ Bachelor of Science in Information Technology/Bachelor of Computer Application on Cloud Technology and Information Security/Bachelor of Computer Application on Artificial Intelligence and Machine Learning/Bachelor of Computer Application on Artificial Intelligence and Machine Learning								
Semester	**Fall/I or Winter/II Semester of First Year of the Programme**								
Course Objectives	To recognize and identify parts of a sentence and their significance in a language. To enhance listening and speaking/skills for self-development. To give insight into English pronunciation and into central concepts in phonetics. Introduction to the various modes of communication will enhance their knowledge of communication.								
CO1	It enables learners to recognize the structure of a sentence and its variations as they learn to understand, speak and write.								
CO2	Introduction to Phonetics and its importance will improve the learners' pronunciation								
CO3	Students will be able to identify, pick and form different kinds of sentences.								
CO4	Knowledge of communication will be enhanced through practical examples.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Unit 1- Basics of Grammar (Flipped classroom)** i. Parts of Speech ii. Articles iii. Auxiliary Verbs iv. Affirmative and Negative Sentences	**4**	Understanding, identifying and analyzing part of speech, articles, auxiliary verbs, affirmative and negative sentences.	1,2,4
II	**Grammar (Flipped classroom)** i. Determiners ii. Sentence Construction iii. Types of Sentences (Assertive, Imperative, etc.) iv. Degree of Comparison	**4**	Understand, identify and construct clear sentences, determiners and degree of comparison.	1,2,3
III	**Speaking Skills** **i**. Introduction and Greetings ii. Pronunciation, Intonation, Stress iii. Asking and offering information	**6**	Develop the ability to effectively introduce themselves, engage in self-discovery, grasp the basics of phonetics and pronunciation, deliver extempore speeches, and use video recording for self-reflection to enhance their speaking skills.	2,4
IV	**Communication Skills** . Introduction to Communication . Process and Types of Communication,	**6**	Understand the fundamentals and importance of communication skills, identify the purpose and types of communication, differentiate between formal and	5

	. Formal and informal communication . Understanding Barriers to Communication		informal communication	
V	**Presentation Skills** . Introduction . Essential characteristics of a good presentation . Use of Visual Aids in Presentation	**8**	Ability to present self-introduction and using visual aids in presentation.	2

TEXT BOOKS:

T1:Chaturvedi, P.D., ChaturvediMukesh, 2011.*Business Communication: Concepts, Cases and Applications*, second edition, Pearson, Noida.

T2:Alex K., Chand, S, 2009. *Soft Skills: Know Yourself and Know the World*, first edition, S.Chand& Company Ltd.: New Delhi.

REFERENCE BOOKS:

R1: Quirk, Randolp. (2010*) A Comprehensive Grammar of the English Language* Randolph Quirk, Sidney Greenbaum, Pearson Education India

R2: Marks, Jonathan. (2017) *IELTS Advantage Speaking and Listening Skills: A step-by-stepguide to a high IELTS speaking and listening score. Book + CD-ROM*, Delta Publishing by Klett

OTHER LEARNING RESOURCES:

- https://youtu.be/bEB8- SWMYhIhttps://youtu.be/-zZau_dttRY

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	It enables learners to recognize the structure of a sentence and its variations as they learn to understand, speak and write.	2 and 6
2	Introduction to Phonetics and its importance will improve the learners' pronunciation	6 and 8
3	Students will be able to identify, pick and form different kinds of sentences.	2 and 6
4	Knowledge of communication will be enhanced through practical examples.	2 and 6

SEMESTER – I									
Course Title	**CO-CURRICULAR ACTIVITY**								
Course code	**24UBCC1101**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives	It is to develop the social and soft skills and to promote a holistic development of the learners								
CO1	Connect and adapt cultural diversity among communities.								
CO2	Enhance team for working toward a shared vision								
CO3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals								
CO4	Learn to effectively communicate, delegate responsibilities and motivate team members.								
CO5	Develop strong teamwork and collaboration skills by engaging in group activities.								
Course Contents	AdtU encourages activities that are aimed to develop the social and soft skills of students and promote a holistic development of the learners. Keeping in mind the 360 degree learning methodology, the students are engaged in different activities other than their regular classes. Experts are invited to conduct workshops that benefit the students. Technical quizzes are conducted to enhance the knowledge of the students. Technical seminars help them to develop their public speaking abilities.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Connect and adapt cultural diversity among communities.	4,5,6,7,8
2	Enhance team for working toward a shared vision	4,5,6,7,8
3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals	4,5,6,7,8
4	Learn to effectively communicate, delegate responsibilities and motivate team members.	5,6,7,8
5	Develop strong teamwork and collaboration skills by engaging in group activities.	5,6,7,8

SEMESTER – I									
Course Title	**MOOCS I (INTRODUCTION TO GIT AND GITHUB)**								
Course code	**24MOSY1103R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives	1. To learn and apply basic Git commands (init, clone, add, commit, status, log) to manage local repositories and track changes. 2. To create and manage branches, perform merges, and resolve conflicts to maintain a smooth development workflow. 3. To create a GitHub account, set up remote repositories, and perform push and pull operations to synchronize local and remote repositories.								
CO1	Acquire a solid foundation in using Git for version control, including initializing repositories, committing changes, and managing version history								
CO2	Master advanced Git functionalities such as branching, merging, and resolving conflicts to manage and streamline development workflows.								
CO3	Develop the skills to effectively use GitHub for hosting remote repositories, collaborating on projects, and managing code reviews.								
CO4	Learn best practices for collaborating with others using Git and GitHub, including forking, pull requests, and managing repository settings.								
CO5	Create and maintain a professional portfolio on GitHub, showcasing projects and demonstrating the ability to use Git and GitHub effectively.								

Unit	**Contents**	**Contact Hrs.**	**Learning Outcome**	**BL**
I	**Introduction to Version Control:** In this module, introduction to the concept of version control, which will make managing and rolling back the code look super easy. Learn how to differentiate between files and the tools at your disposal to make this happen. Next, you'll be introduced to Git and how you can leverage that platform to improve your coding abilities. Once you've got a grasp on what Git is, you'll install it and start using it to create and clone code repositories. Last up, you'll deep dive into Git in order to get more familiar with the different tools and commands it has to offer.	**9**	Understanding and applying the knowledge of Git and GitHub and its version control.	1,3

II	**Using Git Locally:** In this module, you'll dive into advanced Git interactions by skipping the staging area for small code changes, and understand how Git uses the HEAD alias to represent checked-out snapshots. Next, you'll explore how to move and remove files before finally getting a study guide to help you in your Git explorations! The next step of your learning will include how to undo changes before committing and how to amend commits once they're submitted. Finally, you'll be able to identify errors in commits that were submitted a while back. In the final section of this module, you'll explore the concept of branching and merging. You'll learn what a branch is, how to create one, and how they work in harmony with you and your code. Once you're comfortable with branching, you'll dive into merging, how it works with branched data, and how to deal with merge conflicts.	**6**	Remembering, understanding, and applying git knowledge to use different functions.	1,2,3
III	**Working with Remotes:** In this module, you'll be introduced to GitHub and learn how it works with Git. You'll create new repositories and clone those repositories onto your computer. Next, we'll explain what a remote repository is, how we can work with them, and how we can host them. You'll get familiar with commands like modify, stage, and commit, which will be used for local changes, as well as the fetch command, which can pull any changes from remote repositories. We'll cover secure shell protocol and when to use API keys. Our final lesson will focus on learning about conflicts. This will allow you to explore the concepts of pull-merge-push workflows, pushing remote branches and rebasing your changes.	**5**	Understanding and applying the knowledge of git in GitHub and creating new repositories.	1,3,5

IV	**Collaboration:** In this module, you'll continue to explore the collaboration tools available in Git. You'll learn about the tools that are available to help improve the quality of your code and to better track your code. This includes an overview of pull requests and how the typical workflow of a pull request looks like on GitHub. Next, you'll dive into how you can squash changes in your code. We'll finish up by providing you with a study guide on fork and pull requests. Next up, we'll cover what code reviews are and what the code review workflow looks like. Then, you'll learn about how to use code reviews on GitHub. The final lesson of this module will focus on managing projects. We'll take a rundown of best practices on managing projects and how to manage collaboration within those projects. We'll explore different ways of tracking issues and finish up by discussing the concept of continuous integration with your projects.	**6**	Understanding, analyzing, and applying new tools available.	1,3,4

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Acquire a solid foundation in using Git for version control, including initializing repositories, committing changes, and managing version history	3 and 8
2	Master advanced Git functionalities such as branching, merging, and resolving conflicts to manage and streamline development workflows.	2 and 3
3	Develop the skills to effectively use GitHub for hosting remote repositories, collaborating on projects, and managing code reviews.	3 and 7
4	Learn best practices for collaborating with others using Git and GitHub, including forking, pull requests, and managing repository settings.	6 and 7
5	Create and maintain a professional portfolio on GitHub, showcasing projects and demonstrating the ability to use Git and GitHub effectively.	7 and 8

SEMESTER – I									
Course Title	**BRIDGE I (ALGORITHMS FOR PROBLEM SOLVING)**								
Course code	**24BCAO1106R**	**Total credits: 2 Total hours: 15L + 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To understand the basics of algorithm and computational knowledge. 2. To understand computational thinking. 3. To apply algorithmic problem-solving techniques to real world problems. 4. To write efficient code for solving computational challenges.								
CO1	Understand computational thinking and its four pillars.								
CO2	Understand and apply algorithms for various problems.								
CO3	Understand and analyze the principle of divide and conquer.								
CO4	Understanding graph theory and representation.								
CO5	Understanding linked list and trees in data structure.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to Computational Thinking: About computational thinking, four pillars of computational thinking.	**8**	Understanding and applying computational thinking for different real-world computer-based problems.	1,3
II	Introduction to algorithm: About algorithm, basic algorithm like search and sort, pseudocode	**10**	Understanding and applying different algorithm and analyzing the algorithm	1,3,4
III	Divide and Conquer: Principles of divide and conquer, Quick Sort, Merge Sort	**8**	Understanding, applying divide and conquer method, and analyzing quick sort and merge sort.	1,3,4
IV	Graph Algorithms: Basics of graph theory, Graph representation and traversal. Shortest path algorithm.	**10**	Understanding, applying graph structure, and analyzing shortest path algorithm.	1,3,4
V	Data Structures: Basic data structures, linked list, trees, minimum spanning trees	**9**	Understanding data structure, and analyzing minimum spanning trees.	1,4

TEXTBOOKS:

T1: "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.

T2: “Computational Thinking: A Beginner's Guide to Problem-Solving and Programming” by Karl Beecher.

T3: "C Programming Absolute Beginner's Guide" by Perry and Miller

REFERENCES:

R1: "Code: The Hidden Language of Computer Hardware and Software" by Charles Petzold.

R2: "Algorithms to Live By: The Computer Science of Human Decisions" by Brian Christian and Tom Griffiths.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand computational thinking and its four pillars.	1,2,3,4,5,6,7, and 8
2	Understand and apply algorithms for various problems.	1,2,3,4,5,6,7, and 8
3	Understand and analyze the principle of divide and conquer.	1,2,3,4,5,6,7, and 8
4	Understanding graph theory and representation.	1,2,3,4,5,6,7, and 8
5	Understanding linked list and trees in data structure.	1,2,3,4,5,6,7, and 8

SEMESTER – I									
Course Title	**BRIDGE II (FUNDAMENTALS OF STATISTICS)**								
Course code	**24BCAO1107R**	**Total credits: 2** **Total hours: 15L+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To introduce the foundations of probability and statistical methods. 2. To explain the concepts in random variables and several distributions in engineering applications. 3. To learn the concepts of correlation, regression and estimations and their properties and to explain the concept of testing of hypotheses.								
CO1	Understanding the fundamental concepts of probability and statistics.								
CO2	Understanding measures of central tendency and measures of dispersion.								
CO3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.								
CO4	Explain the concept of sampling distributions and estimation.								
CO5	Analyze the concepts of hypothesis testing.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to statistics, population vs sample, collection of data, primary and secondary data, types of variables: dependent, independent, categorical and continuous variables, data visualization,	**10**	Understanding and applying statistical knowledge.	1,3
II	Measures of Central Tendency (Mean, median, mode, Skewness and symmetry) Measures of Dispersion (Range, variance, standard deviation, Interquartile range)	**8**	Understanding and applying different ways of measuring the central tendency.	1,3
III	Introduction to Probability: Probability rules (addition, multiplication, complement) Probability Distributions: Discrete and continuous distributions, Binomial, Poisson, and normal distributions.	**10**	Understanding, applying probability and its method.	1,3
IV	Sampling and Sampling Distributions, Introduction to Random sampling, Confidence Intervals: Basic idea of estimation, Constructing simple confidence intervals.	**8**	Understanding, applying sampling distribution, and analyzing confidence interval.	1,3,4
V	Hypothesis Testing: Introduction to Hypothesis Testing, Formulating hypotheses, Basic understanding of p values, Common Tests.	**9**	Understanding hypothesis testing, and analyzing different hypothesis testing method.	1,4

TEXT BOOKS:

T1: Richard A. Johnson, "Miller & Freund's probability and statistics for engineers", 9th Edition, Pearson, 2017.

T2: S.C. Gupta and V.K. Kapoor, "Fundamentals of Mathematical Statistics", 11th Edition, Sultan Chand & Sons Educational Publications, 2012.

REFERENCE BOOKS:

R1: W. Feller, "An Introduction to Probability Theory and its Applications", 3rd Edition, Wiley, 1968.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understanding the fundamental concepts of probability and statistics.	1,2,3,4,6,7, and 8
2	Understanding measures of central tendency and measures of dispersion.	1,2,3,4,6,7, and 8
3	Describe important probability distributions like Binomial, Poisson Distributions and normal distribution.	1,2,3,4,6,7, and 8
4	Explain the concept of sampling distributions and estimation.	1,2,3,4,6,7, and 8
5	Analyze the concepts of hypothesis testing.	1,2,3,4,6,7, and 8

SEMESTER – I									
Course Title	**MINI PROJECT-I**								
Course code	**24BCAO1105R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	0	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives **(Minimum 3)**	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: • theoretical studies • computer simulations • hardware construction; 2. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 3. Deliver a seminar on the	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	3, 4, 5

	general area of work being undertaken and specific contributions to that field; 4. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

OTHER LEARNING RESOURCES:

As given by the Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written in oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – II									
Course Title	**Data Structure using C**								
Course code	**24BCAO1201R**	**Total credits: 4** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Fundamentals of programming logic	**Co-requisite**	C Programming						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. To impart the basic concepts of data structures and algorithms. 2. To understand basic concepts about stacks, queues, lists, trees and graphs. 3. To understand concepts about searching and sorting techniques								
CO1	Illustrate the Basic concepts of Data Structures.								
CO2	Apply Data Structure techniques on computing problem.								
CO3	Analyse and develop algorithms to solve real world problems.								
CO4	Implement and developed program for various concepts of data structures including array, stack, queue, graphs and trees.								
CO5	Demonstrate and analyze various sorting algorithms and hashing techniques.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.	**8**	Analyze the algorithms to determine the time and computation complexity and justify the correctness	1,2, 3
II	**Stacks and Queues**: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.	**10**	Analyze the problem of stacks, queues and linked list to determine the time and computation complexity	2,3
III	**Linked Lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.	**10**	Analyze the problem of stacks, queues and linked list to determine the time and computation complexity	3

IV	**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis	**9**	Implement basic tree-based traversal and search algorithms, learn about applications using tree	1.3
V	**Sorting and Hashing:** Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing. **Graph:** Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.	**8**	Summarize Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity. Graph search and traversal algorithms and determine the time and computation complexity	2,3
Practical Component				
Practical 1	Array implementation of stack using C program	2	Understand the basics of stack	2,3
Practical 2	Array implementation of queue using C program	2	Understand the basics of queue	2,3
Practical 3	Array implementation of list using C program	2	Understand the basics of array implementation	2
Practical 4	Linked list implementation of list using C program [singly linked list]	2	Understand the basics of linked list	2,3
Practical 5	Linked list implementation of stack using C program	2	Understand and implement stack using linked list	2,3
Practical 6	Linked list implementation of queue using C program	2	Understand and implement queue using linked list	2,3
Practical 7	Applications of list polynomialAddition and subtraction using C program	2	Understand and apply polynomial addition and subtraction	4
Practical 8	Infix to postfix using C program	2	Understand and convert infix to postfix using C program.	3
Practical 9	Expression evaluation using C program	2	Evaluate expression using C program.	3,4
Practical 10	Implementation of avl trees using C program	2	Understand AVL tree and implement using array.	3
Practical 11	Implementation of heap using priority Queues using C program	2	Develop heap using priority queue.	3,4
Practical 12	Representation of graph using C program	2	Understand graph and it representation.	3
Practical 13	Graph traversal-breadth first traversal using C program	2	Understand the concepts of traversal breadth first traversal.	3
Practical 14	Graph traversal-depth first Traversalusing C program	2	Understand and apply the concepts of traversal depth first traversal.	3
Practical 15	Linear search using C program	2	Understand and apply linear search	2,3, 4
Practical 16	Binary search using C program	2	Understand and apply binary search	2,3, 4

Practical 17	Insertion sort using C program	2	Implement and apply Insertion sort.	3,4
Practical 18	Bubble sort using C program	2	Implement and apply Bubble sort.	3,4
Practical 19	Quick sort using C program	2	Implement and apply Quick sort.	3,4
Practical 20	Merge sort using C program	2	Implement and apply Merge sort.	3,4

TEXT BOOKS:

T1: "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, SartajSahni, Computer Science Press.

REFERENCE BOOKS:

R1: Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company

R2: "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate the Basic concepts of Data Structures.	1,2,3
2	Apply Data Structure techniques on computing problem.	1,2,3
3	Analyse and develop algorithms to solve real world problems.	1,2,3
4	Implement and developed program for various concepts of data structures including array, stack, queue, graphs and trees.	1,2,3
5	Demonstrate and analyze various sorting algorithms and hashing techniques.	1,2,3,8

SEMESTER – II									
Course Title	**Computer Organization & Architecture**								
Course code	**24BCAO12012R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. Discuss the organization and architecture of computer systems and electronic computers. 2. Analyse the basic components of computer systems besides the computer arithmetic. 3. Discuss input-output organization, memory organization and management, and pipelining.								
CO1	Outline the basics of instructions sets and their impact on processor design								
CO2	Examine the control unit design approaches, memory design technologies and I/O transfers.								
CO3	Explain the concepts of pipelining in Computer Architecture.								
CO4	Interpret and analyse Parallel Processing Principles and Applications.								
CO5	Demonstrate the concepts of Memory Organization Through Mapping Functions and Replacement Algorithms.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU – registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs.	**9**	Infer the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes	1,2,3
II	Data representation: signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-andadd, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic. Introduction to x86 architecture.	**10**	Explain the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers. Examine the control unit design approaches, memory design technologies and I/O transfers. Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.	3,4

III	CPU control unit design: hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU. Memory system design: semiconductor memory technologies, memory organization Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB	**10**	Explain the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory. Summarize the concepts of memory organization with mapping functions and replacement algorithms	3,4
IV	Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards. Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.	**8**	Summarize the concepts of memory organization with mapping functions and replacement algorithms.	4
V	Memory organization: Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.	**8**	Summarize the concepts of memory organization with mapping functions and replacement algorithms	1,2

TEXT BOOKS:

T1: "Computer Organization and Design: The Hardware/Software Interface", 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.

REFERENCE BOOKS:

R1: "Computer Organization and Embedded Systems", 6th Edition by Carl Hamacher, McGraw Hill Higher Education.

R2: "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill.

R3: "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Outline the basics of instructions sets and their impact on processor design	3,5,6,8
2	Examine the control unit design approaches, memory design technologies and I/O transfers.	2,3,5,6,8
3	Explain the concepts of pipelining in Computer Architecture.	2,3,5,6,7,8
4	Interpret and analyze Parallel Processing Principles and Applications.	2,3,5,7,8
5	Demonstrate the concepts of Memory Organization Through Mapping Functions and Replacement Algorithms.	2,3,7,8

SEMESTER – II									
Course Title	**Introduction to Web Technology**								
Course code	**24BCAO1203R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. To understand the Fundamental principle of Web design, hosting and maintenance. 2. To understand tools required for Web design. 3. To understand server and client.								
CO1	Illustrate elements and attributes of a web page.								
CO2	Build web pages using HTML and Cascading Style Sheets								
CO3	Develop XML documents and Schemas								
CO4	Design and implement static and dynamic website								
CO5	Analyse best technologies for solving web client/server problems								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Basics of Internet:** Client/Server Computing: What is C/S Computing, Middleware, Fat client VS Fat Servers, N-tiered Software Architecture. Web's Robot global access to information, HTML, HTTP, IP address, DNS & ports; Accessing a web server, publishing on web server, secure HTTP, Secure Socket layer, WWW Proxies, IIS, Case study of Apache web server.	**10**	Illustrate elements and attributes of a web page.	1,2,3
II	**Searching and web casting Technique:** Popular web servers, basic feature; bookmarks, cookies, progress indicators, customization of browsers, browsing tricks, next generation web browsing, search engines, architecture of search engines, search tools, web crawler, focused crawler, agent based crawler, case study of any Web Browser.	**12**	Build web pages using HTML and Cascading Style Sheets Develop XML documents and Schemas Design and implement static and dynamic website	3,4
III	**Web page design:** Designing web pages with HTML- use of tags, hyperlinks, URLs, tables, text formatting, graphics & multimedia, imagemap, frames and forms in web pages. Use of Cascading Style Sheet in web pages. Introduction to DHTML.	**8**	Build web pages using HTML and Cascading Style Sheets. Design and implement static and dynamic website. Analyse best technologies for solving web client/server problems	3,4

IV	**Creating interactive and dynamic web pages with JavaScript:** JavaScript overview; constants, variables, operators, expressions & statements; user-defined & built-in functions; client-side form validation; using properties and methods of built-in objects. **Extensible Markup Language (XML):** Introduction- using user-defined tags in web pages; displaying XML contents; XML DTDs; use of XSL.	**8**	Analyse best technologies for solving web client/server problems	4
V	**Web Server:** Web services and web server functionality; web server composition; registration; conceptual architecture of some typical web servers. Server-side scripting: overview of server-side scripts like CGI, ASP, and JSP.Server side scripting using PHP; Web database connectivity-introduction to ODBC; PHP with database connectivity. **Exposure to Advanced Web Technologies:** Distributed Object based models- DCOM, CORBA, EJB; Web services and Related Technologies- ISAPI, SOAP, UDDI, WSDL; Other Advanced Web Technologies- AJAX, ISAPI, .NET.	**7**	Analyse best technologies for solving web client/server problems	4
Practical Component				
Practical 1	Write a HTML program for the demonstration of Unordered Lists.	2	Learn to create Unordered Lists	2,3
Practical 2	Write a HTML program for the demonstration of ordered Lists	2	Learn to create Ordered Lists	2,3
Practical 3	Write a HTML program for the demonstration of Definition Lists	2	Learn to create Definition Lists	2
Practical 4	Write a HTML program for the demonstration of Nested Lists	2	Learn to create Nested Lists Lists	2,3
Practical 5	Write a HTML program for demonstrating Hyperlinks. Navigation from one page to another and Navigation within the page.	2	Understand hyper link and implement hyperlink navigation from one page to another and within the page	2,3
Practical 6	Write a HTML program for time-table using tables.	2	Understand and implement queue using linked list	2,3
Practical 7	Write a HTML program to develop a static Home Page using frames.	2	Understand and apply polynomial addition and subtraction	4
Practical 8	Write a HTML program to develop a static Registration Form	2	Understand and convert infix to postfix using C program.	3
Practical 9	Write a HTML program to develop a static Login Page.	2	Evaluate expression using C program.	3
Practical 10	Write a HTML program to develop a static Web Page for Catalog.	2	Understand AVL tree and implement using array.	3
Practical 11	Write a HTML program to develop a static Web Page for Shopping Cart.	2	Develop heap using priority queue.	3

Practical 12	Write HTML for demonstration of cascading Embedded stylesheets.	2	Understand graph and it representation.	3
Practical 13	Write HTML for demonstration of cascading External stylesheets	2	Understand the concepts of traversal breadth first traversal.	3
Practical 14	Write HTML for demonstration of cascading Inline styles.	2	Understand and apply the concepts of traversal depth first traversal.	3
Practical 15	Write a JavaScript program to validate USER LOGIN page.	2	Develop a user login page	3
Practical 16	Write a program for implementing XML document for CUSTOMER DETAILS.	2	Develop skills in creating well-formed XML documents.	3
Practical 17	Write an internal Document Type Definition to validate XML for CUSTOMER DETAILS	2	Learn how to create an internal DTD to define the structure and constraints of an XML document.	3
Practical 18	Write an external Document Type Definition to validate XML for CUSTOMER DETAILS	2	Develop skills in managing and maintaining DTD files for multiple XML documents.	3
Practical 19	Write a simple servlet that displays a message	2	Develop skills in handling HTTP requests and generating HTTP responses.	3
Practical 20	Write a servlet that reads parameters from employee login page	2	Understand how to read and process parameters from HTTP requests in a servlet.	3

TEXT BOOKS:

T1: C. Xavier, "Web Technology & Design", New Age Publication, 2003

T2: Austin and Pawlan, "Advanced Programming for JAVA2 Platform", Pearson, 2000

REFERENCE BOOKS:

R1: Oliver, Dick; SAMS Teach Yourself Html 4 in 24 Hours; Techmedia.

R2: Ashbacher, Charles; SAMS Teach Yourself XML in 24 Hours; Techmedia.

OTHER LEARNING RESOURCES:

- https://books.google.co.in/books/about/Data_Structures_and_Algorithms.html?id=1lCHYj5eV-EC&redir_esc=y
- https://techdevguide.withgoogle.com/paths/data-structures-and-algorithms/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Illustrate elements and attributes of a web page.	1,2,3,4,8
2	Build web pages using HTML and Cascading Style Sheets	1,2,3,4,7,8
3	Develop XML documents and Schemas	1,2,3,4,7,8
4	Design and implement static and dynamic website	1,2,3,4,7,8
5	Analyse best technologies for solving web client/server problems	1,2,3,4,7,8

SEMESTER – II									
Course Title	**Environmental Science**								
Course code	**24UBES1001R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. To understand the fundamental concepts and principles of environmental science. 2. To analyse the impact of human activities on the environment. 3. To explore ecological processes and the importance of biodiversity.								
CO1	Demonstrate the multidisciplinary nature of environmental studies								
CO2	Learning about natural resource, its importance and environmental impacts of Human activities on natural resource								
CO3	Explain environment and ecosystem, concept of biodiversity.								
CO4	Aware students about problems of environmental pollution, its impact and control measures.								
CO5	Illustrate the human population and the environment								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Multidisciplinary nature of environmental studies:** Definition, scope and importance Need for public awareness.	**5**	Understand basic ecological concepts and environmental processes.	**1,2**
II	**Natural Resources:** **Renewable and non-renewable resources:** Natural resources and associated problems. Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies.	**6**	Identify and analyze the impact of human activities on the environment. Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them.	**2,3,4**
	Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.			

III	**Ecosystems:** Concept of an ecosystem. Structure and function of an ecosystem. Producers, consumers and decomposers. Energy flow in the ecosystem. Ecological succession. Food chains, food webs and ecological pyramids. Introduction, types, characteristic features, structure and function of the Following ecosystem: - Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)	**7**	Identify and analyze the impact of human activities on the environment. Gain knowledge about environment and ecosystem	**3,4**
IV	**Biodiversity and its conservation** Introduction – Definition: genetic, species and ecosystem diversity. Biogeographical classification of India. Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. Biodiversity at global, National and local levels. India as a mega diversity nation. Hot-sports of biodiversity. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. Endangered and endemic species of India. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.	**6**	Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them Gain knowledge about the conservation of biodiversity and its importance. Aware students about problems of environmental pollution, its impact on human and ecosystem and control measures	**4,5**
V	**The Interconnection Between Environmental Pollution and Social Issues: Impacts and Solutions:** Definition Cause, effects and control measures of:-Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards. Solid waste Management: Causes, effects and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies. Disaster management: floods, earthquake, cyclone and landslides.	**6**	Gain knowledge about environment and ecosystem. Understand the concept of biodiversity and respect them Gain knowledge about the conservation of biodiversity and its importance. Aware students about problems of environmental pollution, its impact on human and ecosystem and control measures	**4,5**

	Social Issues and the Environment: From Unsustainable to Sustainable development. Urban problems related to energy. Water conservation, rain water harvesting, watershed management. Resettlement and rehabilitation of people; its problems and concerns. Case Studies. Environmental ethics : Issues and possible solutions. Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies. Waste land reclamation. Consumerism and waste products. Environment Protection Act. Air (Prevention and Control of Pollution) Act.Water (Prevention and control of Pollution) Act. Wildlife Protection Act. Forest Conservation Act. Issues involved in enforcement of environmental legislation. Public awareness.			

TEXT BOOKS:

T1: Harucha E. B, Textbook of Environmental Studies, Orient Blackswan Publishing.
T2: Tiwari V. K A Textbook of Environmental Studies, Himalaya Publishing House
T3: Chatwal G. R. &Sharma H. Environmental Studies, Himalaya Publishing House

REFERENCE BOOKS:

R1: Trivedi R.K., Handbook of Environmental Laws, Rules Guidelines, Compliances and
R2: Stadards, Vol I and II, Enviro Media (R)
R3: Trivedi R. K. and P.K. Goel, Introduction to air pollution, Techno-Science Publication (TB)
R4: Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner. BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email:mapin@icenet.net (R)
R5: Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p Clark R.S., Marine Pollution, Clanderson Press Oxford (TB)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Demonstrate the multidisciplinary nature of environmental studies	4
2	Learning about natural resource, its importance and environmental impacts of Human activities on natural resource	3,4
3	Explain environment and ecosystem, concept of biodiversity.	3,4
4	Aware students about problems of environmental pollution, its impact and control measures.	4,5,6,7
5	Illustrate the human population and the environment	4,8

SEMESTER – II									
Course Title	**ADVANCE IMPLICIT ENGLISH**								
Course code	**24UBPD1203R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 60P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. To enable students to learn, understand and practice transformation of sentences, correct usage of all tenses and rectify common grammatical errors. 2. To help students to expand their Vocabulary strength along with learning new words and collocations. 3. To train and guide students to improve and their sharpen their listening skill and to become good listeners,								
CO1	Enable students to understand grammar to write effectively and speak flawlessly, knowing correct usage of tenses and rectifying grammatical errors.								
CO2	Students will acquire the technique of expanding vocabulary and creating new words and use them in different contexts. Encouraging and helping them to sharpen their listening skills and to become good listeners.								
CO3	Encouraging and helping them to sharpen their listening skills and to become good listeners. Students will be able to develop and showcase their effective reading skills while reading any texts.								
CO4	Helping students to acquire the skill of time managing and apply the same in their profession lives along with the skills and information required for creating an excellent LinkedIn Profile.								
CO5	To make them prepare for various public and private sector exams & placement drives. To enhance the analytical skill and problem-solving skill of the students								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar (flipped classroom)** 1. Interchange of Interrogative and Assertive Sentences, Exclamatory and Assertive Sentences 2. Types of Tenses 3. Common Errors 4. Non Verbal reasoning 5. Counting figures 6. Water images and mirror image 7. Clock images 8. Worksheet1 and Worksheet 2	12	Enrich in vocabulary.	**1,2,3**
II	**Vocabulary Development** 1. One word substitution 2. Homonyms and Homophones 3. Words often confused 4. Idioms and phrases 5. V.Average and Age 6. Average of numbers objects things and quantities 7. Problem on age 8. Solving questions on age and average	12	Identify common errors in English speaking and writing	**3,4**
III	**Listening Skills** 1. What is listening? 2. Types of Listening 3. Understanding Listening Barriers 4. Missing number/Letter 5. Letter Series(advanced) 6. Number series(advanced)	12	Improve listening skills.	**3,4**

IV	**Reading Skills** 1. Techniques of Effective Reading 2. Gathering ideas and information from a text 3. The SQ3R Technique 4. Venn diagram **5.** Introduction to venn diagram	10	Learn effective reading techniques.	**4**
V	**Time-Management Skills** 1. Introduction to Time Management 2. Purpose and Importance of Time Management 3. Basic Tips to maintain time 4. Classification a. Classification of odd one out. Number, meaningful word b. Classification based on priority	14	Learn time management techniques.	**2,4**

TEXTBOOKS:

T1: Barrett, Grant.2016.*Perfect English Grammar: The Indispensible Guide to Excellent Writing and Speaking*, Zephyros Press.

T2: McDowell, Gayle Laakmann. 2008. *Cracking the Coding Interview* (Indian Edition)

REFERENCE BOOKS:

R1: Zinsser, William.(2006)*On Writing Well: The Classic Guide to Writing Nonfiction* HarperPerennial

R2: Taylor J. and Wright, J., *IELTS Advantage Reading Skills: A step-by-step guide to a high IELTS reading score*, Delta Publishing by Klett.

R3: Murphy, Raymond,.(2012)English Grammar in Use Book with Answers: A Self- Study and Practice Book for Intermediate Learners of English ,Cambridge University Press

R4: Quantitative Aptitude and Logical Reasoning by R.S. Agarwal

R5: Quantitative Aptitude and Logical Reasoning by Arihant

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOME

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enable students to understand grammar to write effectively and speak flawlessly, knowing correct usage of tenses and rectifying grammatical errors.	5,6,8
2	Students will acquire the technique of expanding vocabulary and creating new words and use them in different contexts. Encouraging and helping them to sharpen their listening skills and to become good listeners.	5,6,8
3	Encouraging and helping them to sharpen their listening skills and to become good listeners. Students will be able to develop and showcase their effective reading skills while reading any texts.	5,6,8
4	Helping students to acquire the skill of time managing and apply the same in their profession lives along with the skills and information required for creating an excellent LinkedIn Profile.	5,6,7,8
5	To make them prepare for various public and private sector exams & placement drives. To enhance the analytical skill and problem-solving skill of the students	5,7,8

SEMESTER – II									
Course Title	**Mathematics for Basic Programming**								
Course code	**24BCAO1207R**	**Total credits: 2** **Total hours: 30L**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	0	**0**	**0**	**0**	2
Pre- requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of 1st year of the programme**								
Course Objectives	1. To build a strong foundation in mathematics and its application in C programming for problem-solving and algorithm development. 2. To enable students to write efficient programs using arithmetic, logic, control flow, arrays, matrices, and number systems. 3. To develop skills in implementing mathematical solutions for real-world applications, including data processing and optimization tasks.								
CO1	Apply arithmetic operators and mathematical expressions to build programs for real- world applications like calculators and geometry problems.								
CO2	Use Boolean algebra and conditional expressions to implement decision-making and problem-solving programs								
CO3	Implement loops and nested structures to solve mathematical problems, including sequences, prime checking, and matrix patterns.								
CO4	Convert between number systems and optimize programs using bitwise operators and binary manipulations.								
CO5	Use arrays and matrices for complex calculations like summation, searching, and matrix operations to handle and process data effectively.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Arithmetic and Basic Math Operations** Arithmetic Operators: Addition, subtraction, multiplication, division, and modulus. Operator precedence and associativity. Mathematical Expressions: Writing and evaluating mathematical formulas. Practical Applications: Building a basic calculator. Programs for area and perimeter calculations.	**6**	Apply arithmetic operators, operator precedence, and mathematical expressions in programming to build calculators and solve geometry-related programming tasks.	1, 2, 3
II	**Logical and Boolean Operations** Boolean Algebra: Logical operators (&&, ||, !). Truth tables and their use programming. Conditional Expressions: Ternary operator (? :) and nested conditions. Applications: Writing decision-making programs. Programs for logical problem-solving (e.g., eligibility checks).	**6**	Apply Boolean operators, truth tables, and conditional expressions in C to create decision-making programs and solve logical problems effectively.	3
III	**Control Flow with Mathematical Logic** Loops and Iterations: For, while, and do-while loops. Mathematical Applications: Sum of natural numbers, factorial, and Fibonacci sequence. Checking for prime numbers. Nested Loops: Solving matrix	**6**	Implement loops and iterations in C to solve mathematical problems, including sequences, prime checking, matrices, and pattern generation.	2, 3, 4

	and pattern problems.			
IV	**Number Systems and Bitwise Mathematics** Introduction to Number Systems: Binary, octal, hexadecimal, and their conversions. Bitwise Operations: AND, OR, XOR, NOT, left shift, right shift. Applications: Binary representation of numbers. Bitwise manipulation for optimization.	**6**	Understand number systems and implement bitwise operations in C for conversions, binary representation, and optimization of programming tasks.	2, 3, 4
V	**Arrays and Matrices** Mathematical Use of Arrays: Single-dimensional and multi-dimensional arrays. Summation, average, and searching in arrays. Matrices: Matrix addition, subtraction, and multiplication. Practical Applications: Creating programs for data storage and retrieval.	**6**	Utilize arrays and matrices in C for mathematical operations, including summation, searching, and matrix computations, to manage data effectively.	2, 3, 4

TEXT BOOKS:

T1: "Let Us C" by Yashavant Kanetkar

T2: "Discrete Mathematics and its Applications" by Kenneth H. Rosen

REFERENCE BOOKS:

R1: "The C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie R2:

"Mathematical Logic for Computer Science" by Mordechai Ben-Ari

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply arithmetic operators and mathematical expressions to build programs for real-world applications like calculators and geometry problems.	1, 5
2	Use Boolean algebra and conditional expressions to implement decision-making and problem-solving programs	2, 3
3	Implement loops and nested structures to solve mathematical problems, including sequences, prime checking, and matrix patterns.	1, 3
4	Convert between number systems and optimize programs using bitwise operators and binary manipulations.	1, 4
5	Use arrays and matrices for complex calculations like summation, searching, and matrix operations to handle and process data effectively.	3, 5

SEMESTER – II									
Course Title	**Basics of Graph Theory**								
Course code	**24BCAO1206R**	**Total credits: 2 Total hours:**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master in Computer Application**								
Semester	**Fall/ II semester of the First year of the program**								
Course Objectives (Minimum 3)	1. To introduce students to the foundational concepts of graph theory and their applications in various domains. 2. To enable students to model and solve real-life problems using graph theory, focusing on graph properties and algorithms. 3. To develop the ability to analyze and apply different types of graphs to practical scenarios.								
CO1	Model problems using different types of basic graphs like trees, bipartite graphs, and planar graphs.								
CO2	Understand and identify special graphs such as Eulerian and Hamiltonian graphs.								
CO3	Analyze various forms of connectedness in a graph and their implications in solving problems.								
CO4	Apply graph coloring techniques to solve problems and understand their theoretical aspects.								
CO5	Model and analyze real-life problems as graph problems using the concepts learned.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Unit I: Fundamental Concepts** **Content:** Introduction to Graph Theory: Basic Concepts, Definitions, Graph Types Adjacency and incidence matrices, Isomorphisms, paths, walks, cycles, components Cut-edges, cut-vertices, bipartite graphs, Eulerian and Hamiltonian graphs, vertex degrees, Reconstruction conjecture, extremal problems, degree sequences, Directed graphs, Orientations, and tournaments.	**7**	Remembering and understanding basic graph concepts and applying them to different problems..	**1, 2, 3**
II	**Unit II: Trees** **Content:** Trees and forests, characterizations of trees, Spanning trees, radius and diameter, Enumeration of trees, Counting spanning trees, Deletion-contraction, The matrix tree theorem, Minimum spanning trees Prim's and Kruskal's algorithm Single source shortest paths Bellman Ford, Single source shortest paths Dijkstra's algorithm	**7**	Applying and analyzing tree structures in graph theory.	**1, 2, 3, 4**

III	**Unit III: Matching and Covers** **Content:** Matchings, maximal and maximum matchings M-augmenting paths, Hall's theorem and consequences, Min-max theorems, Maximum matchings and vertex covers, Independent sets and edge covers, Connectivity, vertex cuts Edge-connectivity	**8**	Applying and analyzing different types of matchings and covers in graphs.	**3, 4**
IV	**Unit IV: Connectivity and Paths** **Content:** Cuts and Edges connectivity, Blocks, k-connected graphs, Menger's theorem, Network flow problems, Flows and source/sink cuts, Ford-Fulkerson algorithm, Maximum Network Flow	**7**	Describing and analyzing connectivity and path problems in graphs.	**2, 4**
V	**Unit V: Graph Coloring** **Content:** Vertex colorings, Upper Bounds, Brooks Theorem, Bounds on chromatic numbers, Chromatic numbers of graphs constructed from smaller graphs, Chromatic polynomials, Properties of the chromatic polynomial, Chordal Graphs and A Hint of perfect Graphs	**8**	Understanding and analyzing graph coloring techniques and their applications.	**1, 2, 4**
VI	**Unit VI: Planar Graphs** **Content:** Planar graphs, Euler's formula, Kuratowski's theorem, Five and four-color theorems, Line Graph and Edge coloring, Hamiltonian Cycle, Planarity, coloring and Cycles.	**8**	Understanding, analyzing, and applying concepts of planar graphs.	**1, 2, 4**

TEXT BOOKS:

T1: Douglas B West, Introduction to Graph Theory, II Edition, 2017, Pearson.

REFERENCE BOOKS:

R1: Gary Chartrand and Ping Zhang, Introduction to Graph Theory, 2017, Tata McGraw-Hill.

R2: Jonathan L. Gross and Jay Yellen, Graph Theory and Its Applications, 2nd Edition, 2005, Chapman Hall (CRC).

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Model problems using different types of basic graphs like trees, bipartite graphs, and planar graphs.	1 and 3
2	Understand and identify special graphs like Eulerian and Hamiltonian graphs.	2, 3, and 4
3	Analyze various forms of connectedness in a graph and their implications in solving problems.	1, 2, and 7
4	Apply graph-coloring problems and solutions to practical situations.	1, 2, 4, and 7
5	Utilize algorithms and methods in graph theory to solve real-world problems.	1, 2, 5, and 7

SEMESTER – II									
Course Title	**Extra-Curricular Activity**								
Course code	**24UBEC1201**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ II semester of the first year of the program**								
Course Objectives	1. To develop problem-solving abilities through coding challenges and hackathons. 2. To participate in app development contests and exhibitions. 3. To develop entrepreneurial skills and mindset.								
CO1	Importance of different activities under different clubs.								
CO2	Formulate regular activities like workshops, competitions as per their interest and hobbies.								
CO3	Adapt to represent ADTU in various inter university, state and national level competitions.								
CO4	Discuss with invited experts in their respective fields.								
CO5	Develop an appreciation for diverse forms of artistic expression.								
Unit-No.	**Content**		**Contact Hour**		**Learning Outcome**				**KL**
I	AdtU encourages a range of activities outside the regular curriculum intended to meet learner's interest, These activities are aimed to develop the social and soft skills and promote a holistic development of the learners, Keeping in mind the 360 degree learning methodology the students are engaged in different activities headed under different clubs viz. Dance, music, photography, drama, literary etc., The students are encouraged to participate in regular club activities, workshops, competitions as per their interest and hobbies, The student members of the club are trained represent AdtU in various inter University student and national level competitions, Renewed personalities are invited to conduct workshops that benefit the members and students by giving them the platform to learn from experts.		**2**		Demonstrate improved proficiency in programming, software development, and the use of various technological tools and platforms. Apply theoretical knowledge to real-world problems through projects, internships, and industry collaborations.				**1,2,3,4**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students engagement in different activities under different clubs.	2,3,4,5,6,7
2	Role play in regular activities like workshops, competitions as per their interest and hobbies.	2,3,4,5,6,7
3	Adapt and trained to represent ADTU in various inter university, state and national level competitions.	2,3,4,5,6,7,8
4	The students will be given a platform to earn from invited experts in their respective fields.	2,3,4,5,6,7,8
5	Develop an appreciation for diverse forms of artistic expression.	2,3,5,6,7,8

SEMESTER – II									
Course Title	**MINI PROJECT-II**								
Course code	**24BCAO1205**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: Perform a literature search to review current knowledge and developments in the chosen technical area; Undertake detailed technical work in the chosen area using one or more of: ● theoretical studies ● computer simulations ● hardware construction; Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; Deliver a seminar on the general area of work being undertaken and specific contributions to that field; Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	3, 4, 5

OTHER LEARNING RESOURCES:

As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

Detailed Syllabus of Third Semester

SEMESTER – III									
Course Title	**Database Management Systems**								
Course code	**24BCAO2101R**	**Total credits: 4** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives	1. To master core database concepts and SQL for efficient database design and implementation. 2. To explore advanced database features including NoSQL, data warehousing, and security. 3. To apply theoretical knowledge to real-world database management challenges through hands-on projects.								
CO1	Gain an understanding of fundamental database concepts and architectures.								
CO2	Develop proficiency in SQL and relational database design.								
CO3	Master advanced SQL features and database optimization techniques.								
CO4	Learn to design and manage solutions for unstructured data with NoSQL.								
CO5	Apply database knowledge to a real-world data analytics project.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Database Systems:** Overview of Database Systems: Definition and importance of databases, Evolution of DBMS Types of Database Systems: Relational, NoSQL, NewSQL databases Data models: Entity-Relationship diagram, Relational model, Database schema vs. data Database System Architecture: Roles of database users and administrators, DBMS architecture and data independence	**8**	Understand the definition, importance, and evolution of database systems; differentiate between relational, NoSQL, and NewSQL databases; comprehend data models including Entity-Relationship diagrams and relational models; and explain the roles of database users and administrators, DBMS architecture, and data independence.	2,4,5
II	**Relational Database Management:** SQL and Data Manipulation: Basic to advanced SQL, including queries, updates, and management Database Design: Normalization (1NF to BCNF), Integrity constraints, Indexing, and access methods Transaction Management: ACID properties, concurrency control, deadlock prevention	**10**	Utilize SQL for data manipulation, design normalized databases with integrity constraints and indexing, and manage transactions ensuring ACID properties, concurrency control, and deadlock prevention.	3,4,5
III	**Advanced Database Features:** Advanced SQL and Stored Procedures: Incorporating triggers, views, and procedures Database Security: Implementing data security measures, encryption, authorization, and authentication Performance Tuning and Optimization: Techniques for query optimization and database tuning	**8**	Apply advanced SQL features, implement database security measures, and optimize database performance through various tuning techniques.	3,4,5

IV	**Unstructured Data Management:** Introduction to Unstructured Data: Understanding the significance and management challenges NoSQL Databases: Types and applications: Document stores, key-value stores, wide-column stores, graph databases Data Modelling for NoSQL: Approaches to schema design, integration in NoSQL environments	**6**	Understand the significance and challenges of managing unstructured data, explore various types and applications of NoSQL databases, and learn approaches to schema design and integration in NoSQL environments.	2,3,4
V	**NoSQL Databases and Project-Based Applications:** MongoDB: Specifics of working with MongoDB, data model, query language, and tools for data manipulation and administration Practical Use Cases of NoSQL Databases: real-world applications and case studies that effectively utilize NoSQL databases to handle scalability, performance, and flexibility Project Design and Implementation Project Presentation and Evaluation	**5**	Understanding of MongoDB and NoSQL databases, analyze real-world applications, design and implement database projects, and effectively present and evaluate their work.	1,2,4, 5
Practical Component				
Practical 1	**Consider the bank database given, where the primary keys are underlined. Construct the following SQL queries for this relational database.** a. Find the ID of each customer of the bank who has an account but not a loan. b. Find the ID of each customer who lives on the same street and in the same city as customer '12345'. c. Find the name of each branch that has at least one customer who has an account in the bank and who lives in “Harrison”.	**4**	Describe, illustrate, explain, apply and develop DBMS Programming using commands.	1,2,3, 4
Practical 2	**Create the below tables with the given attributes and enter some records.** a. Write a query to find the total number of marks obtained by each student. b. Write a query that gives the total marks obtained by Mohan. c. Write a query that displays the maximum marks of every subject. d. Write the SQL query which displays the name of those students who’s total of all subjects is greater than 205. e. Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2. Pre-Experiment Questions 1. What is the use of sub Queries? 2. Explain different types of Join. Post Experiment Questions 1. How to alter Primary Key?	**4**	Design and Create Database Tables, Write SQL Queries for Data Retrieval, Utilize SQL Functions, Implement Conditional Query Logic, Filter Data Based on Multiple Conditions, Understand and utilize subqueries, Alter table structures, Update data with constraints	2,3

	2. How to update a table by enforcing constraint?			
Practical 3	**Create the table as given below, fill in the data.** Question: Which normal form in the table? After creating the table, create a second table (s) to convert the same into 2NF!	**4**	Understand Database Normalization, Design and Create Database Tables, Identify Normal Forms,4Transform Tables to Achieve Higher Normal Forms, Implement Referential Integrity, Enhance Data Integrity and Reduce Redundancy	2,3
Practical 4	**Write the SQL queries using Group by and having clause using following table.** Create the above tables with the given attributes and enter some records. I. Write a query to find the total number of marks obtained by each student. II. Write a query that gives the total marks obtained by Mohan. III. Write a query that displays the maximum marks of every subject. IV. Write the SQL query which displays the name of those students who's total of all subjects is greater than 205. V. Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2.	**4**	Understanding Table Creation and Data Insertion, Querying and Summarizing Data, Using Aggregate Functions, Filtering Grouped Data	1,2,3
Practical 5	**Create a table, fill it with customer details and do the following queries:** a. Find the total revenue for each customer b. List customers who have made more than one order c. Find the date of the most recent order for each customer d. Calculate the average order amount for each customer e. Retrieve the orders placed in January 2023 f. Find the top 3 customers with the highest total revenue g. Calculate the total revenue for each month in 2023 h. List orders with a total amount greater than the average total amount for all orders i. Find customers who have made orders on consecutive days j. Calculate the total revenue for each customer, including orders made by customers who haven't placed any orders	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders, Identifying High-Value Customers	3,4
Practical 6	(Exercise on retrieving records from the table) EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number,	**4**	Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving	

	Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id) (a) Find out the employee id, names, salaries of all the employees (b) List out the employees who works under manager 100 (c) Find the names of the employees who have a salary greater than or equal to 4800 (d) List out the employees whose last name is 'AUSTIN' (e) Find the names of the employees who works in departments 60,70 and 80 (f) Display the unique Manager_Id		Specific Orders, Identifying High-Value Customers	
Practical 7	(Exercise on updating records in table) Create Client_master table with the following fields (ClientNO, Name, Address, City, State, bal_due) (a) Insert five records (b) Find the names of clients whose bal_due>5000. (c) Change the bal_due of Client NO "C123" to Rs. 5100 (d) Change the name of Client_master to Client12. (e) Display the bal_due heading as "BALANCE"	4	Table Creation and Data Insertion, Summarizingcustomer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders	3,4
Practical 8	Rollback and Commit commands Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary) (a) Insert five records (b) Give Increment of 25% salary for Mathematics Department . (c) Perform Rollback command (d) Give Increment of 15% salary for Commerce Department (e) Perform commit command	**4**	Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders, using roll back and commit command	2,3,4
Practical 9	(Exercise on order by and group by clauses) Create Sales table with the following fields(Sales No, Salesname, Branch, Salesamount, DOB) (a) Insert five records (b) Calculate total sales amount in each branch (c) Calculate average sales amount in each branch . (d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09 (e) Display the name and DOB of	**4**	Creation and Insertion of records, use of group by clouse	2,3,4

	salesman in alphabetical order of the month.			
Practical 10	Create an Emp table with the following fields: (EmpNo, EmpName, Job,Basic, DA, HRA,PF, GrossPay, NetPay) (Calculate DA as 30% of Basic and HRA as 40% of Basic) (a) Insert Five Records and calculate GrossPay and NetPay. (b) Display the employees whose Basic is lowest in each department . (c) If NetPay is less than <Rs. 10,000 add Rs. 1200 as special allowances . (d) Display the employees whose GrossPay lies between 10,000 & 20,000 (e) Display all the employees who earn maximum salary	**4**	Database Design and Table Creation, Data Manipulation and Retrieval, **Querying Data with Conditions,** Aggregate Functions and Grouping, Updating Data,	2,3,4
Practical 11	Create a table, fill it with customer details and do the following queries: a) Find the total revenue for each customer b) List customers who have made more than one order c) Find the date of the most recent order for each customer d) Calculate the average order amount for each customer e) Retrieve the orders placed in January 2023 f) Find the top 3 customers with the highest total revenue g) Calculate the total revenue for each month in 2023 h) List orders with a total amount greater than the average total amount for all orders i) Find customers who have made orders on consecutive days j) Calculate the total revenue for each customer, including orders made by customers who haven't placed any orders.	**4**	Table Creation and Data Insertion, summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders.	
Practical 12	Employee Database An Enterprise wishes to maintain a database to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas Dept (deptno, dname, loc) Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno) a) Update the employee salary by 15%, whose experience is greater than 10 years. b) Delete the employees, who completed 30 years of service.	**4**	Database Design and Table Creation, Data Manipulation and Retrieval, **Querying Data with Conditions,** Aggregate Functions and Grouping, Updating Data,	2,3,4, 5

	c) Display the manager who is having maximum number of employees working under him? d) Create a view, which contain employee names and their manager			
Practical 13	Using Employee Database perform the following queries a) Determine the names of employee, who earn more than their managers. b) Determine the names of employees, who take highest salary in their departments. c) Determine the employees, who are located at the same place. d) Determine the employees, whose total salary is like the minimum Salary of any department. e) Determine the department which does not contain any employees.	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders.	2,3,4, 5
Practical 14	11. Write the SQL queries using Group by and having clause using following table. Create the above tables with the given attributes and enter some records. a) Write a query to find the total number of marks obtained by each student. b) Write a query that gives the total marks obtained by Mohan. c) Write a query that displays the maximum marks of every subject. d) Write the SQL query which displays the name of those students who's total of all subjects is greater than 205. e) Display name of the students whose subjects total is greater than 205 and roll number is between 1 to 2.	**4**	Design and Create Database Tables, Write SQL Queries for Data Retrieval, Utilize SQL Functions, Implement group by clause Logic, Filter Data Based on Multiple Conditions, Understand and utilize subqueries, display data	2,3,4, 5
Practical 15	Create the following tables : Book(accession-no, title, publisher, year, date-of-purchase, status) Member(member-id, name, number-of-books-issued, max-limit) Book-issue(accession-no, member-id, date-of-issue) (a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following : (i) List all those books which are due from the students to be returned. A book is considered to be due if it has been issued 15 days back and yet not returned. (ii) List all those members who cannot be issued any more books.	**4**	Table Creation and Data Insertion, Summarizing customer revenue, Analyging ordering data, Filtering and Retrieving Specific Orders	2,3,4, 5
Practical 16	You are given a dataset containing information about various books in JSON format. Each book has attributes such as	**2**	Connect to a MongoDB database using Python, insert a JSON dataset into a	3,5

	title, author, published_year, genres, and ISBN. Write a Python script to: 1. Connect to a MongoDB database. 2. Insert the dataset into a collection called books. 3. Query the database to find all books published after the year 2000 and print their titles and authors.		MongoDB collection, and perform queries to retrieve specific data based on given criteria. This exercise enhances their understanding of database operations and data manipulation using MongoDB.	
Practical 17	Using Redis as a key-value store, write a Python script to perform the following tasks: 1. Connect to a Redis server. 2. Store user session data with keys as session_id and values as JSON objects containing user_id, login_time, and status. 3. Retrieve and print the session data for a given session_id.	**2**	Connect to a Redis server using Python, store and manage user session data in a key-value format, and retrieve session data for a given session ID.	3,5
Practical 18	You are tasked with creating a data model in Cassandra to store user activity logs. Each log entry should contain user_id, activity_type, timestamp, and details. Write a Python script to: 1. Connect to a Cassandra cluster. 2. Create a keyspace called user_activity. 3. Create a table called activity_logs with appropriate columns and data types. 4. Insert sample data into the table. 5. Query the table to retrieve all activity logs for a specific user_id.	**2**	Connect to a Cassandra cluster using Python, create a keyspace and table to store user activity logs, insert sample data, and query the table to retrieve logs for a specific user ID.	3,5
Practical 19	You are tasked with managing a MongoDB database for an e-commerce application. The database contains a collection called products with documents that include fields such as product_id, name, category, price, and stock_quantity. Write a Python script to: 1. Connect to the MongoDB database. 2. Insert the following products into the products collection: {"product_id": 1, "name": "Laptop", "category": "Electronics", "price": 1200, "stock_quantity": 30} {"product_id": 2, "name": "Smartphone", "category": "Electronics", "price": 800, "stock_quantity": 50} {"product_id": 3, "name": "Office Chair", "category": "Furniture", "price": 150, "stock_quantity": 20} 3. Write a query to find all products in the "Electronics" category and print their names and prices.	**2**	Establish a connection to a MongoDB database, perform CRUD (Create, Read) operations, and construct queries to filter and retrieve specific data.	2,3,5
Practical 20	You are building a scalable blog platform using MongoDB to manage blog posts.	**2**	Handling large-scale data, performing CRUD	3,4,5

	Each blog post document contains post_id, title, author, content, tags, published_date, and views. Write a Python script to: ● Connect to the MongoDB database. ● Insert the following blog posts into the blog_posts collection: {"post_id": 101, "title": "Introduction to MongoDB", "author": "Alice", "content": "This is a blog post about MongoDB.", "tags": ["MongoDB", "Database"], "published_date": "2024-01-15", "views": 120} {"post_id": 102, "title": "Scaling Applications with NoSQL", "author": "Bob", "content": "This post discusses scaling applications using NoSQL databases.", "tags": ["NoSQL", "Scalability"], "published_date": "2024-02-10", "views": 200} ● Write a query to find all blog posts with more than 150 views and print their titles and authors.		operations, and writing complex queries to analyze and retrieve high-traffic data efficiently.	

Text Books:

T1: Database System Concepts, Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, 7th Edition, 2019

T2: SQL in a Nutshell, Kevin Kline, Brand Hunt, and Daniel Kline, 4th Edition, 2020

T3: NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pramod J. Sadalage and Martin Fowler, 1st Edition, 2012

T4: Designing Data-Intensive Applications, The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, Martin Kleppmann, 1st Edition, 2017

Reference Books:

R1: Fundamentals of Database Systems, RamezElmasri&Shamkant B. Navathe, 8th Edition, 2020

R2: Database Management Systems, Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2002

R3: MongoDB: The Definitive Guide, Kyle Banker and Kristina Chodorow

OTHER LEARNING RESOURCES:

- **https://www.jbiet.edu.in/coursefiles/cse/HO/cse2/DBMSI-III.pdf**
- **https://cs.ulb.ac.be/public/_media/teaching/infoh303/dbmsnotes.pdf**
- **SQLZoo-** https://sqlzoo.net/
- DB-Engines- https://db-engines.com/
- MongoDB - https://www.mongodb.com/online

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basic concepts of database management system.	2, 3, 7, 8
2	Discover the basic fundamentals of creating a database by using data definition, data manipulation and control languages.	2, 3, 7, 8
3	Develop a database application and retrieve the values with the help of queries using SQL.	2, 3, 7, 8
4	Design, analyze and evaluate the project developed for an application.	2, 3, 7, 8
5	Demonstrate the concepts of database security, distributed database systems and object-oriented databases	2, 3, 7, 8

SEMESTER – III									
Course Title	**Object Oriented Programming Paradigm**								
Course code	**24BCAO2102R**	**Total credits: 4** **Total hours:45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Basic Programming**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives	1. To enable students to understand the basic object-oriented programming concepts and apply them in problem solving. 2. To illustrate among students, the inheritance concepts for reusing the program. 3. To enable students to develop GUI-based applications using AWT, Swing, JavaFX and Event handling.								
CO1	Understand object-oriented programming concepts and implement in java.								
CO2	Demonstrate building blocks of OOPs language, inheritance, package and interfaces, and analyze real-world problems in terms of these.								
CO3	Apply the exception handling methods on programming.								
CO4	Develop interactive as well as GUI-based java applications in project-based learning.								
CO5	Outline the concept of package, interface, multithreading and File handling in java.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Java Overview:** Java Introduction, Platform Independence, JVM & JDK, Data types, Operators, If, else statement, Switch condition, while, do-while, for loop, break and continue statement. Array and String: Single Array & Multidimensional Array, Library Classes-String, String Buffer & Wrapper Class, Command line arguments and Various String Operations.	**12**	Understand and apply the fundamentals of Java, including control flow, data structures, and string operations, to write basic Java programs.	2,3
II	**Classes, Objects and Methods:** Class and Object, Object reference, Constructor: Constructor Overloading, Method: Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method, Import statement, Static import, Access control, Nested class, Inner class, Anonymous inner class.	**8**	Develop object-oriented Java applications by creating and manipulating classes, objects, constructors, and methods, including the use of inner classes.	3,5
III	**Inheritance and Interfaces in Java:** Overview of Inheritance, inheritance in constructor, Inheriting Data members and Methods, Multilevel Inheritance – method overriding Handle multilevel constructors super keyword, Stop Inheritance, final keyword. Creation and Implementation of an interface, Interface reference, instance of operator, Interface inheritance, Dynamic method dispatch, Abstract class, Comparison between Abstract Class and	**10**	Implement inheritance and interfaces in Java to design complex class hierarchies, promoting code reuse and flexibility in object-oriented applications.	3,5

	interface, inside of System.out.println – statements.			
IV	**Exception Handling in Java:** Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. **JAVA File Handling:** Overview of Different Stream (Byte Stream, Character stream), Readers and Writers class, File Class, File Input Stream, File Output Stream, Input Stream Reader and Output Stream Writer class, File reader and writer class, File Writer, Buffered Reader class.	**7**	Create robust Java applications by implementing exception handling and efficient file I/O operations using Java's exception and file handling mechanisms.	3,5
V	**Applet, AWT, Swing and JavaFX:** Applet: Applet Fundamental, Applet Architecture, Applet Skeleton, Requesting Repainting , Event Handling: various event handling mechanisms, various classes related to event sources and event listeners, AWT: window fundamentals, creating frames, Adding removing various controls, Layout managers, Introduction To Swing and JavaFX.	**8**	Design and develop graphical user interfaces in Java using AWT, Swing, and JavaFX, incorporating event handling and layout management principles.	5
Practical Component				
Practical 1	Program to print all even numbers between 1 and 50 using for loop.	**2**	Demonstrate the understanding of for loops and conditional statements	2
Practical 2	2. Program to calculate and print factorial of a number 'n' using for loop	**2**	Apply the knowledge of for loops and mathematical concepts	3
Practical 3	3. Program to check two arrays are equal or not.	**2**	Analyze and compare two arrays to determine if they are equal by writing a program that iterates through the arrays and checks for element-wise equality	4
Practical 4	Program to add two matrices.	**2**	Apply the understanding of matrix operations	3
Practical 5	Program to display marks, percentage, grade based on attendance.	**2**	Explain how attendance data impacts marks, percentage, and grades	2
Practical 6	Program to check if a number is odd or even.	**2**	Determine and explain whether a number is odd or even by writing a program that uses conditional statements	2
Practical 7	Program to check if a number is prime or not.	**2**	Determine and explain whether a number is prime by writing a program that tests the number's divisibility by integers other than 1 and itself	2
Practical 8	Program to use switch case to print seasons.	**2**	Apply the knowledge of switch case statements by	3

			writing a program that prints the appropriate season based on a given input	
Practical 9	Program to print Volume of box using Constructor overloading.	**2**	Analyze and differentiate between various constructor implementations by writing a program that uses constructor overloading	4
Practical 10	Program to show function overloading.	**2**	Apply the understanding of function overloading	3
Practical 11	Program to show return by object.	**2**	Apply object-oriented programming concepts by writing a program that returns an object from a function to demonstrate encapsulation and data handling	3
Practical 12	Program to show pass by value.	**2**	Understand and explain the concept of pass by value by writing a program that shows how passing arguments by value affects the function's behaviour	2
Practical 13	Program to show pass by reference.	**2**	Understand and explain the concept of pass by reference by writing a program that shows how passing arguments by reference affects the function's behavior	2
Practical 14	Program to find Factorial using recursive function.	**2**	Apply the understanding of recursion by writing a program that calculates the factorial of a number using a recursive function	3
Practical 15	Program to find Month- season using switch case.	**2**	Apply the knowledge of switch case statements by writing a program that determines and prints the season based on the input month	3
Practical 16	Program to print data of employee.	**2**	Demonstrate their understanding of data structures and object-oriented programming	2
Practical 17	Program to print student data.	**2**	Apply object-oriented programming principles	3
Practical 18	Write a Java program to create a new Box class in Java.	**2**	Synthesize the knowledge of object-oriented programming by designing and implementing a new class in Java	6

TEXT BOOKS:

T1: Herbert Schildt and Dale Skrien, "Java Fundamentals – A comprehensive Introduction", McGraw Hill.

T2: Herbert Schildt, "Java the complete reference", McGraw Hill, Osborne.

T3: T. Budd, "Understanding Object- Oriented Programming with Java", Pearson Education.

REFERENCE BOOKS:

R1: P.J.Dietel and H.M.Dietel , "Java How to program", Prentice Hall.

R2: P.Radha Krishna , "Object Oriented programming through Java", CRC Press.

R3: S.Malhotra and S. Choudhary, "Programming in Java", Oxford University Press.

OTHER LEARNING RESOURCES:

Web References:

1. http://java.sun.com
2. http://www.oracle.com/technetwork/java/index.html)
3. http://java.sun.com/javase
4. http://www.oracle.com/technetwork/java/javase/overview/index.html
5. http://download.oracle.com/javase/7/docs/api/index.html

E-Text Books:

1. http://docs.oracle.com/javase/tutorial/
2. iiti.ac.in/people/~tanimad/JavaTheCompleteReference.pdf
3. https://www.codejava.net/books/4-best-free-java-e-books-for-beginners

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the object-oriented programming concepts and implement in java.	1, 2, 5, 8
2	Demonstrate the building blocks of OOPs language, inheritance, package and interfaces, and analyse real-world problems in terms of these.	2, 3, 4, 5, 7, 8
3	Apply the exception handling methods on programming	7
4	Develop interactive as well as GUI-based java applications in project-based learning.	2, 3, 4, 5, 7, 8
5	Outline the concept of package, interface, multi-threading and File handling in java.	1, 2, 3

SEMESTER – III										
Course Title	**WEB AND MOBILE PROGRAMMING TECHNOLOGIES**									
Course code	**24BCAO2104R**	**Total credits: 4** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
			3	**0**	**2**	**0**	**0**	**0**	**4**	
Pre-requisite	**Nil**	**Co-requisite**	**Nil**							
Programme	**Bachelor of Computer Applications**									
Semester	**Winter/ I semester of the second year of the program**									
Course Objectives	1. To create responsive web pages using HTML and CSS. 2. To understand and apply JavaScript in web development. 3. To equip with the knowledge of frontend development with React.									
CO1	Understand the concept of web development and building website with HTML, CSS.									
CO2	Understand JavaScript language and its role in website development, and implementing it.									
CO3	Equip the knowledge of frontend development with React.js.									
CO4	Understanding the concept of backend development and understanding PHP and its role in backend development.									
CO5	Understanding MySQL and mobile app development, implementing PHP with MySQL with different operations.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Overview of HTML, CSS:** Overview HTML, elements and attributes; CSS styling, selectors, box model; Responsive Design Principles.	**6**	Gain foundational knowledge of web and mobile development, build responsive websites using HTML, CSS	2,3,5
II	**Introduction to Java Script:** Basic syntax, data types, variables, operators, control flow, functions, array, object, DOM manipulation; ES6 features, DOM manipulation with JavaScript; Event handling, asynchronous JavaScript.	**10**	Understanding JavaScript, applying JavaScript knowledge to solve different problems.	2,3
III	**Introduction to Frontend Development:** Introduction to react, react components and JSX, state and lifecycle methods, handling events in React. React Hooks, React Router for single-page applications, State management with Context API, Integrating APIs with React.	**10**	Understand the concept of frontend development and JSX. Understanding React.js and applying its knowledge to create webapp with react.	2,3,5
IV	**Introduction to Backend Development:**Introduction to backend development;Introduction to PHP, syntax, data types, variables, operators, control flow, functions, array, object; Form handling and user input, database connectivity.	**10**	Understand the knowledge in server-side programming with PHP, enabling efficient data handling, user interaction, and database integration.	2,3,5
V	**Backend Development II and Mobile App:** MySQL, connecting PHP to MySQL database, CRUD operations; Mobile app development platforms (iOS, Android); Mobile UI/UX design principles; Mobile app architecture.	**12**	Develop robust full-stack web applications with PHP and MySQL, while also gaining insights into mobile app development principles and best practices.	3,5
Practical Component				
Practical 1	Create a basic HTML page with a title, heading, and paragraph. Explain the	**2**	Demonstrate the ability to create a basic HTML	5

	significance of each HTML tag used.		document structure with a title, heading, and paragraph using appropriate HTML tags.	
Practical 2	Create a responsive web page using HTML5 and CSS3. Include a header, main content area, and footer. Design a web page layout using Flexbox. Create a navigation bar with horizontally aligned items that adjust on smaller screens. Implement media queries to adjust the layout for different screen sizes. Test it on different devices.	**2**	Develop and implement a responsive web page layout using HTML5, CSS3 (including Flexbox and media queries), to ensure optimal viewing experience across various screen sizes.	3,5
Practical 3	Create a product card layout using the CSS box model. Include an image, product name, description, and price.Make the product card layout responsive, ensuring it looks good on both desktop and mobile devices.	**2**	Understand and apply the CSS box model to design web page layouts. They will gain experience in creating responsive designs that adapt to different screen sizes, enhancing their web development skills.	2,3
Practical 4	Write a JavaScript function to calculate the factorial of a number entered by the user.	**2**	Demonstrate proficiency in basic JavaScript syntax and functions. They will be able to solve mathematical problems programmatically and improve their problem-solving skills.	3
Practical 5	Create a JavaScript program that takes user input for name and age, then outputs a message greeting the user and mentioning their age.	**2**	Handle user inputs and outputs using JavaScript, which is essential for creating interactive web applications. They will also improve their understanding of JavaScript data types and variables.	3
Practical 6	Write a JavaScript script to store a list of students and their scores in an array of objects. Calculate and display the average score.	**2**	Experience working with arrays and objects in JavaScript. They will learn how to organize data effectively and perform calculations based on that data, which is crucial for developing data-driven applications.	4
Practical 7	Create a to-do list application where users can add, remove, and mark tasks as completed. Use JavaScript to manipulate the DOM.	**2**	Enhance their skills in DOM manipulation using JavaScript. They will learn how to create dynamic, interactive web applications that respond to user actions in real time.	5
Practical 8	Implement a simple counter that increases or decreases a value when buttons are clicked.	**2**	Understand the basics of event handling in JavaScript. They will learn to create simple yet	3

			interactive features in web applications, which is foundational for more complex interactivity.	
Practical 9	Create a simple web app that fetches and displays a list of posts from a public API. Implement error handling for failed API requests.	**2**	Experience with asynchronous JavaScript and API integration. They will learn how to fetch data from external sources and handle potential errors, preparing them for real-world web development scenarios.	5
Practical 10	Build a simple React application that displays a list of products using components. Each product should have a name, image, and price.	**2**	Learn the basics of React, including the creation and use of components. They will develop an understanding of how to structure a React application and manage state, laying the foundation for building complex user interfaces.	5
Practical 11	Create a React component that fetches and displays current weather information based on user input for a city.	**2**	Apply their knowledge of React to build components that interact with APIs. They will gain experience in handling asynchronous operations within React components, which is essential for creating dynamic web applications.	5
Practical 12	Develop a React form component for user login with email and password. Implement form validation and submit handling.	**2**	Learn how to create and manage forms in React, including validating user input and handling form submissions. This is crucial for developing user authentication systems and other interactive features.	5
Practical 13	Convert a class-based component for a counter into a functional component using React Hooks.	**2**	Understand the difference between class-based and functional components in React. They will gain experience using React Hooks to manage state and lifecycle methods, which is vital for modern React development.	4
Practical 14	Set up a React application with multiple pages (e.g., Home, About, Contact) using React Router. Include navigation links.	**2**	Learn how to create single-page applications with multiple views using React Router. They will understand the basics of client-side routing and navigation in React applications.	5

Practical 15	Write a PHP script to calculate the sum, difference, product, and quotient of two numbers provided by the user via a form.	**2**	Demonstrate basic PHP programming skills, including handling user inputs and performing arithmetic operations. This exercise will help them understand the fundamentals of server-side scripting.	3
Practical 16	Create a PHP script that stores a list of books with title, author, and price. Display the list as an HTML table.	**2**	Learn how to work with arrays and data storage in PHP. They will also gain experience in generating dynamic HTML content from server-side scripts, which is a core skill in web development.	5
Practical 17	Develop a PHP form that collects user feedback (name, email, message) and displays the submitted data on a new page.	**2**	Understand how to handle and process form data using PHP. They will learn the basics of form submission, data validation, and user interaction on the server side.	3
Practical 18	Create a PHP script that connects to a MySQL database and displays a list of users from a user's table.	**2**	Learn to implement full CRUD (Create, Read, Update, Delete) operations in a web application using PHP and MySQL. This is a fundamental skill for building and managing web-based databases.	3
Practical 19	Develop a PHP application that allows users to create, read, update, and delete records in a products table.	**2**	Learn to implement full CRUD (Create, Read, Update, Delete) operations in a web application using PHP and MySQL. This is a fundamental skill for building and managing web-based databases.	5
Practical 20	Create a PHP script for user registration and login, including password hashing and session management.	**2**	Understand the basics of user authentication and session management in PHP. They will learn how to securely handle user credentials and manage user sessions, which is crucial for developing secure web applications.	5

Textbooks:

T1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins.

T2. "Eloquent JavaScript: A Modern Introduction to Programming" by MarijnHaverbeke.

T3. "React Up & Running: Building Web Applications" by StoyanStefanov.

Reference:

R1. "Learning Node: Moving to the Server-Side" by Shelley Powers.

R2. "Express in Action: Writing, building, and testing Node.js applications" by Evan Hahn.
R3. "Beginning Flutter: A Hands-on Guide to App Development" by Marco L. Napoli.
R4. "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws" by DafyddStuttard and Marcus Pinto.
R5. "The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations" by Gene Kim, Jez Humble, Patrick Debois, and John Willis.

Other Materials:

- "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley.
- FreeCodeCamp - Creating a RESTful API with Node.js
- https://www.reactnative.express/
- https://flutterbyexample.com/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the concept of web development and building website with HTML, CSS.	2, 3, 5, 6, 7, and 8
2	Understand JavaScript language and its role in website development, and implementing it.	2, 3, 5, 6, 7, and 8
3	Equip the knowledge of frontend development with React.js.	2, 3, 5, 6, 7, and 8
4	Understanding the concept of backend development and understanding PHP and its role in backend development.	2, 3, 5, 6, 7, and 8
5	Understanding MySQL and mobile app development, implementing PHP with MySQL with different operations.	2, 3, 5, 6, 7, and 8

SEMESTER – III									
Course Title	**PE I: FUNDAMENTALS OF AI /ML**								
Course code	**24BCAO2105R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter/ I semester of the second year of the program**								
Course Objectives	1. Learn the basic principles of machine learning and how they can be applied to different scenarios. 2. Understand the differences between supervised and unsupervised learning algorithms. 3. Understand the fundamental concepts and terminology of Artificial Intelligence (AI)								
CO1	Explain machine learning concepts, applications, challenges, and basic data descriptions								
CO2	Apply and evaluate clustering techniques using various methods.								
CO3	Differentiate prediction and classification, apply algorithms, and evaluate performance.								
CO4	Implement and compare various search strategies and algorithms.								
CO5	Use logic and probabilistic models for knowledge representation and reasoning.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Machine Learning:** What is machine learning, applications of machine learning, issues and challenges faced in machine learning, types of data attributes – nominal, binary, ordinal, numeric, discrete and continuous, basic statistical descriptions of data – measures of central tendency and dispersion, major tasks in data pre-processing	**8**	Define machine learning, list its applications, identify common issues, distinguish between types of data attributes, describe basic statistical data measures, and outline key data pre-processing tasks.	1, 2, 4
II	**Unsupervised Learning - Data Clustering:** what is cluster analysis, need for cluster analysis, data matrix, dissimilarity matrix, proximity and dissimilarity measures for different data attribute types, partitioning based clustering methods, density-based clustering methods and hierarchical clustering methods, measuring cluster quality	**8**	Define cluster analysis, explain its necessity, describe data and dissimilarity matrices, discuss proximity and dissimilarity measures, compare partitioning, density-based, and hierarchical clustering methods, and evaluate cluster quality.	1, 2, 3, 4
III	**Supervised Learning – Prediction:** What is prediction, application areas of prediction, Simple linear regression, Multiple linear regression, Predictor Error measures **Classification**: What is classification, differences between classification and prediction, applications of classification, Some classification algorithms, Confusion matrix and metrics for evaluating classifier performance	**8**	Define prediction and classification, identify their applications, explain the differences between them, describe linear regression models and prediction error measures, list classification algorithms, and evaluate classifier performance using confusion matrices and metrics.	1, 2, 3, 4
IV	**Introduction to Search:** Searching for solutions, Uniformed search strategies, Informed search strategies, Local search	**8**	Understand various search strategies, including uninformed and informed	2, 3, 4

	algorithms and optimistic problems, Adversarial Search, Search for games, Alpha – Beta pruning.		methods, explore local search algorithms and adversarial search techniques, and apply alpha-beta pruning in game search scenarios.	
V	**Knowledge Representation & Reasoning:** Propositional logic, Theory of first order logic, Inference in First order logic, Forward & Backward chaining, Resolution, Probabilistic reasoning, Utility theory, Hidden Markov Models (HMM), Bayesian Networks.	**8**	Master propositional and first-order logic, implement inference mechanisms like forward and backward chaining and resolution, understand probabilistic reasoning including Hidden Markov Models and Bayesian Networks, and apply concepts of utility theory.	3, 4, 5

TEXT BOOKS:

T1: Jiawei Han and MichelineKamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, India

T2: Russell, S., &Norvig, P. Artificial intelligence: a modern approach. Third Edition. Pearson new international edition. 2014

REFERENCE BOOKS:

R1: Han, Manilla and Smyth, Principles of Data Mining, PHI, India

R2: Tan, Steinbach, Vipin Kumar, Introduction to Data Mining, Pearson Education Inc.

R3: E. Rich and K. Knight, Artificial Intelligence, Tata McGrawHill.

R4: N. J. Nilsson, Principles of Artificial Intelligence, Narosa

R5: D. W. Patterson, Introduction to Artificial Intelligence and Expert Systems, Prentice Hall of India.

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain machine learning concepts, applications, challenges, and basic data descriptions	1, 2, 3
2	Apply and evaluate clustering techniques using various methods.	2, 3, 4
3	Differentiate prediction and classification, apply algorithms, and evaluate performance.	2, 3, 6
4	Implement and compare various search strategies and algorithms.	2, 3, 7
5	Use logic and probabilistic models for knowledge representation and reasoning.	3, 4, 5

SEMESTER – III									
Course Title	**PE I: BUSINESS INTELLIGENCE**								
Course code	**24BCAO2108R**	**TOTAL CREDITS: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		TOTAL HOURS: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter/ I semester of the third year of the program**								
Course Objectives	1. To learn how managers utilise business analytics to create, address, and assist management decision-making for company challenges. 2. To become accustomed to the procedures required to create, present, and analyse business data. 3. Get knowledge on how to use Excel and its add-ons to solve business challenges								
CO1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.								
CO2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.								
CO3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.								
CO4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods								
CO5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Overview of Business Analytics**: Introduction to Analytics, The Challenge of Decision Making, What Is Business Intelligence, The Business Intelligence Value Proposition, The Combination of Business and Technology	**8**	Define and understand the fundamentals of business analytics, the decision-making challenges, the concept and value of business intelligence, and the synergistic relationship between business and technology.	1,2
II	**Introduction to Data Visualization:** Visualization/ Data Issues, Organization/sources of data, Importance of data quality, Dealing with missing or incomplete data, Data Classification.	**8**	Master the principles of data visualization, including data organization, the significance of data quality, strategies for handling missing data, and data classification techniques.	2, 3
III	**Descriptive Analytics**: Data Visualization and Analytics- Charts (Bars-Pie-Line-Scatter-Map-Bubble-Box & Whisker-Tree map - Heat map-Circle and Area) -Worksheet, Dashboard and Story Board creation	**8**	Create and interpret diverse charts and visualizations such as bar, pie, line, scatter, map, bubble, box & whisker, tree map, heat map, and area charts, and develop comprehensive worksheets, dashboards, and storyboards.	3, 6
IV	**Predictive Analytics**: Linear Regression, Multi-linear Regression and Time Series Forecasting. **Prescriptive Analytics**: Linear optimization, Integer optimization, Non-linear programming, Optimization of Network models and Monte Carlo	**8**	Apply linear and multi-linear regression models and time series forecasting for predictive analytics, and utilize optimization techniques	3, 4

	Simulation			
V	**Applications:** Practical demonstration using R or Python or SPSS by industry experts on Emerging trends in business analytics and intelligence	**8**	Demonstrate proficiency in using software tools like R, Python, or SPSS to implement advanced analytics techniques	3, 5, 6

TEXT BOOKS:

T1: Sharda R, Delen D, Turban E, Aronson J, Liang T. P, (2014), Business Intelligence and Analytics: Systems for Decision Support, 10th edition, Pearson Education.

T2: Powell S. G, Barker K. R, (2014), Management Science: The Art of ModelingWith Spreadsheets, (W/Cd), 4th edition, John Wiley & Sons.

REFERENCE BOOKS:

R1: Linoff G. S, Berry M. J, (2011), Data mining techniques: for marketing, sales, and customer relationship management, 3rd edition, John Wiley & Sons.

R2. Frank B, Green B, Harris T, Van De Vanter K, (2010), Business Intelligence Strategy: A Practical Guide for Achieving BI Excellence, MC Press.

R3. Hair, J. F, Black W. C, Babin B. J, Anderson R. E, Tatham R. L, (2009), Multivariate data analysis, 7th edition, Pearson education.

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.	2, 3, 4
2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.	2, 3
3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.	2, 3, 6
4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods	2, 3, 7
5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.	2, 3, 7, 8

SEMESTER – III									
Course Title	**PE I: Information Security and Cryptosystems**								
Course code	**24BCAO2106R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45 T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter/ I semester of the second year of the program**								
Course Objectives	1. To learn about security policies and their impacts. 2. To assess the framework, lifecycle and controls of security under a variety of scenarios. 3. To analyze the security risk calculations and mitigating them by using various policies.								
CO1	Understand the principles and policies of information security. Analyze and explore the information security controls								
CO2	Assess and evaluate the risk management practices of information security.Identify the disasters and recovering from them with appropriate decisions.								
CO3	Understand the fundamental of Cryptosystems requirements. Identify and apply the concept of Cryptographic algorithms.								
CO4	Analyze and explore the use of authentication and hashing. Gain a deep insight into attacks and emerging security algorithms.								
CO5	Explore and analyze of signature and key exchange algorithms.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Information security principles and framework:** Information Security- Need for Information Security, Assets and Types; CIA - Confidentiality, Integrity, and Availability; Threats -Phishing, DDS, etc; Risk and Impact - Policy & Procedures- Organizational Policy, Standards and Procedures - Information Security Governance - Information Assurance Programme Implementation - Security Incident Management - India Legal Framework: Security Standards and Procedures.	**10**	Understand and implement fundamental information security principles, policies, and governance frameworks to protect organizational assets against diverse threats.	
II	**Security life cycle and controls, Risk Management:** Information Security Life Cycle - Testing, Audit, Review and Controls - Systems Development and Support - General Controls - People Security - User Access Controls - Technical Security - Protection from Malicious Software - Physical Security - Different Uses of Controls; Risk Management Framework and Process - Managing Risk - Risk Treatment- Alternative Risk Management Methodologies.	**8**	Master the security life cycle and develop robust control measures, ensuring comprehensive protection across all layers of an organization's infrastructure.	
III	**Cryptography and Classical Cryptography:** Cryptosystems and basic cryptographic tools, Message integrity- Message authentication codes, Signature Schemes, Nonrepudiation, Certificates, Hash function; Cryptographic protocols, Security; Simple cryptosystems – Shift, Substitution, Affine, Vigenere, Hill, Permutation, and Stream Cipher;	**8**	Understand and implement basic cryptosystems, ensuring message integrity and secure communication using cryptographic tools and protocols.	
IV	**Cryptanalysis, Block Ciphers and Stream Ciphers:**Cryptanalysis – Affine, Substitution, Vigenere, Hill, and LFSR Stream Cipher.	**9**	Analyze and apply cryptographic theories to evaluate and secure	

	Shannon's Theory – Linear Cryptanalysis – Differential Cryptanalysis – Description and Analysis of DES – Description and Analysis of AES – Modes of Operation.		cryptographic systems, with a focus on block ciphers and stream ciphers.	
V	**Hash Function and message authentication, public key cryptography and Discrete Logarithms:** Diffie Hellman, RSA, Hash Functions and Data Integrity – Security of Hash Functions – MD5 – SHA512 – Nested MAC and HMAC – CBC MAC.RSA Cryptosystem – Shanks' Algorithm – Elliptic Curves Over the Reals – Elliptic Curves Modulo a Prime – Elliptic Curves Over Finite Fields – ElGamal Cryptosystems on Elliptic Curves - Elliptic Curve Diffie – Hellman.	**11**	Acquire proficiency in modern cryptographic techniques, including block and stream ciphers, hash functions, and public-key cryptography, to ensure secure communication and data integrity.	

TEXT BOOKS:

T1: Andy Taylor, David Alexander, Amanda Finch and David Sutton, "Information Security Principles",2020, Third Edition, BCS, United Kingdom.

T2: Michael E. Whitman and Herbert J. Mattord, "Management of Information Security", 2018, Sixth Edition, Cengage Learning, United States of America.

T3: Douglas R. Stinson, "Cryptography: Theory and Practice", 2018, 4th Edition, CRC Press, United states.

REFERENCE BOOKS:

R1: Calder, A., and Watkins, S. G., "Information security risk management for ISO27001/ISO27002", 2018, Third Edition, IT Governance Ltd, United States of America.

R2: Susanto, H., and Almunawar, M. N, "Information security management systems: A novel framework and software as a tool for compliance with information security standards", 2018, First Edition, Apple Academic Press, New York.

R3: Bruce Schneier, "Applied Cryptography: Protocols, Algorithms and Source code in C", 2017, 20th edition, John Wiley & Sons, New York

R4: Behrouz A Forouzan, DebdeepMukhopadhyay, "Cryptography and Network Security", 2011, Tata Mcgraw Hill education private limited, India.

R5: Phishing and Communication Channels - A Guide to Identifying and Mitigating Phishing Attacks, Gunikhan Sonowal

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the principles and policies of information security. Analyze and explore the information security controls	1, 2, 3, 4, 5, 6, 7, and 8
2	Assess and evaluate the risk management practices of information security.Identify the disasters and recovering from them with appropriate decisions.	1, 2, 3, 4, 5, 6, 7, and 8
3	Understand the fundamental of Cryptosystems requirements. Identify and apply the concept of Cryptographic algorithms.	1, 2, 3, 4, 5, 6, 7, and 8
4	Analyze and explore the use of authentication and hashing. Gain a deep insight into attacks and emerging security algorithms.	1, 2, 3, 4, 5, 6, 7, and 8
5	Explore and analyze of signature and key exchange algorithms.	1, 2, 3, 4, 5, 6, 7, and 8

SEMESTER – III									
Course Title	**Field Base Learning**								
Course code	**24BCAO2107R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**8**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the first year of the program**								
Course Objectives	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyze and interpret field data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								
Course Contents	The organization will provide the students two field visit for this program. The students will be taken to a specific site for learning purposes. The students will visit a relevant site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners regarding the subject or purpose of the visit. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply theoretical concepts to real-world situations.	2, 4, and 8
2	Conduct field-based research and gather data effectively.	2, 3, and 8
3	Analyze and interpret field data to draw meaningful conclusions.	2, 3, and 7
4	Communicate findings clearly and effectively, both orally and in writing.	2, 5, and 6
5	Demonstrate professional behaviour and teamwork skills in field settings.	5, 6, and 7

SEMESTER – III										
Course Title	**BASIC LIFE SAVING SKILLS (BLSS)**									
Course code	**24UULS2002R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**	
Pre-requisite	**Nil**	**Co-requisite**	**Nil**							
Programme	**Bachelor of Computer Applications**									
Semester	**Winter / I semester of the second year of the program**									
Course Objectives	The aim of the course is to provide the learners with basic knowledge and practical skills needed in an emergency fire situation, and to provide appropriate basic management and treatment for injuries									
CO1	Understand the specific skills of Basic life-saving skill and knowledge that individuals should acquire and demonstrate after completing a BLS training course.									
CO2	Explain the soft skill attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally.									
CO3	Implement the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies.									
CO4	Understand the principles and purpose of the Triage system in healthcare settings.Describe thedifferent levels of triage and their significance in prioritizing patient care.									
CO5	Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage.Recognize the importance of accurate and timely patient assessment during triage.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Basic Life Support (BLS)** Introduction of BLS Chain of survival ABCs Assessment CPR and Ventilation Technique AED Choking for adult and children	**5**	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives	1,2
II	**Soft skills** Introduction Communications Skills Situational Skills Team Work Other Soft Skills	**4**	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life	1,2
III	**Trauma emergencies** Introduction Priorities of Initial approach in pre-hospital care Scene safety Primary assessment Bleeding control Helmet removal Care of amputated body part Extrication of victims and safe transfer	**10**	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders, and anyone involved in emergency care.	1,2,3

	Cervical spine stabilization Cervical collar application Splinting of broken Limbs			
IV	**Triage system** Introduction Flow chart approach of Triage Triage of Multiple Casualties in Pre-Hospital setting Triage of Single casualt	**5**	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate, delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage	1,2,3,4
V	**Medical emergencies:** Introduction Victim centred approach in medical emergency Management of :- a)seizures b)heart attack c)asthma diabetic emergencies emergency childbirth stroke recovery position	**6**	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction (heart attack), stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.	1,2,3,4

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the specific skills of Basic life-saving skill and knowledge that individuals should acquire and demonstrate after completing a BLS training course.	5,6
2	Explain the soft skill attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally.	5,6
3	Implement the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies.	5,6
4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage and their significance in prioritizing patient care.	5,6
5	Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.	5,6

SEMESTER – III									
Course Title	**ENGLISH LANGUAGE FOR EXCELLENCE (Communicative English & Soft Skills**								
Course code	**24UBPD2103 R**	**Total credits: 2** **Total hours: 60P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ I semester of the second year of the program**								
Course Objectives	1. To enable students to learn, understand and practice different ways of presentation and use of correct grammar. 2. To augment the writing skills in different areas including CV and cover letter writing. 3. To boost productivity and performance at work, which assists in the achievement of professional goals.								
CO1	Develop better listening skills and enhance writing abilities.								
CO2	Identify both positive and negative nonverbal gestures and signs.								
CO3	Improve adept presentation skills by practicing delivery, refining communication techniques, and ensuring clarity of expression.								
CO4	Enhance writing abilities and develop the ability to recognize awkward expressions, correct grammatical errors, and enhance the overall coherence of your writing								
CO5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Grammar :** I. Use of Prepositions II. Tag questions III. Simple, complex, compound sentences	**10**	Describe, illustrate, and explain Prepositions, types of sentences	1,2
II	**Grammar:** I. Active and Passive Voice II. Direct and Indirect Speech	**15**	Describe, illustrate, and explain Voice, Types of voice, Speech	1,2
III	**Writing Skills :** The Basics of Writing; avoid ambiguity and vagueness I. Paragraph Writing II. Letter Writing III. Resume, CV and Cover Letter	**15**	Describe, illustrate, and explain about Paragraph writing, letter writing, Creation of resume, cover letter.	1,2,3
IV	**Self- Management Skills** SWOT Analysis Self- Regulation Personal Hygiene	**10**	Describe, illustrate, and explain and apply Self management skills	1,2,3,4
V	**Non- Verbal Communication-Sciences of Body Language** i. What is Non- Verbal Communication & Body Language, ii. Elements of Communication, iii. Types of Body Language, iv. Importance and Impact of Body Language, v. Types of Communication through Body Language, vi. Body	**10**	Describe, illustrate, and explain and apply Non verbal communication.	1,2,3,4

	Language Do's and Don'ts, Doubt Clearing Session.			

TEXT BOOKS:

1. What Employers Want : The Work skills Handbook- Karen Holmes, 2011
2. English Grammar in Use, Raymond Murphy 4th edition, CUP

REFERENCES:

1. Professional Communication, 2015, by Dr.PrachiDr. S. K. Singh

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop better listening skills and enhance writing abilities.	5,6,8
2	Identify both positive and negative nonverbal gestures and signs.	5,6,8
3	Improve adept presentation skills by practicing delivery, refining communication techniques, and ensuring clarity of expression.	3,5,6,7,8
4	Enhance writing abilities and develop the ability to recognize awkward expressions, correct grammatical errors, and enhance the overall coherence of your writing	5,6,8
5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills.	2,3,5,6,7,8

SEMESTER – III										
Course Title	**Personal Financial Planning**									
Course code	**24UUFL2002**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives	1. To create awareness among students about the need for possessing financial literacy education. 2. To Identification of money as a working asset. 3. To impart the ability to make better financial decisions								
CO1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.								
CO2	Design a diversified investment portfolio that addresses several different investment objectives.								
CO3	Differentiate between open- and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.								
CO4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** i. Meaning, need and importance of Financial Literacy; ii. Different components of Financial Literacy; iii. Prerequisites of financial literacy; iv. Savings – Meaning and Difference between savings and investment; v. Types of Financial Institutions and the services provided - Banking and Non-Banking; vi. Different investment avenues.	**7**	Describe, illustrate, and explain concepts of money, Importance of Financial Literacy heir applications to financial planning.	1,2
II	**Financial Planning:** i. Meaning, need and importance for financial planning, ii. Economic needs, balancing between economic need and resources; iii. Three pillars of investments-risk, return, liquidity; iv. Budgeting and its importance in financial planning; v. Steps involved in Financial Planning Process; vi. Preparation of personal budgets, budget surplus and budget deficit, avenues for savings from surplus, sources for meeting deficit. vii. Informal Society funds and crowd funding	**6**	Describe, illustrate, and explain financial planning process, the life cycle of financial plans, and methods of goal achievement.	1,2
III	**Banks & Post Office - As financial service provider:** i. Meaning and evolution of money, ii. Banks – meaning, types &	**6**	Describe, illustrate, and explain Formulation of budget, record keeping system, and tax planning	1,2,3

	functions; types of accounts; Formalities to open various accounts. iii. Different types of Post Office saving schemes: Recurring deposit, savings, term deposit; NSC; KisanVikasPatra; Monthly Income scheme (MIS) Account, iv. ublic Provident Funds (PPF), Senior citizen savings scheme (SCSS), SukanyaSamriddhi Accounts, v. Indian Postal Order; International Money transfer service; Forex Services; vi. Money remittance services; Jansuraksha Scheme.		strategy based on current financial goals.	
IV	**Insurance - As financial service provider:** i. Different types of Risks and their Management, Diversification of risk; ii. Meaning,need and importance of Insurance; Types of Insurance – Life Insurance, Health Insurance, General Insurance, Term Insurance, iii. Pension and retirement policies; iv. Post office life insurance schemes, Postal life insurance and rural postal life insurance.	**6**	Describe, illustrate, and explain and Creation of financial plan that covers Risk management, importance of Insurance.	1,2,3,4
V	**Transformations in Digital Money market:** i. Various functions & innovative services of Banks; Mobile Banking, NEFT, IMPS, RTGS, ii. Money transfer, Different types of cards- Debit & Credit, E-Banking,Unified payment interface(UPI), iii. Credit Scoring - CIBIL, Digital Banking, crypto currency and related transactions, **iv.** Fintech, Block chain; Understanding Digital Payments.		Explain the Digital money market , credit score, CIBIL score.	

Textbooks:

T1. Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems and the Million Dollar Financial Advisor: Powerful Lessons and Proven Strategies from Top Producers by David J. Mullen Jr.
T2. Personal Finance and Planning by Dr.Rajni
T3. Peaceful Personal Finance: A short read on the basics of personal finance and planning Kindle Edition by Hema Singh.
T4. Be your own financial advisor: Financial Planning, Investment Options, Risk Management, Tax Management, Succession Planning Kindle Edition y Sushil Bali.

T5. The dumb things smart people do with their money: Thirteen ways to right your financial wrongs Kindle Edition y Jil Schlesinger.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	1,2,4,5,6,7,8
2	Design a diversified investment portfolio that addresses several different investment objectives.	2,3,4,5,6,7
3	Differentiate between open- and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.	4,5,6,8
4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.	1,2,3,4,5,6,7,8
5	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	1,2,3,5,6,7,8

<table>
<tr><th colspan="9">SEMESTER – III</th></tr>
<tr><td>Course Title</td><td colspan="8">MINI PROJECT-III</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BCAO2103R</td><td rowspan="2">Total credits: 1</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>8</td><td>0</td><td>0</td><td>2</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="6">Nil</td></tr>
<tr><td>Programme</td><td colspan="8">Bachelor of Computer Application</td></tr>
<tr><td>Semester</td><td colspan="8">Fall/ I semester of the second year of the program</td></tr>
<tr><td>Course Objectives</td><td colspan="8">1. To develop the knowledge, skills and attitudes of a professional Computer Application professional.
2. To become confident in designing Software solutions to complex software problems utilising a systems approach.
3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.</td></tr>
<tr><td>CO1</td><td colspan="8">Demonstrate a sound technical knowledge of their selected project topic.</td></tr>
<tr><td>CO2</td><td colspan="8">Undertake problem identification, formulation and solution.</td></tr>
<tr><td>CO3</td><td colspan="8">Design Software solutions to complex problems utilising a systems approach.</td></tr>
<tr><td>CO4</td><td colspan="8">Communicate with the community at large in written an oral-forms.</td></tr>
<tr><td>CO5</td><td colspan="8">Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="3">Learning Outcome</td><td>KL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:
1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
● theoretical studies
● computer simulations
● hardware construction;
Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
Deliver a seminar on the general area of work being undertaken and specific contributions to that field;
Prepare a formal report describing the work undertaken and results obtained so far; and
Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.</td><td>90</td><td colspan="3">Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.</td><td>3, 4, 5</td></tr>
</table>

OTHER LEARNING RESOURCES:

As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

Detailed Syllabus of 4th Semester

SEMESTER – IV									
Course Title	**Design and Analysis of Algorithms**								
Course code	**24BCAO2201R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Data Structures**	**Co-requisite**	**C /C++ Programming**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives	1. Analyze the asymptotic performance of algorithms. 2. Demonstrate a familiarity with major algorithms and data structures. 3. Apply important algorithmic design paradigms and methods of analysis.								
CO1	For a given algorithm, analyze worst-case running time based on asymptotic analysis and justify the correctness of algorithm.								
CO2	Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.								
CO3	Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.								
CO4	Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For given problems of dynamic-programming, develop the dynamic programming algorithms, and analyze it to determine its computational complexity.								
CO5	For a given model engineering problem, model it using graph and write the corresponding algorithm to solve the problems.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.	**8**	Define and Identify Characteristics of Algorithms, Understand and Perform Asymptotic Analysis, Evaluate Algorithm Performance, Solve Recurrence Relations, Use Master's Theorem,	**1,2,3**
II	Fundamental Algorithmic Strategies: Brute-Force, Greedy, Dynamic Programming, Branch and Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics –characteristics and their application domains.	**12**	Understand Brute-Force Techniques, Apply Greedy Algorithms, Implement Dynamic Programming Solutions, Use Branch and Bound Techniques, Solve Problems Using Backtracking, Practical Application of Strategies,	**3,4**
III	Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth 6First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	**8**	Solve Problems Using Backtracking, Understand and Implement DFS and BFS, Apply Shortest Path Algorithms, Calculate Transitive Closure, Find Minimum Spanning Trees, Perform Topological Sorting, Understand and Implement Network Flow Algorithms	**3,4**
IV	Tractable and Intractable Problems: Computability of Algorithms,	**6**	Understand and Explain Cook's Theorem, Identify	**4**

	Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques		Standard NP-complete Problems, Solve and Analyze NP-complete Problems, Apply Reduction Techniques,	
V	Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE	**5**	Understand Approximation Algorithms, Apply Approximation Algorithms, Understand Randomized Algorithms, Understand P SPACE	**4**

TEXT BOOKS:

T1: Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms, 4TH Edition, MIT Press/McGraw-Hill.

T2: Fundamentals of Algorithms – E. Horowitz et al.

REFERENCE BOOKS:

R1: Jon Kleinberg and ÉvaTardos, Algorithm Design, 1ST Edition, , Pearson.

R2: Michael T Goodrich and Roberto Tamassia, Wiley Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition,.

R3: Udi Manber, Algorithms -- A Creative Approach, 3RD Edition, Addison-Wesley, Reading, MA.

OTHER LEARNING RESOURCES:

1. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/
2. https://www.vssut.ac.in/lecture_notes/lecture1428551222.pdf

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	For a given algorithm, analyze worst-case running time based on asymptotic analysis and justify the correctness of algorithm.	1,8
2	Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.	1,8
3	Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.	1, 2, 3, 8
4	Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For given problems of dynamic-programming, develop the dynamic programming algorithms, and analyze it to determine its computational complexity	1, 2, 3
5	For a given model engineering problem, model it using graph and write the corresponding algorithm to solve the problems.	1, 2, 3

SEMESTER – IV									
Course Title	**Basics of Python Programming**								
Course code	**24BCAO2202R**	**Total credits: 5** **Total hours:30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			2	**1**	**4**	**0**	**0**	**0**	**5**
Pre-requisite	**Data Structures**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives	1. Understand the programming basics (operations, control structures, data types, etc.) 2. Understand and begin to implement compound data like python lists, tuples etc 3. Read/write to files and learn to use basic libraries in python for data analysis								
CO1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.								
CO2	Express proficiency in the handling of strings and functions.								
CO3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets								
CO4	Identify the commonly used operations involving file systems and regular expressions.								
CO5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Python:** Unique features of Python, Python-2 and Python-3 differences, Install Python and Environment Setup, First Python Program, Python Identifiers, Keywords and Indentation, inserting comments in python, Command line arguments, Getting User Input, Data types, Operators, Precedence and associativity	**6**	Understand Python's Features, Compare Python Versions, Install Python, Manage Virtual Environments, Write a Basic Python Program, Understand Python Syntax, Use Comments Effectively, Use Command Line Arguments, Get User Input, Understand Python Data Types	**1,2,3**
II	**Conditional execution and Iteration:** Boolean expressions, logical operators, conditional and alternative execution, chained and nested conditionals, the while statement, infinite loops and *break*, finishing iterations with *continue*, *for* loops	**6**	Understand Boolean Expressions, Use Logical Operators, Implement Conditional Execution, Use Alternative Execution, Use Chained Conditionals, Implement Nested Conditionals, Understand and Use While Loops, Understand and Use For Loops	**3,4**
III	**Functions:** python built-in functions, defining and calling a function, parameters and arguments, fruitful functions and void functions, anonymous functions	**8**	Understand Built-in Functions, Define Functions, Use Parameters and Arguments, Use Parameters and Arguments, Understand Fruitful Functions	**3,4**
IV	**Data Structures and strings in Python:** Lists – traversing a list, list operations, list slices, list methods Dictionaries – dictionary as a set of counters, looping and dictionaries, Tuples – tuple assignment, dictionaries	**12**	Understand and Traverse Lists, Perform List Operations, Utilize List Slices, Apply List Methods, Work with Dictionaries, Utilize Dictionaries as	**4**

	and tuples, using tuples as keys in dictionaries Strings – string slices, looping and counting, the *in* operator, string comparison, string methods ,		Counters, Understand Tuple Assignment, Work with Dictionaries and Tuples, Utilize String Slices, Loop through Strings and Count, Use the in Operator, Compare Strings, Apply String Methods	
V	**Basic File and Exception Handling in python:** Reading and Writing to Files, Exception Handling **Introduction to basic libraries for data analysis:** Introduction to Matplotlib library, NumPy, Pandas	**8**	Understand the Importance of Exception Handling, Implement Basic Exception Handling, Understand Basic Concepts of Matplotlib, Understand the Role of Numpy in Data Analysis, Use Numpy for Data Manipulation, Understand the Importance of Pandas in Data Analysis	**4**
Practical				
Practical 1	Program to find area and circumference of a circle.	**1**	Creating a python program to find the area and circumference of a circle.	5
Practical 2	Program to convert temperature from degree centigrade to Fahrenheit	**1**	Creating a python program to convert degree to Fahrenheit.	5
Practical 3	Program to calculate Sum of 5 subject and find Percentage	**1**	Creating a python program to calculate the sum of 5 subjects and find the percentage.	5
Practical 4	Program to show swap of two numbers by using third variable	**1**	Creating a python program to swap two numbers using a third variable.	5
Practical 5	Program to show swap of two numbers by without using third variable	**1**	Creating a python program to swap two numbers without using the third variable.	5
Practical 6	Program to reverse a given number	**1**	Creating a python program to reverse a given number.	5
Practical 7	Program to find greatest among 3 Numbers	**1**	Creating a python program to find the greatest number from three inputs.	5
Practical 8	Program to find whether the given number is even or odd	**1**	Creating a python program to find the given number is odd or not.	5
Practical 9	Program to use switch statement	**1**	Creating a python program to make use of the switch case statement.	3, 5
Practical 10	Program to display first 10 natural number and their sum	**1**	Creating a python program for displaying the natural numbers and its sum.	5
Practical 11	Program to find Fibonacci series up to a range	**1**	Creating a python program to print the Fibonacci series.	5
Practical 12	Program to count Number of digits	**1**	Creating a python program to count the number of digits.	5
Practical 13	Program to find factorial of a number	**1**	Creating a python program to find the factorial of a given number.	5

Practical 14	Program to check whether the given number is palindrome or not	**1**	Creating a python program to check a palindrome number	5
Practical 15	Program to find whether the given number is prime or not	**1**	Creating a python program to check for a prime number	5
Practical 16	Program to show dynamic list implementation	**1**	Creating a python program for dynamic lists.	5
Practical 17	Program to find dynamic array implementation	**1**	Creating a python program for dynamic array	5
Practical 18	Program to display matrix	**1**	Creating a python program to display a matrix	5
Practical 19	Program to show the sum of 10 elements of array and show their average	**1**	Creating a python program to calculate the sum of 10 elements of an array and show their average.	5
Practical 20	Program to show the sum of 10 elements of list and show their average	**1**	Creating a python program to calculate the sum of 10 elements of a list and show their average.	5
Practical 21	Program to find the maximum and minimum number in an array	**1**	Creating a python program to find the maximum and minimum number in an array	5
Practical 22	Program to find the maximum and minimum number in a list	**2**	Creating a python program to find the maximum and minimum number in a list	5
Practical 23	Program to create a list and perform the following operation ● Insert a new value in 4th index ● Delete the value from the 2nd index ● Replace the existing value of 1st position ● Insert the value at the end of the list ● Sort the list in ascending order ● Print the reverse of the list	**2**	Creating a python program insert, delete, and replace the value in the list.	5
Practical 24	Program to implement queue using list	**2**	Creating a python program to implement a queue.	5
Practical 25	Program to implement stack using list	**2**	Creating a python program to implement a stack	5
Practical 26	Program to show the sum of two matrices	**2**	Creating a python program to display the addition of two matrices.	5
Practical 27	Program to calculate the square of the elements of a list by using list comprehension method and print the final list	**2**	Creating a python program to calculate the square of the element of a list.	5
Practical 28	Program to explain tuple conversion function	**2**	Creating a python program understand the tuple conversion function.	5
Practical 29	Program to perform the following operation ● Creation of a tuple ● Updation of a tuple ● Deletion of a tuple	**2**	Creating a python program for a different operation on a tuple.	5

	Slicing of a tuple			

TEXT BOOKS:

T1: Allen B. Downy, O'reilly, Think Python

T2: Charles Severance, Python for Everybody: Exploring Data Using Python 3

REFERENCE BOOKS:

R1: Jakes Vander Plas O' Reilly, Python Data Science Handbook

R2: Eric Matthes, "Python Crash Course, A Hands – on Project Based Introduction to Programming", 2nd Edition, No Starch Press, 2019

OTHER LEARNING RESOURCES:

1. https://www.w3schools.com/python/python_intro.asp
2. https://www.tutorialspoint.com/python/index.htm

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.	1, 2, 3
2	Express proficiency in the handling of strings and functions.	1, 2, and 8
3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets	1, 2, and 8
4	Identify the commonly used operations involving file systems and regular expressions.	1, 2, and 8
5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python	1, 2, and 8

SEMESTER – IV									
Course Title	**Computer Networks**								
Course code	**24BCAO2203R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Programming Skills**	**Co-requisite**	**LINUX**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter/ II semester of the second year of the program**								
Course Objectives	1. Understand the fundamentals of data communications principles of media access, switching, routing and flow control. 2. Understand the basics of network protocol design and analysis. 3. Be familiar with the TCP/IP protocol suite and with application layer protocols.								
CO1	Explain the basics of data communication, networking, internet, physical layer techniques and circuit switching.								
CO2	Explain the different data link layer techniques and protocols including flow and error control.								
CO3	Discuss network layer protocols along with routing issues.								
CO4	Summarize transport and application layer operations and protocols along with QoS services								
CO5	Analyse the different protocols used at application layer i.e. HTTP, SNMP, SMTP, FTP, TELNET and VPN.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to networks, internet, protocols and standards, the OSI model, layers in OSI model, TCP/IP suite, Addressing, Analog and digital signals. Physical Layer: digital transmission, multiplexing, transmission media, circuit switched networks, Datagram networks, virtual circuit networks, switch and Telephone network.	**8**	Understand the Concept of Computer Networks, Explain the Internet and its Components, Describe the OSI Model, Understand the TCP/IP Protocol Suite, Understand Signals in Networking, Define switching in networking	**1,2,3**
II	Data link layer: Introduction, Block coding, cyclic codes, checksum, framing, flow and error control, Noiseless channels, noisy channels, HDLC, point to point protocols	**8**	Introduction to Data Link Layer, describe block coding and its use in error detection and correction, describe block coding, Explain cyclic redundancy check (CRC), Define flow control and its importance	**3,4**
III	Network Layer: Logical addressing, internetworking, tunnelling, address mapping, ICMP, IGMP, forwarding, uni-cast routing protocols, multicast routing protocols.	**8**	Define logical addressing, Describe the concept of internetworking, define tunnelling, Explain address mapping techniques	**3,4**
IV	Transport Layer: Process to process delivery, UDP and TCP protocols, SCTP, data traffic, congestion, congestion control, QoS, integrated services, differentiated services, QoS in switched networks.	**9**	Explain address mapping techniques, Define ICMP and its role in reporting errors, Explain IGMP and its role, Define forwarding, Describe unicast routing protocols	**4**
V	Application Layer – Domain name space, DNS in internet, electronic mail, FTP, WWW, HTTP, SNMP, multi-media, network security	**6**	Define domain name space, Describe the role of DNS, Define electronic mail, Define FTP and its role in transferring files, Describe the World Wide Web, Define HTTP and its role	**4**

TEXT BOOKS:

T1: Behrouz A. Forouzan, Data Communications and Networking, Fourth Edition TMH,2006

T2: Andrew S Tanenbaum, Computer Networks, 4th Edition, Pearson Education.

REFERENCE BOOKS:

R1: S.Keshav, An Engineering Approach to Computer Networks, 2nd Edition, Pearson Education

R2: W.A.Shay, Understanding communications and Networks, Cengage Learning 3rd Edition

R3: Nader F. Mir, Computer and Communication Networks, Pearson Education

R4: James F.Kurose,K.W.Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 3rd Edition, Pearson Education

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the basics of data communication, networking, internet, physical layer techniques and circuit switching.	1, 2, 3, 8
2	Explain the different data link layer techniques and protocols including flow and error control.	1, 2, 3, 8
3	Discuss network layer protocols along with routing issues.	1, 2, 3, 8
4	Summarize transport and application layer operations and protocols along with QoS services.	1, 2, 3, 8

SEMESTER – III									
Course Title	**Operating Systems**								
Course code	**23BCAO212R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Winter / I semester of the second year of the program**								
Course Objectives	1. To learn the mechanisms of OS to handle processes and threads and their communication and the mechanisms involved in memory management in contemporary OS. 2. To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols 3. To know the components and management aspects of concurrency management.								
CO1	Explain the basic concepts of Operating Systems and related concepts.								
CO2	Summarize the concepts of processes and threads, process scheduling including Throughput, Turnaround Time, Waiting Time, Response Time.								
CO3	Identify the concept for optimally allocating memory to processes by increasing memory utilization and improving the access time.								
CO4	Demonstrate and implement the concepts of deadlocks and related concepts								
CO5	Implement various techniques of memory and file management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**7**	Understand the Concept of Operating Systems, Trace the Generations of Operating Systems, Identify Types of Operating Systems, Comprehend OS Services, Utilize System Calls, Understand OS Structures, Conceptualize the Virtual Machine	
II	**Processes:** Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. **Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, **Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.	**8**	Understand the Definition of a Process, Comprehend Process Relationships, Identify Different States of a Process, Understand Process State Transitions, Explore the Process Control Block (PCB), Understand Context Switching, Define Threads, Identify Various States of Threads, Understand the Benefits of Threads, Conceptualize Multithreading, Understand the Foundation and Objectives of Process Scheduling, Identify Types of Schedulers, Understand Scheduling Criteria, Explore Scheduling Algorithms, Comprehend Multiprocessor Scheduling	
III	**Inter-process Communication:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution	**8**	Understand the Concept of Inter-Process Communication, Comprehend the Critical	

	Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery		Section Problem, Identify Race Conditions, Understand Mutual Exclusion, Explore Hardware Solutions for Mutual Exclusion, Understand the Definition of Deadlock, Identify Necessary and Sufficient Conditions for Deadlock, Understand Deadlock Prevention, Explore Deadlock Avoidance, Comprehend Deadlock Detection and Recovery	
IV	**Memory Management:** Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging. **Virtual Memory**: Basics of Virtual Memory – Hardware and control, l structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).	**8**	Understand the Basic Concept of Memory Management, Comprehend Logical and Physical Address Mapping, Explore Memory Allocation Techniques, Identify Internal and External Fragmentation, Understand Compaction, Explore Paging, Understand the Basics of Virtual Memory, Comprehend Hardware and Control Structures for Virtual Memory, Explore the Concept of Locality of Reference, Understand Page Faults, Comprehend the Working Set Model, Understand Dirty Pages and Dirty Bits, Explore Demand Paging, Understand Page Replacement Algorithms,	
V	**I/O Hardware:** I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure.. **File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management. **Disk Management:** Disk structure, Disk scheduling, Disk reliability, Disk formatting, Boot-block, Bad blocks.		Understand the Different Types of I/O Devices, Comprehend Device Controllers, Understand Direct Memory Access (DMA), Comprehend the Goals of Interrupt Handlers, Understand Device Drivers, Explore Device-Independent I/O Software, Understand Secondary-Storage Structure, Understand the Concept of a File, Explore File Access Methods, Comprehend File Operations, Understand Directory Structure, Explore File System Structure, Understand File Allocation Methods, Comprehend Free-Space Management, Understand Disk Structure, Explore Disk Scheduling,	

			Comprehend Disk Reliability, Understand Disk Formatting, Explore the Boot Block and Bad Blocks,	
Practical				
Practical 1	CPU scheduling algorithms first come first serve (FCFS).	2	Implement and understand the working of the First Come First Serve (FCFS) CPU scheduling algorithm.	2,3
Practical 2	CPU scheduling algorithms shortest job first (SJF).	2	Apply and analyze the Shortest Job First (SJF) CPU scheduling algorithm.	3,4
Practical 3	CPU scheduling algorithms round robin.	2	Implement and evaluate the performance of the Round Robin CPU scheduling algorithm.	3,6
Practical 4	CPU scheduling algorithms priority.	2	Design and simulate the Priority CPU scheduling algorithm.	4,5
Practical 5	Producer-consumer problem using semaphores.	2	Solve synchronization problems using semaphores in the Producer-Consumer scenario.	3
Practical 6	Dining-philosophers problem.	2	Implement the Dining Philosophers problem to understand deadlock and concurrency control.	3
Practical 7	Memory management techniques multi programming with fixed number of task (MFT).	2	Analyze and implement memory allocation using Multiprogramming with Fixed Number of Tasks (MFT).	3,4
Practical 8	Memory management techniques multi programming with variable number of task (MVT).	2	Implement memory management using Multiprogramming with Variable Number of Tasks (MVT).	3
Practical 9	Contiguous memory allocation worst fit.	2	Apply and evaluate the Worst Fit algorithm for contiguous memory allocation.	3,6
Practical 10	Contiguous memory first fit.	2	Implement and assess the First Fit algorithm for contiguous memory allocation.	3,6
Practical 11	Contiguous memory best fit.	2	Analyze and implement the Best Fit algorithm for contiguous memory allocation.	3,4
Practical 12	Page replacement algorithms in first in first out (FIFO).	2	Implement the FIFO page replacement algorithm and analyze its performance.	3
Practical 13	Page replacement algorithms least recently used (LRU).	2	Apply the LRU page replacement algorithm and evaluate its effectiveness.	3
Practical 14	Page replacement algorithms optimal.	2	Implement the Optimal page replacement algorithm and compare it with other algorithms.	3,6

Practical 15	File organization techniques single level directory.	2	Design and implement file organization using a Single Level Directory structure.	3,5
Practical 16	File organization techniques two level directory.	2	Understand and implement file organization using a Two Level Directory structure.	2,3
Practical 17	File allocation sequentially, indexed, and linked.	2	Implement and compare different file allocation techniques: Sequential, Indexed, and Linked.	3,6
Practical 18	Dead lock avoidance and prevention.	2	Understand and implement techniques for deadlock avoidance and prevention.	2,3
Practical 19	Disk scheduling algorithms FCFS.	2	Implement and evaluate the First Come First Serve (FCFS) disk scheduling algorithm.	3,6
Practical 20	Disk scheduling algorithms SCAN.	2	Apply and analyze the SCAN disk scheduling algorithm.	3,4

TEXT BOOKS:

T1: Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.

T2: Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

REFERENCE BOOKS:

R1: Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing.

R2: Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley

R3: Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India

R4: Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the basic concepts of Operating Systems and related concepts.	2, 3, 5,6,8
2	Summarize the concepts of processes and threads, process scheduling including Throughput, Turnaround Time, Waiting Time, Response Time.	1, 2, 3, 5, 6, 8
3	Identify the concept for optimally allocating memory to processes by increasing memory utilization and improving the access time.	2, 3, 5, 6, 8
4	Demonstrate and implement theconcepts of deadlocks and related concepts	1,2
5	Implement various techniques of memory and file management.	1,2

SEMESTER – IV									
Course Title	**PE II: DATA ANALYTICS USING PYTHON**								
Course code	**24BCAO2205R**	**Total Credits: 3** **Total Hours: 30T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. Equip students with essential skills in algebra and statistical analysis fundamental to data science. 2. Teach students to effectively manipulate data using Python libraries like NumPy and Pandas and create comprehensive visualizations. 3. Enable students to implement unsupervised learning methods and engage with the latest data science advancements through practical applications and case studies.								
CO1	Understand and apply concepts of functions, exponentials, logarithms, polynomials, alternate coordinate systems, and statistical distributions								
CO2	Demonstrate proficiency in using NumPy for various operations and broadcasting, and manipulate data frames in Pandas, including loading data in different formats								
CO3	Create and interpret various types of data visualizations such as histograms, box plots, bar plots, pie charts, and line charts								
CO4	Implement and evaluate unsupervised learning techniques in Python, including K-Means clustering, hierarchical clustering, and Principal Component Analysis (PCA)								
CO5	Analyze the latest improvements and applications in data science through case studies and exploratory data analysis (EDA) in interdisciplinary research areas.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Intermediate Algebra& Linear Algebra: Functions, Exponentials and Logarithm's, Polynomial's, Alternate Coordinate systems, Binomial Distribution, Poisson distribution and Normal distribution its properties, Assumption of ANOVA, Measures of Central Tendency in Data.	**8**	Demonstrate understanding and application of intermediate algebra and linear algebra concepts	1,2
II	**Introduction about NumPy:** Different NumPy Operations, Broadcasting with NumPy, Introduction about Pandas, Reading or Loading data into Data frame, Pandas Data Frame Manipulations, Data Loading /Reading in different formats (CSV, Excel, Json, HTML)	**8**	Master the use of NumPy for performing array operations, broadcasting techniques, and Pandas for data manipulation, including reading, loading, and transforming data within data frames across various formats	2, 3
III	**Introduction to data Visualizations:** Principles Behind Data Visualizations, Histograms Visualize, Box plots-Visualize, the Distribution of Continuous Numerical Variables (Bar Plots Pie Chart Line Chart). Data Visualization using R- Line Plots and Regression.	**8**	Develop the ability to create and interpret various data visualizations, including histograms, box plots, bar plots, pie charts, line charts, and utilize R for generating line plots and conducting regression analysis to effectively communicate data insights.	3, 4
IV	**Unsupervised Learning in Python:**K-Means Theory/ Implementation,	**8**	Apply unsupervised learning techniques in Python,	3, 4

	Quantifying K-Means Clustering Performance, Hierarchical Clustering Theory, Principal Component Analysis (PCA) theory / Implementation. Selection criteria for number of clusters choosing.		including K-Means clustering and hierarchical clustering, evaluate their performance, implement Principal Component Analysis (PCA), and determine the appropriate number of clusters based on selection criteria.	
V	The improvements and most recent developments in the course's topics as well as their most recent applications in the field Data Science. Case Study and EDA in the interdisciplinary research areas of Data Science.	**8**	Analyze recent advancements and applications in data science, applying knowledge through case studies and exploratory data analysis (EDA) in various interdisciplinary research areas to understand current trends and innovations.	4, 5
Practical				
Practical 1	Write a Python Program to implement basic mathematical functions.	**2**	Visualize and Analyse polynomial behaviour.	**4**
Practical 2	Write a Python Program to Plot normal distribution using Matplotlib.	**2**	Implement Binomial Distribution.	**3**
Practical 3	Write a Python Program to calculate mean, median, and mode	**2**	Visualize Cartesian and polar coordinates conversion.	**4**
Practical 4	Write a Python Program to create and manipulate arrays using NumPy.	**2**	Calculate the dot product of two matrices using NumPy.	**3**
Practical 5	Write a program to add a 1D array to a 2D array using broadcasting.	**2**	Create a DataFrame from a dictionary.	**5**
Practical 6	Write a program to load a JSON file into a DataFrame.	**2**	Read data from an Excel file into a Pandas DataFrame	**3**
Practical 7	Write a Python Program to visualize data distribution with histograms.	**2**	Create a histogram for any dataset.	**5**
Practical 8	Write a Python Program to visualize data spread and outliers with box plots.	**2**	Analyze two datasets using box plots.	**4**
Practical 9	Write a Python Program to visualize proportions using pie charts.	**2**	Plot a line chart for stock prices over time.	**5**
Practical 10	Write a Python Program to implement K-Means.	**2**	Evaluate clustering performance using inertia and silhouette scores.	**6**
Practical 11	Write a Python Program to implement PCA (Principal Component Analysis).	**2**	Apply PCA to reduce data dimensions and visualize results.	**3, 4**
Practical 12	Write a Python Program to visualize the inertia for different numbers of clusters.	**2**	Use the elbow method and silhouette analysis to determine the optimal number of clusters	**3, 4**
Practical 13	Write a Python Program to visualize Data Distributions.	**2**	Use Python libraries like seaborn and matplotlib for visualizations.	**3**
Practical 14	Write a Python Program to implement Predictive Modelling.	**2**	**Dataset**: Sales data with features like product ID, sales amount, date, category, etc	**3**
Practical 15	Predicting Rainfall Dataset: Historical	**2**	Predicting Student	**3, 4**

	weather data. **EDA** (Exploratory Data Analysis) Goal: Identify correlations between humidity, temperature, and rainfall patterns.		Performance **Dataset**: Student grades, attendance, and engagement data. **EDA Goal**: Determine factors influencing academic success.	

TEXT BOOKS:

T1: Think Python, Allen B. Downy, O'reilly

T2: Python for Everybody: Exploring Data Using Python 3 Book by Charles Severance

REFERENCE BOOKS:

R1: Python Data Science Handbook, Jakes Vander Plas O' Reilly

R2: Eric Matthes, Python Crash Course, A Hands – on Project Based Introduction to Programming, 2nd Edition, No Starch Press, 2019

OTHER LEARNING RESOURCES:

https://www.w3schools.com/python/python_intro.asp

https://www.tutorialspoint.com/python/index.htm

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and apply concepts of functions, exponentials, logarithms, polynomials, alternate coordinate systems, and statistical distributions	2, 3, 4, 5, and 8
2	Demonstrate proficiency in using NumPy for various operations and broadcasting, and manipulate data frames in Pandas, including loading data in different formats	1, 2, 3, and 5
3	Create and interpret various types of data visualizations such as histograms, box plots, bar plots, pie charts, and line charts	2, 3, 5, and 6
4	Implement and evaluate unsupervised learning techniques in Python, including K-Means clustering, hierarchical clustering, and Principal Component Analysis (PCA)	2, 3, 5, and 7
5	Analyze the latest improvements and applications in data science through case studies and exploratory data analysis (EDA) in interdisciplinary research areas.	2, 3, 5, 7, and 8

SEMESTER – IV									
Course Title	**PE II –Ethical Hacking Fundamentals**								
Course code	**24BCAO2206R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+15P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. Acquire knowledge and skills needed for performing the hacking process. 2. Understand the various hacking methodologies. 3. Analyze the hacking process on different platforms and the formal reporting writing process.								
CO1	Discuss the basics of ethical hacking.								
CO2	Elaborate on the hacking methodology.								
CO3	Demonstrate the hacking process on different platforms.								
CO4	Illustrate the various application of attacks on different wireless platforms.								
CO5	Create formal reports for the security attacks launched.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Ethical Hacking** Ethical Hacking Concepts, Scope, Types, and Phases, Process of Malicious Hacking, Information Security Controls, Physical Security, Incident Management, Vulnerability Assessment, and Penetration Testing.	**5**	Understand the concepts, scope, and phases of ethical hacking, describe the processes of malicious hacking and information security controls, and apply knowledge of physical security, incident management, vulnerability assessment, and penetration testing.	1,3
II	**Pre Hacking Phases** Footprinting, Reconnaissance, Scanning, and Enumeration; Foot printing concepts, Footprinting methodology, Website Footprinting. Email Footprinting, Information Gathering, WHOIS Footprinting, DNS Footprinting, Footprinting through Social Engineering Footprinting tools. Footprinting Countermeasures. Overview of Scanning, Network, Scanning, methodology. Enumeration Concepts.	**6**	Analyze foot printing, reconnaissance, scanning, and enumeration concepts and methodologies, apply tools for various foot printing techniques such as website, email, and DNS, and develop countermeasures to mitigate risks in pre-hacking phases.	3,4, 5
III	**The system, Web, and Network Hacking** System Hacking, Trojans and Black Box Vs White Box Techniques. Denial of Service, Botnets, Sniffers, and Hacking Web Servers: Session Hijacking, Web Application Vulnerabilities, and Web-Based Password Cracking Techniques, SQL Injection, Viruses, Worms and Physical Security: Viruses and Worms, Physical Security. Linux Hacking: Linux Hacking. Evading IDS and Firewalls: Evading IDS and Firewalls.	**6**	Identify vulnerabilities in system hacking, Trojans, DoS attacks, and botnets, examine web application vulnerabilities and password cracking techniques, and apply methods for Linux hacking and evading IDS and firewalls.	1,3, 4

IV	**Hacking Wireless Networks, Mobile Platforms** Wireless threats: Access control Attacks, Integrity Attacks, Rouge Access Point attacks, Ad Hoc Connection attacks, Jamming Signal attacks. Wireless Hacking Methodology: Foot Printing wireless Network, Wi-Fi discovery tools. Mobile Platform Attack: Vulnerabilities in the Business environment, OWASP mobile top 10 risks, Security Issues – Mobile Apps, Hacking Android OS, Apple iOS, Windows Phone OS, Jailbreaking, Mobile Device Management (MDM), Bring Your Own Device (BOYD), BOYD Risks, Mobile Security guidelines.	**7**	Evaluate wireless threats and attack types, apply wireless hacking methodologies using Wi-Fi discovery tools, and assess mobile platform vulnerabilities while proposing security guidelines for mitigation.	3,5, 6
V	**Report Writing & Mitigation** Introduction to Report Writing & Mitigation, requirements for low-level reporting & high-level reporting of Penetration testing results, Demonstration of vulnerabilities, and Mitigation of issues identified including tracking.	**7**	Develop comprehensive penetration testing reports at different levels, demonstrate vulnerabilities with actionable mitigation strategies, and track the resolution of identified issues.	3,5
Practical				
Practical 1	Performing Active and Passive reconnaissance.	**2**	Demonstrate the ability to gather information about a target system through active and passive reconnaissance techniques, enhancing skills in intelligence gathering and risk identification.	3
Practical 2	Port Scanning information using Nmap.	**2**	Analyze network configurations and security posture by performing port scans with Nmap, identifying open ports, and evaluating potential vulnerabilities.	4
Practical 3	Vulnerability Scanning using Wireshark.	**2**	Perform vulnerability scanning using Wireshark, interpret captured packets, and identify potential network vulnerabilities.	3, 4
Practical 4	Exploiting NetBIOS Vulnerability.	**2**	Demonstrate exploitation of NetBIOS vulnerabilities to understand risks associated with network protocols, and propose mitigation strategies.	3, 5
Practical 5	Creating and analyzing virus.	**2**	Develop a simple virus for educational purposes, analyze its behaviour, and evaluate security measures to mitigate similar threats.	4,5, 6

Practical 6	Performing SQL injection attack.	**2**	Perform an SQL injection attack to understand web application vulnerabilities and evaluate techniques to secure databases from similar exploits.	3,6
Practical 7	Creating and analyzing spoofed emails.	**2**	Create and analyze spoofed emails to understand phishing attacks and recommend countermeasures to enhance email security.	4,5, 6
Practical 8	Wi-Fi password cracking.	**2**	Demonstrate Wi-Fi password cracking techniques to identify security gaps in wireless networks and propose measures to strengthen wireless security.	3,5
Practical 9	Performing MITM attack.	**2**	Conduct a Man-In-The-Middle (MITM) attack to understand the risks of data interception and develop mitigation strategies.	3,5
Practical 10	Launching DDoS attack.	**2**	Simulate a DDoS attack to analyze its impact on network performance and evaluate strategies to defend against such threats.	3,5

Textbooks:

1. CEH v9: Certified Ethical Hacker - Version 9 Study, Sean-Philip Oriyano (Author), Wiley Publications, 2016.

References:

1. *"The Basics of Hacking and Penetration Testing"* by Patrick Engebretson.
2. *"Ethical Hacking and Penetration Testing Guide"* by Rafay Baloch.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Discuss the basics of ethical hacking.	2,3,4,5,6, and 8
2	Elaborate on the hacking methodology.	2,3,5,6, and 8
3	Demonstrate the hacking process on different platforms.	2,3,5,6,7, and 8
4	Illustrate the various application of attacks on different wireless platforms.	2,3,4,5,6,7, and 8
5	Create formal reports for the security attacks launched.	2,3,5,6,7, and 8

SEMESTER – IV										
Course Title	**PE II – Cyber Forensics**									
Course code	**24BCAO2210R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+15P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. Introduce students to the fundamental concepts and procedures in computer forensics and cybercrime investigations. 2. Provide an in-depth understanding of storage devices, data acquisition, and recovery techniques. 3. Familiarize students with forensic techniques for Windows, Linux, mobile devices, networks, and email tracking.									
CO1	Describe the principles of computer forensics and implement cybercrime investigation procedures.									
CO2	Analyze storage devices and apply techniques for data recovery and volatile data analysis.									
CO3	Utilize forensic tools and techniques for investigating various platforms, including networks, mobile devices, and email systems.									
CO4	Apply cyber law principles and evidence-handling procedures to ensure legal compliance during investigations.									
CO5	Conduct forensic analysis of web applications, including server logs and security breaches, to identify intruders and vulnerabilities.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Computer Forensics** Introduction to Computer Forensics, Forms of Cyber Crime, First Responder Procedure- Non-technical staff, Technical Staff, Forensics Expert and Computer Investigation procedure, Case Studies.	**8**		
II	**Storage Devices & Data Recover Methods** Storage Devices- Magnetic Medium, Non-magnetic medium and Optical Medium, Working of Storage devices-Platter, Head assembly, spindle motor, Data Acquisition, Data deletion and data recovery method and techniques, volatile data analysis, Case Studies.	**10**		
III	**Forensics Techniques** Windows forensic, Linux Forensics, Network forensics – sources of network-based evidence, other basic technical fundamentals, Mobile Forensics – data extraction & analysis, Steganography, Password cracking-Brute force, Cross-drive analysis, Live analysis, deleted files, stochastic forensics, Dictionary attack, Rainbow attack, Email Tacking – Header option of SMTP, POP3, IMAP, examining browsers, Case Studies.	**8**		
IV	**Cyber Law** Corporate espionage, digital evidences handling procedure, Chain of custody, Main features of Indian IT Act 2008 (Amendment), Case Studies, Incident	**6**		

	specific procedures – virus and worm incidents, Hacker incidents, Social incidents, physical incident, Guidelines for writing forensic report.			
V	**Forensic Analysis of Web Application** Forensic analysis of web server, network analysis of web server compromise, web server log analysis, web application forensic, forensic analysis of web application security, intruder profiling, forensic for code injection attack, Case Studies	**5**		
Practical Component – NIL				

Text Books:

T1. Computer Forensics: Computer Crime Scene Investigation by John Vacca, Laxmi Publications, 1st ed; 2015

T2. Digital Forensic: The Fascinating World of Digital Evidences by Nilakshi Jain, et.al, Wiley, 1st ed; 2016

T3. The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics by John Sammons, Syngress, 2nd ed; 2014

T4. Cyber Forensics in India: A Legal Perspective by Nishesh Sharma, Universal Law Publishing - an imprint of LexisNexis; First 2017 edition

T5. NetworkForensics:Tracking Hackers Throu by Davidoff, Pearson India, 1st ed; 2013

Reference Books:

R1.Hacking Exposed Computer Forensics by Aaron Philipp, David Cowen, McGraw Hill, 2nd ed; 2009

R2.Mastering Mobile Forensics by SoufianeTahiri, Packt Publishing, 1st ed; 2016

R3.Computer Forensics: A Beginners Guide by David Cowen, McGraw Hill, 1st ed; 2013

R4.Practical Digital Forensics Kindle Edition by Richard Boddington, Packt Publishing, 1st ed; July 2016

R5.Learning Network Forensics by Samir Datt, Packt Publishing, 1st ed; 2016

OTHER LEARNING RESOURCES:

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Describe the principles of computer forensics and implement cybercrime investigation procedures.	2,3,5,6,7, and 8
2	Analyze storage devices and apply techniques for data recovery and volatile data analysis.	1,2,3,5,6,7, and 8
3	Utilize forensic tools and techniques for investigating various platforms, including networks, mobile devices, and email systems.	1,2,3,5,6,7, and 8
4	Apply cyber law principles and evidence-handling procedures to ensure legal compliance during investigations.	2,3,4,5,6,7, and 8
5	Conduct forensic analysis of web applications, including server logs and security breaches, to identify intruders and vulnerabilities.	1,2,3,5,6,7, and 8

SEMESTER – IV									
Course Title	**PE II: KNOWLEDGE ENGINEERING**								
Course code	**24BCAO2209R**	**Total Credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total Hours: 30T+15P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives	1. Understand the basics of Knowledge Engineering. 2. Discuss the knowledge representation and reasoning methods. 3. Apply reasoning and uncertainty for intelligent systems								
CO1	Understand the basics of Knowledge Engineering								
CO2	Interpret the knowledge representation and reasoning methods.								
CO3	Apply reasoning and uncertainty for intelligent systems								
CO4	Design and develop ontologies								
CO5	Understand learning and rule learning								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Knowledge Engineering:** Introduction Data, Information and Knowledge Skills of Knowledge Engineer Knowledge based systems Types of Knowledge based systems Expert Systems Neural Networks Case Based Reasoning Genetic Algorithms Intelligent Systems Data Mining	**8**	Gain a foundational understanding of data, information, and knowledge distinctions; acquire skills essential for a knowledge engineer	1,2
II	**Knowledge Representation and Reasoning:** Knowledge Acquisition Knowledge Representation and Reasoning Using Knowledge Logic, Rules and Representation Developing Rule based Systems Semantic Networks Frames	**8**	Master the processes of knowledge acquisition and representation, develop skills in reasoning with knowledge using logic and rules, and effectively create rule-based systems, while understanding the applications of semantic networks and frames in artificial intelligence.	2, 3
III	**Reasoning under Uncertainty:** Introduction Abductive reasoning Probabilistic reasoning Enumerative Probabilities Subjective Bayesian view Belief Functions Baconian Probability Fuzzy Probability Uncertainty methods Evidence based reasoning Intelligent Agent Mixed Initiative Reasoning Knowledge Engineering.	**8**	Understand and apply various methods of reasoning under uncertainty; develop competence in integrating these approaches within intelligent agents and mixed-initiative reasoning environments for effective knowledge engineering.	3, 4
IV	**Ontologies Design and Development:** Concepts and Instances Generalization Hierarchies Object Features Defining Features Representation Transitivity Inheritance Concepts as Feature Values Ontology Matching Design and Development Methodologies Steps in Ontology Development	**8**	Acquire skills to design and develop ontologies, understanding the structural elements such as concepts, instances, generalization hierarchies, and object features, and apply methodologies for ontology matching and development	3, 4
V	**Learning and Rule Learning:**	**8**	Master the foundational	3, 4, 5

	Machine Learning Concepts Generalization and Specialization Rules Types of Generalization and Specialization Formal definition of Generalization. Modelling, Learning and Problem Solving		concepts of machine learning, focusing on generalization and specialization principles; understand formal definitions and different types of generalization, and apply these concepts in modelling, learning, and problem-solving contexts to develop effective rule-based learning systems	
Practical				
Practical 1	Implement a basic expert system in Python using a set of rules to classify diseases based on symptoms input by the user.	**2**	Apply rule-based reasoning to create an expert system and develop a knowledge-based system that classifies input data.	**3, 5**
Practical 2	Write a Python program that uses a neural network (e.g., Multi-Layer Perceptron) to predict stock market trends based on historical data.	**2**	Implement a neural network to make predictions and analyze historical data for trend prediction.	**3, 4**
Practical 3	Implement a case-based reasoning system in Python that recommends movies based on user preferences and ratings from a dataset.	**2**	Design and implement a case-based reasoning system and apply similarity measures to make recommendations.	**3, 5**
Practical 4	Implement a Python program that represents knowledge using semantic networks. Use this to represent relationships between various entities in a knowledge domain.	**2**	Create semantic network representations of knowledge and apply logical reasoning to entities and their relationships	**5**
Practical 5	Write a Python script that represents knowledge using frames. Develop a system to infer information based on predefined frame structures.	**2**	Implement frame-based knowledge representation and apply reasoning techniques to infer new knowledge.	**3, 4**
Practical 6	Implement a rule-based reasoning system in Python using logical rules to deduce new facts based on known information.	**2**	Apply rule-based reasoning to derive new conclusions from known facts and create systems that work based on logical rules.	**3**
Practical 7	Implement a Python script that demonstrates probabilistic reasoning by calculating the probability of an event given prior probabilities and conditional probabilities.	**2**	Apply probabilistic reasoning techniques and calculate probabilities based on known data and relationships.	**3**
Practical 8	Write a Python program to apply Bayesian reasoning for decision making in uncertain environments, such as predicting the weather based on past observations.	**2**	Apply Bayesian reasoning to solve real-world problems and implement decision-making models using probabilistic techniques.	**3, 5**
Practical 9	Implement a Python function that uses belief functions to model uncertain knowledge and make inferences.	**2**	Apply belief functions for reasoning under uncertainty and demonstrate uncertainty handling in a knowledge-based system.	**3**

Practical 10	Write a Python program to represent an ontology for a given domain (e.g., animals or vehicles) using classes and instances.	**2**	Design ontologies to represent knowledge in specific domains and create classes and instances to represent real-world concepts.	**5**
Practical 11	Implement a Python function to demonstrate ontology matching, where two ontologies are compared to identify similar concepts.	**2**	Apply ontology matching techniques and compare ontologies to find similar or equivalent concepts.	**3, 6**
Practical 12	Write a Python program to define a hierarchy of concepts in an ontology and demonstrate inheritance using object-oriented principles.	**2**	Design ontological hierarchies and define relationships between concepts and apply inheritance principles in an ontology structure.	**1, 5**
Practical 13	Implement a rule learning algorithm in Python to extract rules from a dataset, such as classifying objects based on attributes (e.g., Iris dataset).	**2**	Implement rule learning algorithms to extract useful patterns and apply these rules to classify data effectively.	**3**
Practical 14	Write a Python program to demonstrate generalization and specialization by modifying rules to adapt to new data.	**2**	Apply generalization and specialization techniques to adapt rules to new situations and develop systems that can refine rules based on incoming data.	**3, 5**
Practical 15	Develop a Python script that uses machine learning techniques to generalize from a set of training data and apply it to make predictions on new data.	**2**	Apply machine learning techniques to model data and generalize from training data to make predictions on unseen data.	**3**

TEXT BOOKS:

T1: Ela Kumar, Knowledge Engineering, I K International Publisher House, 2018

T2: Gheorghe Tecuci, DorinMarcu, Mihai Boicu, David A. Schum, Knowledge Engineering Building Cognitive Assistants for Evidence-based Reasoning, Cambridge University Press, First Edition, 2016

T3: Ronald J. Brachman, Hector J. Levesque: Knowledge Representation and Reasoning, Morgan Kaufmann, 2004.

T4: John F. Sowa: Knowledge Representation Logical, Philosophical, and Computational Foundations, Brooks Cole, Thomson Learning, 2000

REFERENCE BOOKS:

R1: King, Knowledge Management and Organizational Learning, Springer, 2009.

R2: Jay Liebowitz, Knowledge Management Learning from Knowledge Engineering, 1st Edition, 2001

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basics of Knowledge Engineering	1, 2, 3, 4, 5, and 8
2	Interpret the knowledge representation and reasoning methods.	2, 3, 5, 7
3	Apply reasoning and uncertainty for intelligent systems	2, 3, 5, 7
4	Design and develop ontologies	2, 3, 5, 7
5	Understand learning and rule learning	1,2,3,5,6,7,8

SEMESTER – IV									
Course Title	**BASIC ACCLIMATIZING SKILLS (BAS)**								
Course code	**23UULS221R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**All the Under Graduate Programmes**								
Semester	**Fall/I or Winter/II Semester of Second Year of the Programme**								
Course Objectives	1. To impart knowledge of the fundamentals of Hospitality industry and its applications. 2. To familiarize with the cooking equipment& Utensils. 3. To handle different modes of reservations.								
CO1	Students will have basic knowledge of cooking methods.								
CO2	Students will gain the knowledge of organizing & Cleaning of Rooms.								
CO3	Students will be able to gain the travel management concept.								
CO4	Students will be able to acquire the knowledge of basic households amenities for day- to-day use.								
CO5	Understand the importance of time management and organization in acclimatizing to new academic demands.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Accommodation Management:** Telephone handling technique, Organizing of Rooms, Cleaning agents, Cleaning equipments and uses, Bed making Process.	**7**	Describe, illustrate, and explain cell organization and functions, microscopy, and structural differences.	1,2
II	**Fundamentals of Cooking**: Definition of cookery –Aim & Objectives of cooking, Use of basic Cooking equipment, Personal Hygiene and Safety, Use of Fire & Fuels	**10**	Describe, illustrate, and explain membrane structure, function; cell organization, and the proteins involved in transportation.	1,2
III	**Methods of Cooking:** Different Cuts. Use of Herbs and Spices. Basic Food and Beverage Preparation. Regional food Habits	**10**	Describe, illustrate, and explain chromosomal structure and types.	1,2
IV	**Forms & Format's**: C –form Reservation form Registration form Passport Application form Legal Rent Agreement	**8**	Describe, illustrate, and explain the mechanism of cell-to-cell communication	1,2
Practical	1. Staining and microscopic observation of various stages of Mitosis of given sample(s). 2. Staining and microscopic observation of various stages in Meiosis of given sample(s).	**30**	Describe, illustrate and explain and apply staining techniques and carry out microscopic examination.	1,2,3,4

TEXT BOOKS:

T1 Arora K (2011). Theory of cookery, Frank brothers & company (pub) pvtltd-New Delhi.

T2 Bruce H. Axler, Carol A. Litrides (2010) Food and Beverage Service Volume 1 ofWiley Professional Restauranteur, Guides.

T3 Mohammed Zulfikar (2010) - Introductions to Tourism and Hotel Industry Introduction to Tourism and Hotel Industry. Vikas Publishing.

T4 Sudhir Andrews (2013) Food and Beverage Service: A Training Manual, Tata McGraw Hill, 2013.

REFERENCE BOOKS:

R1: Cooper GM. The Cell: A Molecular Approach. 2nd edition. Sunderland (MA): Sinauer Associates; 2000.
R2: Ambrose and Dorothy. Cell Biology. 2nd Edition. MEasty, ELBS Publications; 1970.
R3: Sharp, Lester W. Fundamentals of Cytology. 1st edition. Mc Graw Hill Company; 1943.

OTHER LEARNING RESOURCES:

https://www.ncbi.nlm.nih.gov/books/NBK9839/?term=cell%20Biolpgy

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Able to explain the fundamentals and advances of cytology including structure and functions of cell and cell organelles.	1,3 & 4
2	Able to explain the cell cycle and cell division.	1,2
3	Learn and develop skills for operating microscope, preparing slides by various staining techniques	7.9,10
4	Apply knowledge of cellular processes to explain how cells operate and interact within living organisms.	5,7
5	Demonstrate a comprehensive understanding of cell structure and function.	5,8

SEMESTER – IV										
Course Title	**ENGLISH FOR EMPLOYABILITY (Communicative English, Soft Skills & Logical Reasoning)**									
Course code	**24UBPD2202R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**	
Pre-requisite	English Language for Excellence	**Co-requisite**	**Nil**							
Programme	**All the Under Graduate Programmes**									
Semester	**Fall /II Semester of Second Year of the Programme**									
Course Objectives	1. To familiarize students with Public Speaking and helping them overcome stage fear. 2. To enhance the writing skills in different areas including CV and cover letter writing. 3. To enable students, know about the email etiquettes and draft professional emails.									
CO1	Develop skills in script preparation and understand nonverbal communication pivotal to effective public speaking.									
CO2	Improve student's ability in drafting clear, concise, and effective emails.									
CO3	Master the creation, submission, and understanding of screening processes for resumes.									
CO4	Bulid conflict management skills within themselves.									
CO5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Public Speaking:** 1.Preparation of Scripts and understanding Nonverbal cues of Public Speaking 2.Understanding and Overcoming Fear of Public Speaking 3.Practice strategies of Public Speaking Pipe and cistern i. Introduction of pipes and cistern ii. Solving different types of questions	**10**	Understand and overcome the fear of public speaking, and practice strategies to improve their delivery and confidence.	1,2
II	**Practical session on Resume and Cover letter:** i. Preparation, submission & screening of Resume. ii. Practical session on cover letter screening session Mixture allegation and Clock i.Introduction of basics ii. Solving questions on mixture	**10**	Students will master the creation, submission, and understanding of screening processes for resumes.	2,3
III	**Email Etiquettes** I. Different Parts of Email and Usage ii.Drafting emails effectively Statement and Course of action i. Revision of syllogism ii. Statement and conclusion Iii. Course of action based on statement	**15**	Identify, address, and resolve conflicts effectively, enhancing their ability to maintain positive and productive relationships in both personal and professional settings.	1,2,3
IV	**Interview Skills (Mock sessions)** I. Preparing Commonly asked Interview Questions	**15**	Familiarity with Common Interview Questions, Practice in Answering Interview	2,3

	ii.Mock Interview sessions Sitting arrangement (puzzle) i. Linear arrangement puzzle ii. Circular arrangement puzzle iii. Matrix		Questions, Feedback and Improvement, Building Confidence, Understanding Sitting Arrangement Puzzles, Enhancing Logical Reasoning Skills, Teamwork and Collaboration, Preparedness for Assessment Tests	
V	**Conflict Management** i. Definition ii. Type of Conflict Management iii. Effects of Conflict Management Profit loss and discount i. Introduction to basics ii. Introduction to discount iii. Probems related on the topic	**10**	Understand what conflict management entails, including the identification, resolution, and mitigation of conflicts in various contexts, concepts related to profit and loss, including revenue, costs, profit margin, concept of discounting, including types of discounts and their significance in pricing strategies.	1,2,3,4

TEXT BOOKS:

T1 Barrett, Grant. 2016. Perfect English Grammar: The Indispensible Guide to Excellent Writing and Speaking, Zephyros Press.

T2 McDowell, Gayle Laakmann. 2008. Cracking the Coding Interview (Indian Edition)

T3 Professionalism Skills for Workplace Success, Lydia E. Anderson, Sandra B. Bolt, Publisher: Pearson Education

T4 The Art of Public Speaking, Dale Carnegie, Publisher: Diamond Pocket Books Pvt Ltd

T5 English for Academic CVs, Resumes, and Online Profiles, Adrian Wallwork, Publisher: Springer International Publishing

T6 Employment & Volunteering: Job Interview Basics, Lisa Renaud, Publisher: Classroom Complete Press

REFERENCE BOOKS:

R1 Zinsser, William. (2006) On Writing Well: The Classic Guide to Writing Nonfiction, Harper Perennial

R2 Taylor J. and Wright, J., IELTS Advantage Reading Skills: A step-by-step guide to a high IELTS reading score, Delta Publishing by Klett

R3 Murphy, Raymond,.(2012) English Grammar in Use Book with Answers: A Self- Study and Practice Book for Intermediate Learners of English, Cambridge University Press

R4 Real-resumes for Teachers, Anne McKinney, Publisher: Prep Pub.

R5 Public Speaking for Success, Dale Carnegie, Publisher: Penguin Publishing Group

R6 Job Interview Skills , Paige Labert, Publisher: DiDio Calderone Giuseppina

R7 Dressology: The Science of Power Dressing, Nandita Pandey, Publisher: Pan Macmillan

R8 The Anatomy of Peace: Resolving the Heart of Conflict (Paperback) by The Arbinger Institute

OTHER LEARNING RESOURCES:

https://learning.shine.com/talenteconomy/career-help/top-group-discussion-skills/https://www.coursera.org/articles/conflict-management

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop skills in script preparation and understand nonverbal communication pivotal to effective public speaking.	1,3 & 4
2	Improve student's ability in drafting clear, concise, and effective emails.	1,2
3	Master the creation, submission, and understanding of screening processes for resumes.	7.9,10
4	Bulid conflict management skills within themselves.	5,7
5	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills	5,8

SEMESTER – IV									
Course Title	**Computational Systems and Digital World**								
Course code	**24UCDL2201 R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite		**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall / II semester of the second year of the program**								
Course Objectives	1. To understand the fundamentals of computer systems and Internet search along with advanced features of Ms Office. 2. To learn data management, statistical analysis and visualization. 3. To use social media and e-commerce portals, Digital Payment systems, and other utility software.								
CO1	Basic understanding of computer systems and internet search.								
CO2	Solve data analysis, management and visualization issues using Ms Office products.								
CO3	Efficiently and ethically use social media and e-commerce sites.								
CO4	Introduction to various utility software used in research and information management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Fundamentals of Computer Systems, Office Automation and Internet Search:** Components of a computer and their functions; Office automation using Ms Word, Ms Excel, and Ms PowerPoint; Data management, statistical data analysis and data visualization with Ms Excel; Use of functions, graphs, and charts in Ms Excel.	**4**	Understand the components of computers and its functions. Apply the knowledge of different Ms Office for different real-world problem.	**2,3**
II	**Internet and Cyber World:** Introduction to computer networks, internet and world wide web, websites and web portals; Creation and use of email accounts; Web browsing, web searching, different aspects of web searching – search keyword, conditions and combinations; Study different search engines like Google, Microsoft Bing, Yahoo, Yandex, DuckDuckGo, Ask.com, etc; Cyber Crimes, cyber laws and IT Act 2000, India.	**4**	Understanding the basics of Computer network, internet and WWW. Creating email accounts for different purpose. Understanding the search engine and study the different search engines.	**2, 3, 5**
III	**Introduction to Social-Media and E-Commerce:** Relevance of Social-Media in present scenario. Posting different types of contents in Social-Media; Creating accounts and using some popular social media portals and Apps like WhatsApp, Facebook, etc. Social Media etiquettes and crimes; Definition of e-commerce, e-commerce versus traditional commerce; Cast studies of popular e-commerce portals like Amazon; E-commerce etiquettes and crimes.	**4**	Understand what social-media is, and what is the impact of social media on different field. Creating social-media account of different platforms. Understanding e-commerce and case studying on popular e-commerce portals.	**2,3,5**
IV	**Digital payments and Digital Transactions:** Introduction to digital payment systems; Creating accounts and	**4**	Understand the process if digital payment and transactions system. Creating an	**2, 3, 5**

	using Digital Payment Systems like Credit Cards, Debit Cards, Net banking, UPI; Digital payments etiquettes and crimes.		account for digital payment.	
V	**Basic Accounting and Utility Software:** Introduction to Basic accounting concepts; Introduction to an Accounting Software like GnuCash or Tally; Introduction to Technical Document writing using LaTex; Introduction to Data Visualization software – Sigma, Google Charts, Tableau.	**4**	Equips you with basic accounting principles, accounting software usage, technical document writing with LaTex, and data visualization tools for practical application.	**2,3,5**
Practical				
Practical 1	Create a new document with Ms Word, save it with the name "*Student_Name*" in the Documents folder. Formatting text, paragraph alignment.	**1**	Create, save, and format documents in Ms Word, including text formatting and paragraph alignment.	**5**
Practical 2	Creating tables, inserting shapes and images. Creating profile with description and wrapping the images.	**1**	Create tables, insert and format images and shapes in Ms Word documents. Design a profile with a description and incorporate image wrapping for a visually appealing layout.	**5**
Practical 3	Using mail merge, creating documents with existing templates, enable track changes.	**1**	Utilize mail merge in Ms Word to create personalized documents from existing templates. Enable track changes functionality for collaborative document editing in Ms Word.	**5**
Practical 4	Create a simple document with title, author, and body text. Learn basic text formatting options like bold, italic, and underline.	**1**	Create a basic LaTex document with proper structure. Gain control over text appearance and presentation in your document.	**5**
Practical 5	Structure your document with sections and subsections. Learn how to write mathematical equations in LaTex.	**1**	Organize your document for better readability and navigation. Represent mathematical expressions clearly in your document.	**5**
Practical 6	Creating a presentation with Ms PowerPoint, adding and formatting text	**1**	Create presentations in Ms PowerPoint by adding and formatting text content.	**5**
Practical 7	Applying slide layouts, using transitions, inserting and formatting shapes.	**1**	Apply various slide layouts and transitions for visually engaging presentations in MS PowerPoint. Insert and format shapes effectively in MS Power Point presentations.	**5**
Practical 8	Creating and managing animations, using slide master, embedding and linking media.	**1**	Create and manage animations in Ms PowerPoint presentations to enhance visual impact. Utilize the slide master feature in Ms PowerPoint to maintain consistent formatting across slides.	**5**
Practical 9	Creating new excel file, entering data,	**1**	Create and manage	**3, 5**

	formatting cells, using basic formulas.		spreadsheets in Ms Excel by entering and formatting data. Utilize basic formulas in Ms Excel for calculations.	
Practical 10	Creating and formatting table, using functions like VLOOKUP, IF, CONCATENATE, etc.	**1**	Create and format tables in Ms Excel for organized data presentation. Utilize advanced formulas like VLOOKUP, IF, and CONCATENATE in Ms Excel for complex calculations and data manipulation.	**5**
Practical 11	Creating and inserting graphs using different kind of data.	**1**	Create and insert various charts and graphs to visually represent data in Ms Excel.	**5**
Practical 12	Use of advanced formulas, data validation, pivot tables, macros, and automation.	**1**	Utilize advanced formulas in Ms Excel, including data validation for error prevention. Create pivot tables in Ms Excel for data summarization and analysis. Gain an introduction to macros and automation for repetitive tasks in Ms Excel.	**5**
Practical 13	Exploration on search engines, how it works and so on.	**1**	Understand the basic principles of how search engines work and explore techniques for effective online searching.	**2**
Practical 14	Creating a social media account, e-commerce websites, etc. Upload images, videos, exploring the e-commerce websites and learning about etiquette and crimes.	**1**	Create social media accounts and navigate e-commerce websites. Understand online etiquette for responsible interactions on social media platforms. Be aware of potential crimes associated with social media and e-commerce platforms.	**5**
Practical 15	Creating an account in any platform for digital payments and learning about digital payment etiquettes and crimes.	**1**	Create accounts for digital payments. Understand safe practices for online transactions and be aware of potential digital payment crimes.	**5**
Practical 16	Learn how to create basic tables in LaTex.	**1**	Present tabular data in a well-organized manner.	**5**

TEXT BOOKS:

T1: Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems & Applications. 3rd ed. New Delhi: BPB Publications.

T2: Goel, A, 2010. Computer Fundamentals, Pearson India.

REFERENCE BOOKS:

R1: Balaguruswamy, E. 2009 Fundamentals of Computers, Tata McGraw-Hill Education.

R2: Balaguruswamy, 2014. E. Fund of Comp & Programming (Updated Ed Sem. I, Au) Tata McGraw-Hill Education.

R3: Lawson, C. 2022. Introduction to Social Media, Oklahoma State University.

OTHER LEARNING RESOURCES:

- https://www.w3schools.com
- https://edu.gcfglobal.org
- https://www.tutorialspoint.com
- https://www.javatpoint.com
- Latest updates available in WWW.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Basic understanding of computer systems and internet search.	2
2	Solve data analysis, management and visualization issues using Ms Office products.	2,3
3	Efficiently and ethically use social media and e-commerce sites.	2,3
4	Introduction to various utility software used in research and information management.	2

SEMESTER – IV									
Course Title	**MINI PROJECT-IV**								
Course code	**24BCAO2208R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**8**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	4. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 5. To become confident in designing Software solutions to complex software problems utilising a systems approach. 6. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral-forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: Perform a literature search to review current knowledge and developments in the chosen technical area; Undertake detailed technical work in the chosen area using one or more of: ● theoretical studies ● computer simulations ● hardware construction; Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; Deliver a seminar on the general area of work being undertaken and specific contributions to that field; Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	3, 4, 5

OTHER LEARNING RESOURCES:

As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

Detailed Syllabus of 5th Semester

SEMESTER – V									
Course Title	**ADVANCE APPLICATION DEVELOPMENT**								
Course code	**24BCAO3101R**	**Total credits: 4** **Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			2	**0**	**4**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter / V semester of the third year of the program**								
Course Objectives	1. To understand the MERN stack for application development. 2. To develop full-stack proficiency for front-end and back-end development. 3. To create RESTful APIs to handle requests and responses.								
CO1	Understanding the concept of MERN stack.								
CO2	Create RESTful APIs using Express.js.								
CO3	Implementing authorization and authentication using JWT.								
CO4	Creating UI and responsive design using React.js								
CO5	Implementing authorization and authentication in an application.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to MERN Stack:** Introduction to Full-Stack Development - Overview of MERN Stack components (MongoDB, Express.js, React.js, Node.js) - Setting up development environment with Node.js and package managers (npm/yarn)	**8**	This unit lays the groundwork for the MERN stack by establishing a foundational understanding of full-stack development, exploring the core components (MongoDB, Express.js, React.js, Node.js), and setting up the development environment.	**1,2**
II	**Databases and MongoDB and JavaScript Deep Dive:** Introduction to database concepts (relational vs. NoSQL) - MongoDB fundamentals (data models, documents, collections) - Connecting to MongoDB from Node.js applications - Mongoose ODM (Object Data Modeling) for MongoDB; Advanced JavaScript concepts (ES6+, closures, modules), Asynchronous programming (promises, async/await), Introduction to DOM manipulation and event handling	**12**	This unit dives deeper into data management, exploring relational vs. NoSQL concepts, the fundamentals of MongoDB (data models, documents, collections), and connecting to it from Node.js applications using Mongoose ODM. Additionally, students will solidify their grasp of advanced JavaScript features (ES6+, closures, modules) and asynchronous programming (promises, async/await) while gaining an introduction to DOM manipulation.	**2,3**
III	**Building User Interfaces with React.js:** Introduction to React components (functional, class-based) - JSX syntax and component lifecycle - State management with React hooks (useState, useEffect); Introduction to routing and user authentication	**12**	This unit focuses on building interactive user interfaces with React.js. Students will learn to create reusable components (functional and class-based), work with JSX syntax and the component lifecycle, manage application state effectively using hooks	**2,3**

			(useState, useEffect), and gain a basic understanding of routing and user authentication in React applications.	
V	**Backend Development with Node.js & Express.js:** Introduction to Node.js and server-side programming concepts - Building RESTful APIs with Express.js - Connecting to MongoDB databases using Mongoose ODM; Implementing user authentication and authorization.	**12**	This unit delves into server-side development using Node.js and Express.js. Students will learn to build RESTful APIs, connect Node.js applications to MongoDB databases with Mongoose ODM, and implement user authentication and authorization mechanisms to secure their backend applications.	**3,4**
V	**Deployment and Project Development:** Testing strategies for web applications (unit testing, integration testing); Deployment considerations and options (cloud platforms); Introduction to CI/CD pipelines.	**10**	Explore testing strategies (unit testing, integration testing), evaluate deployment considerations (cloud platforms), gain an introduction to CI/CD pipelines, and culminate the course by developing a full-stack web application using the MERN stack, demonstrating their ability to apply the learned concepts to real-world development.	**3,5**
Practical				
Practical	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
Practical 1	Install Node.js and npm/yarn on your machine. Create a simple "Hello World" application using Node.js. Initialize a new Node.js project using npminit or yarn init. Install a few packages (e.g., lodash) and demonstrate their usage in your project.	**2**	Learn to install Node.js and npm/yarn, and initialize a Node.js project.	**3**
Practical 2	Set up a basic folder structure for a MERN stack application. Create a server.js file to serve as the entry point for your Node.js server. Set up a simple Express server with a few routes (/, /about, /contact).	**2**	Understand the basic folder structure for a MERN stack application and set up an Express server.	**3**
Practical 3	Create a RESTful API using Express.js with endpoints for GET, POST, PUT, and DELETE operations. Use middleware to handle errors and log requests.	**2**	Develop a RESTful API with Express.js, handling various HTTP operations and middleware.	**5**
Practical 4	Install MongoDB and set up a local database. Create a collection and insert a few documents using the MongoDB shell.	**2**	Install and set up MongoDB, and perform basic operations using the MongoDB shell.	**3**

Practical 5	Write a few ES6+ features like arrow functions, template literals, and destructuring.	**2**	Apply modern JavaScript features such as arrow functions, template literals, and destructuring.	**3**
Practical 6	Create a Node.js application that connects to a MongoDB database. Perform basic CRUD operations using Mongoose ODM.	**2**	Create a Node.js application that connects to MongoDB and performs CRUD operations using Mongoose.	**3**
Practical 7	Write functions using promises and async/await for asynchronous operations. Manipulate the DOM using vanilla JavaScript to create a simple interactive page.	**2**	Write asynchronous functions with promises and async/await, and manipulate the DOM with JavaScript.	**3**
Practical 8	Define complex data models with relationships (e.g., user and posts). Implement advanced queries and data validation using Mongoose.	**2**	Define complex data models and implement advanced queries with Mongoose.	**5**
Practical 9	Create a small project demonstrating closures, modules, and event handling.	**2**	Demonstrate JavaScript concepts like closures and modules through a small project.	**5**
Practical 10	Set up a new React project using Create React App. Create functional and class-based components. Write components using JSX. Implement component lifecycle methods in class-based components.	**2**	Set up a React project, create components, and manage component lifecycle methods.	**3**
Practical 11	Use React hooks like useState and useEffect to manage state and side effects. Create a simple form with state management.	**2**	Manage state and side effects in React using hooks like useState and useEffect.	**3**
Practical 12	Implement routing in your React application using react-router-dom.	**2**	Implement routing in a React application using react-router-dom.	**3**
Practical 13	Manage complex state using context API or state management libraries like Redux. Implement user authentication and conditional rendering based on user state.	**2**	Manage complex state and implement user authentication in a React application.	**3**
Practical 14	Design and implement reusable components with proper props and state management.	**2**	Design and implement reusable components with proper props and state management.	**5**
Practical 15	Set up a basic Express server with routes for different HTTP methods. Create middleware for request logging.	**2**	Set up an Express server with various HTTP method routes and create middleware for logging.	**3**
Practical 16	Develop a RESTful API with endpoints for creating, reading, updating, and deleting resources. Connect the API to a MongoDB database using Mongoose. Implement user authentication using JWT (JSON Web Tokens).	**2**	Develop a RESTful API connected to MongoDB and implement user authentication using JWT.	**5**
Practical 17	Implement user authorization and role-based access control. Create and handle complex query parameters and request	**2**	Implement user authorization and role-based access control in an API.	**5**

	validation.			
Practical 18	Write unit and integration tests for your API using testing libraries like Mocha or Jest.	**2**	Write unit and integration tests for APIs using testing libraries like Mocha or Jest.	**6**
Practical 19	Develop a full-fledged MERN stack application with user authentication, data validation, and complex state management. Implement advanced features like real-time updates using WebSockets or data visualization.	**4**	Develop a comprehensive MERN stack application with advanced features like real-time updates and data visualization.	**5**

TEXT BOOKS:

T1: "Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node" by Vasan Subramanian

T2: "Learning React: A Hands-On Guide to Building Web Applications Using React and Redux" by KirupaChinnathambi

REFERENCE BOOKS:

R1: "Node.js Design Patterns" by Mario Casciaro

R2: "Express.js Guide: The Comprehensive BookonExpress.js" by AzatMardan

R3: "MongoDB: The Definitive Guide" by Shannon Bradshaw, Kristina Chodorow, and Michael Dirolf.

OTHER LEARNING RESOURCES:

- https://frontendmasters.com/
- https://www.codecademy.com/catalog
- https://v2.scrimba.com/
- https://learn.mongodb.com/
- https://expressjs.com/
- https://nodejs.org/en/download/package-manager
- https://www.freecodecamp.org/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understanding the concept of MERN stack.	1,2,3,5,6,8
2	Create RESTful APIs using Express.js.	1,2,3,5,6,7,8
3	Implementing authorization and authentication using JWT.	1,2,3,5,6,7,8
4	Creating UI and responsive design using React.js	1,2,3,5,6,7,8
5	Implementing authorization and authentication in an application.	1,2,3,5,6,7,8

SEMESTER – V										
Course Title	**CLOUD COMPUTING AND VIRTUALIZATION**									
Course code	**24BCAO3102R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives	1. To elucidate both theoretical and practical aspects of cloud computing. 2. To equip students with the skills to design, implement, and manage cloud computing solutions effectively in both personal and professional settings. 3. To foster proficiency in evaluating cloud-based systems and optimizing them for enhanced performance and security.									
CO1	Understand cloud computing's fundamental concepts, including its history, architecture, and key advantages and disadvantages.									
CO2	Acquire knowledge of different cloud service and deployment models, assessing their impact on security and privacy.									
CO3	Comprehend virtualization techniques and their practical applications within cloud infrastructures.									
CO4	Explore the relationship between IoT and cloud computing, including emerging technologies like fog computing.									
CO5	Master the essentials of cloud security, focusing on risk management and protective measures for cloud-based systems.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Cloud Computing - An Introduction:** Introduction to Cloud Computing, History of Cloud Computing, Cloud Architecture, Cloud Storage, Why Cloud Computing Matters, Advantages of Cloud Computing, Disadvantages of Cloud Computing, Cloud Services.	**8**	Understand and describe the foundational aspects of cloud computing, including its history, architecture, storage solutions, and the overall significance, as well as the advantages and disadvantages of using cloud services.	**1,2**
II	**Cloud Deployment Models and Cloud Architecture:** Cloud reference model, Platform as service, Software as a service, Infrastructure as service, Cloud deployment models, public clouds, Private clouds, Community cloud, Hybrid clouds, security, trust and privacy	**8**	Understand and differentiate various cloud deployment models and architectures, including public, private, community, and hybrid clouds, along with cloud service models like Platform as a Service (PaaS), Software as a Service (SaaS), and Infrastructure as a Service (IaaS). Evaluate their implications for security, trust, and privacy.	**2, 4**
III	**Cloud Virtualization Technology:** Overview of Virtualization techniques, Types of Virtualizations, Implementation Levels of Virtualization Structures, Virtualization benefits, Server virtualization, Hypervisor management software, Virtual infrastructure requirements	**8**	Understand and explain various virtualization techniques and types, assess the benefits and levels of virtualization structures, and apply knowledge of server virtualization and hypervisor management to meet specific virtual infrastructure	**2, 3**

			requirements.	
IV	**IoT and the Cloud Computing** Introduction to IoT, Difference between Cloud Computing and IoT, Fog Computing: The Next Evolution of Cloud Computing, Role of Cloud Computing in IoT, Living on the Edge, An Abstract Edge Architecture Model, Connecting devices at the edge and to the cloud	**8**	Understand the fundamentals of IoT and its distinction from cloud computing, analyze the integration and role of cloud computing within IoT ecosystems, and apply concepts of fog computing and edge architectures to enhance connectivity and functionality between devices and the cloud.	**2, 3, 4**
V	**Cloud security:** Introduction to Security, Cloud Security challenges and Risks, Software-as-a-Service Security, Security Monitoring, Security Architecture Design, Data Security, Application Security, Virtual Machine Security, Identity Management and Access Control	**8**	Understand cloud security fundamentals and challenges; design and apply security measures for data, applications, and identity management; and evaluate the efficacy of these security solutions in cloud environments.	**3, 4, 5**

TEXT BOOKS:

T1: Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, August 2008

REFERENCE BOOKS:

R1: K.ChandraSekaran, "Essentials of Cloud Computing", 1st Edition, 2015, CRC Press, Taylor & Francis Group.

R2: A.Srinivasan and J.Suresh, "Cloud Computing, A practical approach for learning and implementation", Pearson, 2014.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand cloud computing's fundamental concepts, including its history, architecture, and key advantages and disadvantages.	3, 4
2	Acquire knowledge of different cloud service and deployment models, assessing their impact on security and privacy.	2, 3
3	Comprehend virtualization techniques and their practical applications within cloud infrastructures.	3
4	Explore the relationship between IoT and cloud computing, including emerging technologies like fog computing.	3, 4
5	Master the essentials of cloud security, focusing on risk management and protective measures for cloud-based systems.	2, 5, 7

SEMESTER – V									
Course Title	**PROJECT-I**								
Course code	**24BCAO3107R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30 P	**0**	**0**	**4**	**0**	**6**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives	1. To develop the knowledge, skills and attitudes of a professional Computer Application professional. 2. To become confident in designing Software solutions to complex software problems utilising a systems approach. 3. Develop a deep understanding of computer applications to enhance professional knowledge, skills, and attitudes necessary for the field.								
CO1	Demonstrate a sound technical knowledge of their selected project topic.								
CO2	Undertake problem identification, formulation and solution.								
CO3	Design Software solutions to complex problems utilising a systems approach.								
CO4	Communicate with the community at large in written an oral forms.								
CO5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: o theoretical studies o computer simulations o hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.	**90**	Analyze current knowledge through a comprehensive literature review in a chosen technical field, apply advanced methods like theoretical studies, computer simulations, and hardware construction. Synthesize findings into progress reports and professional journals, evaluate and present these findings in seminars, and demonstrate practical outcomes through poster presentations and operational demonstrations.	**3, 4, 5**

OTHER LEARNING RESOURCES:

As given by Project supervisor.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will be able to demonstrate a sound technical knowledge of their selected project topic.	1, 2, 4
2	Students will be able to undertake problem identification, formulation, and solution.	1, 3
3	Students will be able to design Software solutions to complex problems utilising a systems approach.	2, 3, 5, 7
4	Students will be able to communicate with the community at large in written an oral forms.	2, 3, 5, 6, 7
5	Demonstrate the knowledge, skills and attitudes of a Computer Application or IT professional.	1, 2, 3, 7

SEMESTER – V									
Course Title	**UI/UX DESIGN**								
Course code	**24BCAO3104R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15T+30P	**1**	**0**	**4**	**0**	**6**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Winter / I semester of third year**								
Course Objectives	1. To introduce students to industry-standard UI/UX design tools such as Figma and Visily, providing hands-on experience in creating wireframes, prototypes, and interactive designs. 2. To introduce the fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, enabling them to create aesthetically pleasing and user-friendly interfaces. 3. To guide in creating and implementing design systems, ensuring consistency and reusability of UI components across various projects, and promoting efficient team collaboration.								
CO1	Demonstrate proficiency in using Figma and Visily for designing wireframes, prototypes, and interactive UI components, leveraging these tools to create user-centered designs.								
CO2	Understand and apply fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, to create visually appealing and functional interfaces.								
CO3	Develop interactive prototypes and user flows, using advanced features of Figma and Visily to simulate real-world user interactions and test design hypotheses.								
CO4	Create and implement comprehensive design systems, ensuring consistency and reusability of UI components across multiple projects, and facilitating effective collaboration within design teams.								
CO5	Integrate UI/UX designs with development workflows, exporting assets and collaborating with developers to ensure seamless transition from design to implementation, and managing content within CMS or E-commerce platforms								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to UI/UX Design:** Definition of UI and UX, Importance of UI/UX Design, Key Differences between UI and UX; User Research Methods, Creating User Personas, User Journey Mapping; Introduction to Design Thinking, Phases of Design Thinking, Empathy in Design; Types of Prototypes, Tools for Prototyping, Low-fidelity vs. High-fidelity Prototypes.	**8**	Understanding the concept of UI/UX design, differentiate between UI and UX, understanding	**1, 4**
II	**UI Design Fundamentals:** Introduction to Figma, Sketch, Visily, and Adobe XD; Color Theory, Typography, Layout and Composition; Buttons, Icons, Forms, and Input Fields; Responsive Design, Mobile-First Design, Grid Systems.	**10**	Understand the theory of color, typography with different tools of designing. Apply buttons, icons, forms, and inputs. Creating responsive design.	**2, 3, 5**
III	**UX Design Fundamentals:** User Research Techniques - Surveys, Interviews, Focus Groups, Usability Testing; Information Architecture - Site Mapping, Card Sorting, Navigation Design; Interaction Design - Principles of Interaction Design, Designing User Flows, Micro-interactions; Wireframing and Mockups - Importance of Wireframing, Tools for Wireframing, Creating High-fidelity Mockups	**10**	Applying research techniques through surveys, interviews, focus groups. Designing navigation, sorting card.	**3, 4, 5**
IV	**Advance UI/UX Design Concepts:** Accessibility in design - Principles of Accessible Design, WCAG Guidelines,	**12**	Understanding the accessibility design, using tools for checking the accessibility, creating a	**2,3, 4,5**

	Tools for Checking Accessibility; Design Systems - What is a Design System, Benefits of Design Systems, Creating a Design System; Advanced Prototyping Techniques - Interactive Prototypes, Animation and Transitions, Tools for Advanced Prototyping; Usability Testing and Evaluation - : Planning a Usability Test, Conducting Usability Tests, Analyzing Results.		design system, and interactive prototypes, animations and transitions, analyzing the results.	
V	**UI/UX Designing and Portfolio Development:** Project Planning and Scope - Defining Project Goals, creating a Project Plan, Setting Milestones; Design Execution - Iterative Design Process, Collaboration and Feedback, Design Handoff; Building a Portfolio - Importance of a Portfolio, Elements of a Strong Portfolio, Showcasing Projects; Final Presentation.	**12**	Creating a project by applying all the techniques and tools.	**3, 5**
Practical				
Practical 1	Create a simple wireframe for a landing page using basic shapes and text in Figma. Explore Basic Tools and Interface Components in Figma	**2**	Gain familiarity with Figma's interface and basic tools. Understand how to use shapes and text to create a simple wireframe.	**1,5**
Practical 2	Design a Simple Wireframe for a Landing Page. Plan and sketch a wireframe layout for a landing page.	**2**	Practice planning and sketching wireframe layouts. Learn to structure content effectively for a landing page.	**5`**
Practical 3	Create multiple layers by adding different shapes and text. Rename layers and organize them using frames and groups. Use the layers panel to reorder and hide/show layers.	**2**	Understand how to manage layers and organize them using frames and groups in Figma. Learn efficient workflow practices.	**2, 5**
Practical 4	Use Figma's layout grids and frames to structure the wireframe. Add placeholders for images, text blocks, and buttons.	**2**	Master the use of layout grids and frames to structure wireframes effectively. Practice adding placeholders for various content elements.	**3**
Practical 5	Define a color palette using Figma'scolor picker or hex codes. Apply colors to different elements (background, text, shapes) in your wireframe. Use the color styles panel to create reusable color swatches	**2**	Develop skills in defining and applying cohesive color palettes. Learn to use Figma'scolor styles for efficient design consistency.	**5**
Practical 6	Design a component (e.g., card, button) using Figma's vector tools. Convert the component into an Auto Layout frame. Resize the frame to see how elements adjust automatically.	**2**	Create and manipulate components using Figma's Auto Layout feature. Understand how to design responsive UI elements.	**3**
Practical 7	Create a new frame sized for a mobile screen (e.g., iPhone X). Design UI components using adaptive variants for different screen sizes. Preview the design in Figma's device preview mode.	**2**	Practice designing UI components for mobile screens. Learn to preview designs and adjust for different device sizes in Figma.	**3**
Practical 8	Design multiple screens for a simple app or	**2**	Gain proficiency in prototyping	**4**

	website flow. Use Figma's prototyping feature to link screens together. Define interactions (e.g., click, hover) to simulate user navigation		with Figma. Learn to link screens, define interactions, and simulate user navigation.	
Practical 9	Design custom icons or illustrations using Figma's vector tools. Experiment with stroke styles, fills, and path operations. Use these assets to enhance your wireframes or UI designs.	**2**	Develop skills in designing custom icons and illustrations using Figma's vector tools. Understand stroke styles and path operations.	**5**
Practical 10	Define a design system for consistent UI elements (buttons, forms, typography). Create component libraries and style guides using Figma's design tokens. Share the design system with your team for reuse across projects.	**2**	Learn to create and manage design systems in Figma. Practice using design tokens and component libraries for consistent UI elements.	**6**
Practical 11	Set up design handoff using Figma plugins or integrations (e.g., Zeplin, Avocode). Export CSS, XML, or JSON code snippets for developers.	**2**	Explore Figma plugins for design handoff. Learn to export CSS, XML, or JSON code snippets for developers.	**6**
Practical 12	Creating sitemaps and user flows.	**2**	Practice creating sitemaps and user flows. Understand how to visually represent website or app structures and user interactions.	**5**
Practical 13	Designing user flows, Interactive wireframes, User Flow Design in Figma	**2**	Develop skills in designing interactive wireframes and user flows using Figma's prototyping features.	**4**
Practical 14	Use Figma's prototyping and animation features to create and simulate micro-interactions within a mobile app interface.	**2**	Learn to create and simulate micro-interactions and animations within a mobile app interface using Figma's prototyping tools.	**5**
Practical 15	Create two variants of the same web page in Figma, focusing on different design elements (e.g., button placement, color schemes) for A/B testing.	**2**	Practice creating different variants of web pages for A/B testing. Understand the impact of design elements on user behavior.	**6**
Practical 16	Use Figma's vector tools to create and customize basic shapes (rectangles, circles, lines) and combine them to form UI elements.	**2**	Master Figma's vector tools to create and customize basic UI elements like rectangles, circles, and lines.	**5**
Practical 17	Set up a grid system in Figma and use guides to ensure consistent spacing and alignment across your design.	**2**	Develop skills in setting up grid systems and using guides for consistent spacing and alignment in Figma designs.	**3**
Practical 18	Use Figma's text tools to add, style, and organize text elements, and experiment with different fonts and typographic hierarchies.	**2**	Practice adding, styling, and organizing text elements. Experiment with fonts and typographic hierarchy in Figma.	**3**
Practical 19	Import and position images and video placeholders, and adjust their properties to create an engaging design.	**2**	Learn to import, position, and adjust properties of images and video placeholders in Figma designs.	**3**
Practical 20	Create a Visily Account and Start Your First Project, add basic shapes and text elements to the canvas.	**2**	Gain familiarity with Visily's interface and basic tools. Learn to add and customize UI	**1,5**

			elements using drag-and-drop.	
Practical 21	Explore Basic Tools and Interface Components in Visily: Use Visily's drag-and-drop tools to add and customize UI elements. Arrange and group elements for organization.	**2**	Practice designing form layouts and implementing input validation rules using Visily's form elements.	**5**
Practical 22	Create Interactive Forms with Input Fields and Validation. Design a form layout using Visily's form elements (text inputs, checkboxes). Implement input validation rules (e.g., required fields, email format).	**2**	Learn to plan and structure wireframes using Visily's layout grids and containers. Incorporate placeholder content effectively.	**3**
Practical 23	Plan a wireframe layout for a website homepage (header, hero section, features). Use Visily's layout grids and containers to structure the wireframe. Add placeholder content such as images and text blocks.	**2**	Develop skills in defining color schemes and applying colors to various elements across pages using Visily.	**3**
Practical 24	Define a color scheme using Visily'scolor picker or palette options. Apply colors to different elements (background, text, buttons) across pages.	**2**	Practice creating navigation structures and interactive components like dropdown menus in Visily. Apply hover effects for interactive navigation.	**3**
Practical 25	Create a navigation bar or menu structure in Visily. Add dropdown menus using Visily's interactive components. Apply hover effects to menu items for interactive navigation.	**2**	Master Visily's grid and flexbox features to create responsive grid layouts for webpages or app screens.	**3**
Practical 26	Create a responsive grid layout in Visily for a webpage or app screen. Use Visily's grid and flexbox features to align elements across different screen sizes.	**2**	Develop proficiency in designing multiple pages and linking them together using interactive elements in Visily.	**4**
Practical 27	Design multiple pages for a website prototype in Visily. Link pages together using interactive elements (e.g., buttons, text links). Define transition effects and animations for a seamless user experience.	**2**	Learn to design and convert templates into reusable components in Visily for efficient project workflow.	**5**
Practical 28	Create a responsive grid layout in Visily for a webpage or app screen. Use Visily's grid and flexbox features to align elements across different screen sizes.	**2**	Practice integrating Visily with Content Management Systems (CMS) or E-commerce platforms for efficient content management.	**3**
Practical 29	Design reusable templates (e.g., headers, footers) in Visily. Convert templates into custom components for easy reuse across projects.	**2**	Design and develop reusable templates in Visily, converting them into custom components for consistent and efficient use across multiple projects.	**5**
Practical 30	Integrate Visily with CMS or E-commerce Platforms for Content Management	**2**	Integrate Visily designs with CMS or E-commerce platforms, enhancing their ability to manage and deploy content effectively in real-world applications.	**3**

TEXT BOOKS:

T1: "Don't Make Me Think, Revisited: A Common-Sense Approach to Web Usability" by Steve Krug."
T2: "The Design of Everyday Things" by Don Norman."

T3: "Interaction Design: Beyond Human-Computer Interaction" by Yvonne Rogers, Helen Sharp, and Jenny Preece”

REFERENCE BOOKS:
R1: "Sketching User Experiences: Getting the Design Right and the Right Design" by Bill Buxton”
R2: "Designing Interfaces: Patterns for Effective Interaction Design" by Jenifer Tidwell”

OTHER LEARNING RESOURCES:
https://www.interaction-design.org/courses/user-experience-the-beginner-s-guide
https://uxplanet.org/
https://material.io/design/Software/Hardware Tools to be used in
"Figma for UI Design" (Online Course Material)

MOOCS:
https://www.coursera.org/learn/ux-design-fundamentals
https://www.coursera.org/learn/introtoux-principles-and-processes?specialization=michiganux
https://www.coursera.org/learn/visual-elements-user-interface-design https://www.udemy.com/course/ui-ux-web-design-using-adobe-xd/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate proficiency in using Figma and Visily for designing wireframes, prototypes, and interactive UI components, leveraging these tools to create user-centered designs.	2,3,6,7,8
2	Understand and apply fundamental UI/UX design principles, including layout, color theory, typography, and responsive design, to create visually appealing and functional interfaces.	2,3,6,8
3	Develop interactive prototypes and user flows, using advanced features of Figma and Visily to simulate real-world user interactions and test design hypotheses.	2,3,6,7,8
4	Create and implement comprehensive design systems, ensuring consistency and reusability of UI components across multiple projects, and facilitating effective collaboration within design teams.	2,3,6,7,8
5	Integrate UI/UX designs with development workflows, exporting assets and collaborating with developers to ensure seamless transition from design to implementation, and managing content within CMS or E-commerce platforms	2,3,6,7,8

SEMESTER – V									
Course Title	**SUMMER INTERNSHIP**								
Course code	**24BCAO3110R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**12**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Application**								
Semester	**Fall/ V semester of the third year of the program**								
Course Objectives	1. Apply academic knowledge and develop industry-specific skills in a professional setting. 2. Gain insight into career pathways and enhance problem-solving, communication, and teamwork skills. 3. Build a professional network and understand workplace dynamics.								
CO1	Demonstrate the application of theoretical knowledge in a practical environment.								
CO2	Identify and develop key professional skills relevant to the field of study.								
CO3	Reflect on personal and professional growth throughout the internship.								
CO4	Produce professional documents, such as a resume, cover letter, and portfolio.								
CO5	Present the internship experience and learning outcomes effectively.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Orientation and Preparation:** Introduction to the internship, setting goals, understanding workplace expectations, and professional behaviour.	**8**	Demonstrate the application of theoretical knowledge in a practical environment.	**1, 2**
II	**Skills Development:** Developing industry-specific skills, working on assigned projects, and regular feedback sessions.	**8**	Identify and develop key professional skills relevant to the field of study.	**2, 3**
III	**Professional Growth:** Understanding workplace dynamics, building a professional network, and learning from mentors and colleagues.	**8**	Reflect on their personal and professional growth throughout the internship.	**2, 3**
IV	**Professional Documentation**: Creating and refining professional documents such as resumes, cover letters, and portfolios.	**8**	Produce professional documents, such as a resume, cover letter, and portfolio.	**3, 4**
V	**Presentation and Reflection:** Presenting the internship experience, discussing learning outcomes, and reflecting on personal growth and future career plans.	**8**	Present their internship experience and learning outcomes effectively.	**3, 4, 5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate the application of theoretical knowledge in a practical environment.	1, 2
2	Identify and develop key professional skills relevant to their field of study.	2, 3, 7
3	Reflect on their personal and professional growth throughout the internship.	8
4	Produce professional documents, such as a resume, cover letter, and portfolio.	6
5	Present their internship experience and learning outcomes effectively.	6

Detailed Syllabus of 6th Semester

SEMESTER – VI										
Course Title	**COMPUTER VISION**									
Course code	**24BCAO3201R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+15P	**2**	**1**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Programming Skills, Mathematics	**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application**								
Semester	**VI**								
Course Objectives	1. Introduce the fundamentals of computer vision and its applications. 2. Equip students with the skills to acquire, pre-process, and analyse image and video data. 3. Explore various techniques for image feature extraction, segmentation, and object recognition.								
CO1	Explain the essential concepts of computer vision, including its applications, challenges, and limitations, drawing comparisons with human vision.								
CO2	Classify the different image feature extraction techniques (edge detection, corner/blob detection, texture analysis, shape descriptors) to extract relevant information from images.								
CO3	Develop various image segmentation algorithms (thresholding, region-based, edge-based, morphological) to partition images into meaningful regions for further analysis.								
CO4	Justify supervised and unsupervised learning methods (k-NN, SVM, neural networks, K-means, PCA) for object recognition and classification tasks in images.								
CO5	Examine the diverse applications of computer vision in various domains like image retrieval, medical imaging, robotics, and security systems, highlighting its potential impact.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Computer Vision:** Definition, history, and applications of computer vision, the human visual system vs. computer vision Images in the Spatial Domain: Pixels and Windows	**6**	**Describe and explain Computer vision and Spatial Domain of Images**	**1, 2**
II	Image Formation: 2D transformations, 3D transformations, 3D rotations, 3D to 2D projections, Photometric image formation, The digital camera, Sampling and aliasing, Compression.	**8**	**Describe, Illustrate and Analyse the Image Formation and their different form**	**1,2,4**
III	Image processing: Point operators, Linear filtering, Fourier transforms, Pyramids and wavelets, Parametric transformations, Mesh-based warping, Application: Feature-based morphing	**8**	**Explain and Applying Image processing like Fourier transforms, Parametric transform etc**	**2, 3**
IV	Feature detection and matching: Points and patches, Performance-driven animation, Edge detection, Edge linking, Vanishing points Image **Segmentation,Snakes** Image classification, Application: Visual similarity search	**8**	**Describe and Applying Feature detection and matching in different images**	**2, 3**
V	Applications of Computer Vision: Image and video retrieval: searching for specific images or videos in a database, Medical imaging: diagnosis and analysis of medical images, Surveillance and security: object tracking and anomaly detection	**6**	**Describe and Applying Various Computer Vision techniques like Searching, Medical images diagnosis, etc**	**2, 3**

Practical				
Practical 1	**Image Loading and Display:** Learn how to read and display images using OpenCV.	**2**	Describe, illustrate and explain and apply different imaging techniques and carry out practical in lab.	**1,2,3, 4**
Practical 2	Image Grayscale Conversion: Understand the process of converting color images to grayscale.	**2**	Understand the importance of grayscale conversion, implement color-to-grayscale transformation using OpenCV, and analyze its impact on image processing.	**3,4**
Practical 3	Image Thresholding: Explore thresholding techniques for image binarization.	**2**	Explore different thresholding techniques (binary, adaptive, Otsu's), apply thresholding for image segmentation, and analyze their effects on images.	**3,4**
Practical 4	Image Smoothing and Blurring: Implement various filters for smoothing and blurring images.	**2**	Implement Gaussian, median, and bilateral filters for noise reduction, and evaluate their effectiveness in image enhancement.	**3,5**
Practical 5	Edge Detection: Detect edges in images using Canny edge detection and other methods.	**2**	Detect edges using Canny, Sobel, and Laplacian operators, and analyze edge detection results for feature extraction.	**3,4**
Practical 6	Image Morphological Operations: Apply morphological operations like erosion and dilation for image processing.	**2**	Apply erosion, dilation, opening, and closing operations to enhance or remove image features, and understand their role in image pre-processing.	**3,4**
Practical 7	Image Geometric Transformations: Perform scaling, rotation, and other geometric transformations on images.	**2**	Perform image scaling, rotation, translation, and affine transformations, and analyze their use in computer vision applications.	**3,4**
Practical 8	Contour Detection and Object Recognition: Find contours in images and use them for object recognition.	**2**	Extract contours from images using OpenCV and apply them for object recognition and shape analysis.	**3,4**
Practical 9	Histogram Analysis and Equalization: Analyze image histograms and perform histogram equalization for image enhancement.	**2**	Understand image histograms, apply histogram equalization for contrast enhancement, and evaluate its impact on image clarity.	**4,5**
Practical 10	Template Matching: Find occurrences of a template image within a larger image.	**2**	Detect specific objects within images using template matching techniques, and compare different matching methods.	**4,5**
Practical 11	Feature Detection and Keypoints: Detect keypoints like corners and Harris corners in images.	**2**	Detect keypoints like Harris corners and Shi-Tomasi points, and analyze their importance in feature	**3,4**

			extraction.	
Practical 12	Feature Matching and Homography: Match features between images and estimate homography for image registration.	**2**	Perform feature matching using SIFT, ORB, or SURF, and estimate homography for aligning images.	**3,5,6**
Practical 13	Image Pixel Manipulation and Display with OpenCV.	**2**	Learn how to access and modify pixel values in an image, and apply transformations based on pixel intensity.	**3,4**
Practical 14	Python program to draw various shapes.	**2**	Implement OpenCV functions to draw basic shapes like lines, rectangles, and circles for graphical applications.	**3,6**
Practical 15	Python program to draw a simple bullseye with the cv2.circle function.	**2**	Use OpenCV to generate circular patterns, and explore its applications in visual representation.	**3,6**
Practical 16	Program to perform various transformation in an image.	**2**	Apply geometric transformations like perspective shift, shearing, and flipping, and analyze their effects.	**3,4**
Practical 17	Python program that finds and draws contours of objects in an image.	**2**	Detect, analyze, and highlight object boundaries using contour detection methods.	**3,4**
Practical 18	Detecting Blue Objects.	**2**	Use color thresholding techniques to detect objects of a specific color in an image.	**3,4**
Practical 19	Program for detecting face.	**2**	Implement face detection using Haar cascades or DNN models and analyze its accuracy.	**3,5**
Practical 20	Program for Face and Smile Detection.	**2**	Combine facial recognition with smile detection to develop real-time emotion tracking applications.	**3,5,6**
Practical 21	Image segmentation based on color	**2**	Implement color-based segmentation using K-means clustering, watershed algorithm, or thresholding.	**3,5**
Practical 22	Image Color Conversion: Convert the color space of the image (e.g., RGB to grayscale, HSV, LAB, etc.) and analyze the channels separately.	**2**	Convert images between different color spaces (RGB, HSV, LAB) and analyze how different color representations affect processing.	**3,4**

Text Books:

1. Szeliski, Richard. *Computer Vision: Algorithms and Applications*. Springer Nature, 2022.
2. Davies, E. Roy. *Computer and Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, 2012.
3. Gonzalez, Rafael C. *Digital Image Processing*. Pearson Education India, 2009.

Reference Books:

1. Forsyth, David A., and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
2. Brownlee, Jason. *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, And Data Transforms In Python*. Machine Learning Mastery, 2020.

Additional Resources:

- OpenCV Tutorials and Documentation: https://opencv.org/
- Python libraries for computer vision: Scikit-image, TensorFlow, PyTorch
- https://nptel.ac.in/courses/117105079
- https://nptel.ac.in/courses/106105216/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain the essential concepts of computer vision, including its applications, challenges, and limitations, drawing comparisons with human vision.	1, 2, 7 and 8
2	Classify the different image feature extraction techniques (edge detection, corner/blob detection, texture analysis, shape descriptors) to extract relevant information from images.	1, 2, 7 and 8
3	Develop various image segmentation algorithms (thresholding, region-based, edge-based, morphological) to partition images into meaningful regions for further analysis.	1, 2, 7 and 8
4	Justify supervised and unsupervised learning methods (k-NN, SVM, neural networks, K-means, PCA) for object recognition and classification tasks in images.	1, 2, 7 and 8
5	Examine the diverse applications of computer vision in various domains like image retrieval, medical imaging, robotics, and security systems, highlighting its potential impact.	1, 2, 7 and 8

SEMESTER – VI									
Course Title	**Introduction to Cybersecurity**								
Course code	**24BCAO3202R**	**Total credits: 4** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite		**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives	1. To understand the foundational concepts of cyberspace, including the architecture of the internet, web technologies, and the regulatory frameworks governing cyberspace.. 2. To identify and classify various types of cybercrimes, comprehend cybercriminals' methodologies, and analyze the legal aspects of cybercrimes under the IT Act 2000 and its amendments. 3. To explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption algorithms, and demonstrate knowledge of Public Key Infrastructure (PKI) and digital certificates.								
CO1	Understand the foundational concepts of cyberspace, including the architecture of the internet, communication protocols, and governance structures.								
CO2	Identify and classify various types of cybercrimes, their modus operandi, and the legal frameworks governing cybercrimes in India.								
CO3	Explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption, and understand the role of PKI and digital certificates.								
CO4	Analyze the security challenges and best practices in e-commerce, including threats, digital payment mechanisms, and regulatory guidelines.								
CO5	Implement cybersecurity risk management strategies, including risk assessment methodologies, incident response planning, and compliance with security policies and procedures.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Cybersecurity:** Defining Cyberspace and Overview of Computer and Web-technology, Architecture of cyberspace, Communication and web technology, Internet, World wide web, Advent of internet, Internet infrastructure for data transfer and governance, Internet society, Regulation of cyberspace, Concept of cyber security, Issues and challenges of cyber security.	**9**	After completion of this module, students would be able to understand the concept of Cyber security and issues and challenges associated with it.	**1,2,3**
II	**Cybercrime and Cyber law:** Classification of cybercrimes, Common cybercrimes-cybercrime targeting computers and mobiles, cybercrime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Cybercriminals modus-operandi , Reporting of cybercrimes, Remedial and mitigation measures, Legal perspective of cybercrime, IT Act 2000 and its amendments, Cybercrime and	**8**	Students, at the end of this module, should be able to understand the cybercrimes, their nature, legal remedies and as to how report the crimes through available platforms and procedures.	**2,3,4**

	offences, Organisations dealing with Cybercrime and Cyber security in India, Case studies.			
III	**Cryptography and Encryption:** Fundamentals of cryptography and encryption algorithms Symmetric vs. asymmetric encryption Public key infrastructure (PKI) and digital certificates.	**10**	Explain network security principles, assess vulnerabilities, and implement measures to secure networks against external and internal threats.	**2,3,4**
IV	**E-Commerce and Digital Payments:** Definition of E-Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, E-Commerce security best practices, Introduction to digital payments, Components of digital payment and stake holders, Modes of digital payments- Banking Cards, Unified Payment Interface (UPI), e-Wallets, Unstructured Supplementary Service Data (USSD), Aadhar enabled payments, Digital payments related common frauds and preventive measures. RBI guidelines on digital payments and customer protection in unauthorised banking transactions. Relevant provisions of Payament Settlement Act,2007,	**9**	After the completion of this module, students would be able to understand the basic concepts related to E-Commerce and digital payments. They will become familiar with various digital payment modes and related cyber security aspects, RBI guidelines and preventive measures against digital payment frauds.	**2,3,4**
V	**Cybersecurity Risk Management, Digital Devices Security, Tools and Technologies for Cyber Security:** Risk assessment methodologies and frameworks Security policies, procedures, and compliance Incident response and disaster recovery planning; End Point device and Mobile phone security, Password policy, Security patch management, Data backup, Downloading and management of third-party software, Device security policy, Cyber Security best practices, Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus, Wi-Fi security, Configuration of basic security policy and permissions.	**9**	Discuss cybersecurity governance frameworks, analyze regulatory requirements, and develop policies to align with organizational goals and ensure compliance. Understand the basic security aspects related to Computer and Mobiles. They will be able to use basic tools and technologies to protect their devices.	**1,2,3**

Practical				
Practical 1	Implement Caesar cipher encryption in Python and decrypt a given text.	**2**	Implementing Caesar cipher encryption in Python enables students to understand classical cryptographic techniques, apply encryption and decryption logic, and analyze the security limitations of substitution ciphers.	**3,4**
Practical 2	Explain the differences between symmetric and asymmetric encryption with examples.	**2**	Explaining the differences between symmetric and asymmetric encryption helps students understand their fundamental principles, compare their use cases, and evaluate their advantages and disadvantages in real-world applications.	**2,5**
Practical 3	Write a Python program to encrypt and decrypt text using AES encryption.	**2**	Writing a Python program to encrypt and decrypt text using AES encryption allows students to apply advanced cryptographic algorithms, manage encryption keys securely, and analyze the importance of block ciphers in data protection.	**3,6**
Practical 4	Generate an RSA key pair using OpenSSL and encrypt/decrypt a sample message.	**2**	Generating an RSA key pair using OpenSSL and encrypting/decrypting a sample message enables students to understand public-key cryptography, apply RSA encryption for secure communication, and evaluate key management best practices.	**3,5**
Practical 5	Verify the SSL certificate of a website. Identify the Certificate Authority (CA) and encryption algorithm used.	**2**	Verifying the SSL certificate of a website helps students analyze HTTPS encryption, identify the Certificate Authority (CA), and evaluate the role of TLS security in web communication.	**4,5**
Practical 6	Use GPG (GNU Privacy Guard) to encrypt and sign an email message.	**2**	Using GPG (GNU Privacy Guard) to encrypt and sign an email message enables students to implement secure email communication, apply digital signatures, and evaluate the importance of end-to-end encryption.	**3,5**
Practical 7	Analyze the encryption methods used in messaging apps like WhatsApp.	**2**	Analyzing the encryption methods used in messaging apps like WhatsApp allows students to understand end-to-end encryption (E2EE), explore secure key exchange mechanisms, and evaluate the effectiveness of encrypted communication.	**4,5**
Practical 8	Analyze a real-world E-Commerce fraud case and suggest preventive measures.	**2**	Analyzing a real-world E-Commerce fraud case and suggesting preventive measures helps students understand financial cybercrime, assess fraud detection techniques, and propose security solutions for online transactions.	**4,5**

Practical 9	Conduct a mock UPI transaction and analyze the security features in Google Pay/PhonePe.	**2**	Conducting a mock UPI transaction and analyzing the security features in Google Pay/PhonePe allows students to explore digital payment security, identify authentication methods, and evaluate fraud prevention measures in financial transactions.	**3,5**
Practical 10	Analyze a phishing website pretending to be a bank portal using SSL Labs security scanner.	**2**	Analyzing a phishing website pretending to be a bank portal using SSL Labs security scanner enables students to detect fraudulent websites, evaluate SSL/TLS security configurations, and understand how cybercriminals exploit weaknesses in online banking.	**4,5**
Practical 11	Perform a risk assessment for an organization's IT infrastructure. What are the potential threats?	**2**	Performing a risk assessment for an organization's IT infrastructure helps students identify cybersecurity threats, evaluate risk mitigation strategies, and propose security controls for protecting sensitive information.	**4,5**
Practical 12	Create a basic cybersecurity policy for an educational institution.	**2**	Creating a basic cybersecurity policy for an educational institution allows students to develop security guidelines, define access control measures, and ensure compliance with best practices in data protection.	**5,6**
Practical 13	Configure and analyze Windows Defender Firewall settings.	**2**	Configuring and analyzing Windows Defender Firewall settings enables students to manage host-based security, implement firewall rules, and evaluate the effectiveness of network protection measures.	**3,5**
Practical 14	Develop an incident response plan for a data breach scenario.	**2**	Developing an incident response plan for a data breach scenario helps students understand forensic investigation, define response strategies, and evaluate compliance requirements for cybersecurity incidents.	**5,6**
Practical 15	Write a Python script to automate data backups and store them securely.	**2**	Writing a Python script to automate data backups and store them securely allows students to implement secure backup solutions, explore encryption-based data protection, and evaluate best practices for disaster recovery.	**3,6**

Text Books:

T1: Introduction to Computer Security" by Michael T. Goodrich and Roberto Tamassia
T2: "Cybersecurity: A Practical Guide to the Law of Cyber Risk" by Andrew Serwin
T3: Network Security Essentials: Applications and Standards" by William Stallings

Reference Books:

R1: "Cryptography and Network Security: Principles and Practice" by William Stallings
R2:"Security Engineering: A Guide to Building Dependable Distributed Systems" by Ross Anderson
R3: "Principles of Computer Security: CompTIA Security+ and Beyond" by Wm. Arthur Conklin, Greg White, Chuck Cothren, Roger L. Davis, Dwayne Williams

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the foundational concepts of cyberspace, including the architecture of the internet, communication protocols, and governance structures.	1,2,3,4,5,6,7, and 8
2	Identify and classify various types of cybercrimes, their modus operandi, and the legal frameworks governing cybercrimes in India.	1,2,3,4,5,6,7, and 8
3	Explain the fundamentals of cryptography, distinguish between symmetric and asymmetric encryption, and understand the role of PKI and digital certificates.	1,2,3,4,5,6,7, and 8
4	Analyze the security challenges and best practices in e-commerce, including threats, digital payment mechanisms, and regulatory guidelines.	1,2,3,4,5,6,7, and 8
5	Implement cybersecurity risk management strategies, including risk assessment methodologies, incident response planning, and compliance with security policies and procedures.	1,2,3,4,5,6,7, and 8

SEMESTER – VI									
Course Title	**PE IV: Exploratory Data Analysis**								
Course code	**24BCAO3203R**	**Total credits: 4 Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Probability and Statistics**	**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives	1.To gain insights and understanding from a dataset through visual and quantitative methods. 2.To identify patterns, relationships, and anomalies in the data, this can inform further analysis or decision-making. 3.To extract meaningful information from data and communicate it effectively to stakeholders.								
CO1	Demonstrate the importance of exploratory data analysis in data analysis and decision-making.								
CO2	Develop skills in data visualization using various tools and techniques.								
CO3	Learn how to effectively communicate insights and findings from data analysis.								
CO4	Apply EDA techniques to real-world data analysis problems.								
CO5	Enhance critical thinking skills to identify patterns, trends, and outliers in data.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Exploratory Data Analysis** EDA Introduction, what is EDA? EDA vs Classical & Bayesian, EDA vs Summary, EDA Goals, The Role of Graphics, An EDA/Graphics Example, General Problem Categories.	**6**	Explain EDA and understand and Apply EDA in different datasets with Examples.	**1, 2, 3**
II	**Underlying Assumptions** Importance, Techniques for Testing Assumptions, Interpretation of 4-Plot, Consequences.	**6**	Understand and Applying Different Testing Assumptions and Analysis with Graph.	**2, 3, 4**
III	**EDA Techniques** Introduction, Analysis Questions, Graphical Techniques: Alphabetical, Graphical Techniques: By Problem Category, Quantitative Techniques, Probability Distributions.	**8**	Understanding and Applying different datasets and analyses with Graph Quantitatively.	**2, 3, 4**
IV	**EDA Case Studies** Case Studies Introduction, Case Studies: Normal random numbers, Uniform random numbers, Random walk, Josephson Junction Cryo thermometry, Beam Deflections, .Filter Transmittance, Standard Resistor, Heat Flow Meter 1, Airplane Glass Failure Time, Ceramic Strength.	**10**	Understanding different case study and Applying with suitable example. Analyze with graphs	**2 , 3, 4**
V	**Data Visualization** Design principles for charts and graphs, ggplot2 and Tableau tools for creating data visualizations, The process creating visualizations and selecting the appropriate visual display, Designing effective digital presentations, Visualization as exploration, Visualizing categorical data, Visualizing time series data, Visualizing multiple variables, Visualizing geospatial data, Dashboard design, Web- based visualizations, Interactive visualizations and motion.	**6**	Understanding the different charts, graphs and their interpretations. And Analyses with different datasets.	**2, 3, 4**
Practical				
Practical 1	Introduction to NumPy and Pandas for EDA	2	Understand basic operations in NumPy and Pandas for	2

			data manipulation	
Practical 2	Importing and Cleaning Datasets using Pandas	2	Load datasets, handle missing values, and preprocess data	3
Practical 3	Checking Data Distribution and Summary Statistics	2	Use NumPy and Pandas to calculate mean, median, mode, variance, and standard deviation	3
Practical 4	Identifying Outliers using Z-score and IQR	2	Detect and handle outliers in datasets using statistical methods	4
Practical 5	Data Transformation Techniques	2	Apply log transformation, min-max scaling, and standardization	3
Practical 6	Checking Data Distribution and Summary Statistics	2	Use NumPy and Pandas to calculate mean, median, mode, variance, and standard deviation	3
Practical 7	Identifying Outliers using Z-score and IQR	2	Detect and handle outliers in datasets using statistical methods	4
Practical 8	Data Transformation Techniques	2	Apply log transformation, min-max scaling, and standardization	3
Practical 9	Feature Engineering using Pandas	2	Create new features from existing datasets to improve analysis	4
Practical 10	Correlation Analysis between Variables	2	Compute and interpret correlation coefficients using Pandas	4
Practical 11	EDA on Real-World Dataset (Titanic Dataset)	3	Perform data exploration on a case study dataset	5
Practical 12	EDA on Real-World Dataset (Iris Dataset)	3	Explore dataset features and relationships	5
Practical 13	Handling Categorical Data and Encoding Techniques	2	Apply label encoding, one-hot encoding, and ordinal encoding	3
Practical 14	Univariate Data Visualization using Matplotlib & Seaborn	2	Create histograms, boxplots, and KDE plots for single variables	3
Practical 15	Bivariate Data Visualization using Seaborn	2	Use scatter plots, heatmaps, and pair plots to analyze relationships	4)
Practical 16	Time-Series Data Analysis and Visualization	2	Analyze trends, seasonality, and forecasting using Pandas & Matplotlib	4
Practical 17	Interactive Dashboards using Plotly	3	Create dynamic and interactive visualizations for data storytelling	5

Text Books:

1. Exploratory Data Analysis John Tukey Pearson;1st edition

Reference Books:

1. Engineering Statistics Handbook http://www.itl.nist.gov/div898/handbook/nist.gov

2. Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize and investigate your data Paperback Suresh Kumar Mukhiya, Usman Ahmed Packet Publishing Limited

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate the importance of exploratory data analysis in data analysis and decision-making.	1, 2 and 3
2	Develop skills in data visualization using various tools and techniques.	1, 2, 3, 5 and 8
3	Learn how to effectively communicate insights and findings from data analysis.	1, 2, and 3
4	Apply EDA techniques to real-world data analysis problems.	1, 2, 3, 5 and 7
5	Enhance critical thinking skills to identify patterns, trends, and outliers in data.	1, 2, 3, 5, 7 and 8

SEMESTER VI										
Course Title	**PE - IV Pattern Recognition**									
Course code	**24BCAO3209R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
		Total hours: 45T+15P	**3**	**0**	**2**	**0**	**0**	**0**	**4**	
Pre-requisite	Fundamentals of probability, linear algebra and Digital Image Processing	**Co-requisite**	NIL							
Programme	**Bachelor of Computer Application (BCA)**									
Semester	**Fall / II semester of the Second Year**									
Course Objectives	1.Pattern recognition which has several important applications on classical computer engineering problems like word/sentence-based searches, etc. 2.The emphasis of the course is on algorithms for pattern recognition 3.The representation of patterns and classes and the proximity measures are an important aspect of pattern recognition									
CO1	Summarize the various techniques involved in pattern recognition.									
CO2	Categorize the various pattern recognition techniques into supervised and unsupervised									
CO3	Understand feature selection and extraction techniques.									
CO4	Illustrate the artificial neural network-based pattern recognition									
CO5	Discuss the applications of pattern recognition in various real-world applications									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction and Mathematical Preliminaries**: Pattern Recognition: Definition, Applications and Examples, Clustering Vs Classification, Supervised Vs Unsupervised, Basic of Linear Algebra, Vector Spaces, Basics of Probability, Basics of Estimation Theory, Decision Boundaries, Decision Regions, Metric Spaces	**9**	Explain Pattern Recognition, Understanding different Clustering Techniques and their theory. Applying in real world scenario.	**1, 2, 4**
II	**Classification:** Bayes Decision Rules, Error Probability, Examples, Normal Distribution, Linear Discriminant Function, Non-Linear Decision Boundaries, Mahalanobis Distance, K-NN Classifier, Single and Multi-Layer Perceptron, Training Set, Test Set, Standardization and Normatization	**9**	Explaining different Classification technique and understanding their interpretation by applying in different datasets.	**1, 2, 4**
III	**Clustering:** Basics, Similarity/Dissimilarity Measures, Clustering Criteria, Different distance functions and similarity measures, within cluster distance criterion, K-means algorithm, Single linkage and complete linkage algorithms, MST,K-medoids, DBSCAN, Data sets: Visualization, Unique Clustering	**9**	Explaining different Clustering technique and understanding their interpretation by applying in different datasets.	**1, 2, 4**
IV	**Feature Selection and Extraction**: Problem statement and Uses, Branch and Bound Algorithm, Sequential Forward / Backward	**9**	Explaining different Feature Selection and Extraction technique and understanding their role in enhancing models.	**1, 2, 3, 4**

	Selection Algorithms, (l,r) algorithm, Probabilistic separability based criterion functions, interclass distance based criterion functions, Feature Extraction, PCA + Kernel PCA			
V	**Recent Advances in Pattern Recognition**: Structural PR, SVMs, FCM, Soft-Computing and Neuro-Fuzzy Techniques, Real-Life Examples	**9**	Understanding Recent Advance Pattern Recognition Technique with Real life Examples	**1, 2, 3, 4**
Practical				
Practical 1	Implement a basic pattern recognition algorithm in Python to classify points in a 2D plane based on their distance from the origin. Use the concept of Euclidean distance.	**2**	Applying pattern recognition algorithms to classify data based on distance metrics and visualize the results using Python libraries	**3, 5**
Practical 2	Write a Python function to generate random data points and visualize them using Matplotlib. Label the points with distinct colors based on their class (e.g., red and blue) using a simple threshold.	**2**	Creating random datasets, apply color-coding techniques to label data points, and visualize the data effectively using Python tools	**3, 5**
Practical 3	Create a Python script that performs vector space transformations (e.g., translation, scaling, and rotation) on a set of 2D points. Visualize the points before and after the transformations using Matplotlib.	**2**	Demonstrate an understanding of vector space transformations and apply them to datasets, followed by visualizing the transformations	**3, 5**
Practical 4	Implement Bayes' Theorem in Python to classify data points based on the likelihood of their attributes. Use Gaussian Naive Bayes for the classification task and apply it to a simple dataset (e.g., two-dimensional points with Gaussian distribution).	**2**	Implement Bayes' Theorem and apply Gaussian Naive Bayes for classification tasks	**3, 4**
Practical 5	Write a Python function to calculate the error rate of a classification model. Apply it to a set of predicted and actual class labels.	**2**	Calculate the error rate of a classification model and analyze the model's performance using Python	**3, 4**
Practical 6	Implement a Linear Discriminant Analysis (LDA) classifier in Python. Test the classifier on a 2D dataset and visualize the decision boundary using Matplotlib.	**2**	Apply Linear Discriminant Analysis (LDA) for classification and visualize decision boundaries	**3, 5**
Practical 7	Write a Python script to perform K-means clustering on a 2D dataset. Plot the resulting clusters and the cluster centroids. Use scikit-learn'sKMeans implementation.	**2**	Apply the K-means clustering algorithm to a dataset and visualize clusters and centroids	**3, 5**
Practical 8	Implement a Python function	**2**	Implement distance calculations and apply	**3, 4**

	to compute the Euclidean distance between two data points. Use this function to perform hierarchical clustering using agglomerative (single linkage) and divisive methods.		them in hierarchical clustering using Python	
Practical 9	Implement the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm in Python. Visualize the results of DBSCAN on a 2D dataset and compare it with K-means clustering.	**2**	Implement the DBSCAN algorithm, compare it with K-means, and visualize the clustering results	**3, 4, 6**
Practical 10	Write a Python function to perform feature selection using the sequential forward selection algorithm. Apply it to the Iris dataset and print the selected features.	**2**	Implement feature selection algorithms and apply them to real datasets	**3, 4**
Practical 11	Implement PCA (Principal Component Analysis) in Python from scratch to reduce the dimensionality of a dataset. Visualize the data before and after dimensionality reduction.	**2**	Implement PCA from scratch and visualize the effects of dimensionality reduction	**3, 5**
Practical 12	Write a Python script that applies feature normalization (e.g., Min-Max scaling or Z-score normalization) to a dataset and visualize the effect of normalization on the data distribution.	**2**	Apply feature normalization techniques and visualize the effect on data distribution	**3, 5**
Practical 13	Implement a basic Support Vector Machine (SVM) classifier using the scikit-learn library in Python. Train it on a simple dataset, such as the Iris dataset, and visualize the decision boundaries.	**2**	Implement an SVM classifier and visualize decision boundaries for model evaluation	**3, 5**
Practical 14	Write a Python program to implement Fuzzy C-Means (FCM) clustering using skfuzzy. Apply it to a 2D dataset and visualize the resulting fuzzy clusters.	**2**	Implement the Fuzzy C-Means algorithm and visualize fuzzy clusters in a dataset	**3, 5**
Practical 15	Implement a neuro-fuzzy system in Python using the anfis library (Adaptive Neuro-Fuzzy Inference System). Use it for a regression task on a simple dataset and visualize the results.	**2**	Implement a neuro-fuzzy system for regression tasks and visualize the results using Python	**3, 5**

Books:

1. J.I. Tou& R.C. Gonzalez, Pattern Recognition Principles, Addition-Wesley.
2. MR. Schalkoff, Pattern Recognition - Statistical, Structural and Neural Approaches, John Wiley, 1992.

Reference Books:

1. P.A. Devijer& J. Kittler, Pattern Recognition - A Statistical Approach, Prentice-Hall.
2. Christopher. M. Bishop, 'Pattern recognition and machine learning, Springer, 2006.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Summarize the various techniques involved in pattern recognition.	1,2,3,7, and 8
2	Categorize the various pattern recognition techniques into supervised and unsupervised	1,2,3,7, and 8
3	Understand feature selection and extraction techniques.	1,2,3,7, and 8
4	Illustrate the artificial neural network based pattern recognition	1,2,3,7, and 8
5	Discuss the applications of pattern recognition in various real-world applications	1,2,3,5,7, and 8

SEMESTER – VI									
Course Title	**Applied Analytics in NLP**								
Course code	**24BCAO3205R**	**Total credits: 3** **Total hours: 45L+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**NIL**	**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives (Minimum 3)	1. To familiarize the concepts and techniques of Natural Language Processing for analysing words based on Morphology and CORPUS. 2. To introduce the fundamentals of language processing from the algorithmic point of view. 3. To apply the Statistical learning methods and cutting-edge research models from deep learning.								
CO1	Realize the principles and Processes of Human Languages such as English and other Indian Languages using computers.								
CO2	Describe the concepts of morphology, syntax, semantics, discourse, and pragmatics of natural language.								
CO3	Perform POS tagging for a given natural language and Select a suitable language modeling technique based on the structure of the language.								
CO4	Demonstrate advanced algorithms and techniques for text-based processing with respect to morphology.								
CO5	Develop Statistical Methods for Real World Applications and explore deep learning based NLP								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Introduction to Natural Language Processing, Ambiguities in language, Regular expression, Words, Morphology, Morphology parsing, Word tokenization, Lemmatization & stemming, Edit distance. Morphological analysis and generation using Finite State Automata and Finite State transducer.	**8**	Describing NLP and understanding the different NLP Processing techniques. Applying with examples.	**1, 2, 3**
II	Introduction, English word classes, Rule-based part of speech tagging, Transformation based part of speech tagging, Evaluation and error analysis, Issues Tag indeterminacy and tokenization. Applications of Tagging	**8**	Understanding different word rules, tags etc. Applying and analyzing different evaluation techniques.	**2, 3, 4**
III	Parsing Basic concepts: Statistical Parsing basics: Probabilistic Context Free Grammar; Probabilistic CKY Parsing of PCFGs. Problems with PCFGs. Real life example and related case study.	**8**	Applying and analysing different Grammar or Parsing Techniques with real world examples.	**3, 4**
IV	Lexical Semantics, Cosine for measuring similarity, TF-IDF N-Gram Models, Language Model Adaptation, Types of Language Models, and Language-Specific Modelling Problems. Lexicalized models using derivational histories and dependency-based models. Real-life example and related	**8**	Applying and Analysis Lexical Semantics different modeling techniques form case study.	**3, 4**

	case study.			
V	Introduction and Application to WordNet. Recent trends in NLP. Question Answering Systems, Social Network analysis.	**8**	Applying different applications and analyses by using different datasets.	**3, 4**
Practical				
Practical 1	Implementing Regular Expressions for Text Preprocessing	**2**	Understand how to use regex for text processing tasks like extracting patterns and filtering data	**3**
Practical 2	Word Tokenization using NLTK and Pandas	**2**	Learn how to split text into meaningful words for further processing	**3**
Practical 3	Lemmatization and Stemming using Pandas and SpaCy	**2**	Understand word normalization techniques for NLP tasks	**3**
Practical 4	Computing Edit Distance using NumPy	**2**	Implement edit distance algorithms like Levenshtein Distance for spelling correction	**4**
Practical 5	Morphological Analysis using Finite State Automata	**2**	Develop an FSA model for word morphology analysis	**4**
Practical 6	Part of Speech Tagging using Rule-based and Statistical Methods	**3**	Apply different tagging methods and analyze their accuracy using Pandas	**4**
Practical 7	Transformation-based POS Tagging and Error Analysis	**3**	Implement and evaluate transformation-based tagging using NumPy	**4**
Practical 8	Probabilistic Context-Free Grammar (PCFG) Parsing	**3**	Understand how PCFG works and implement a basic CKY parser	**4**
Practical 9	Parsing with Probabilistic CKY Algorithm	**3**	Use NumPy to develop a CKY parser and analyze its performance	**4**
Practical 10	Computing Cosine Similarity for Text Similarity	**3**	Apply vector space models to measure text similarity using TF-IDF	**3**
Practical 11	Implementing TF-IDF and N-Gram Models using Pandas	**3**	Learn how to generate and analyze N-Gram frequency distributions	**3**

Practical 12	Creating a Simple Language Model using NumPy	**3**	Build and evaluate an N-Gram-based language model	**4**
Practical 13	Using WordNet for Lexical Semantics	**3**	Extract synonyms, antonyms, and word relations from WordNet	**3**
Practical 14	Social Network Analysis for NLP Applications	**3**	Apply Pandas to analyze text-based interactions in social networks	**4**
Practical 15	Developing a Basic Question Answering System	**3**	Implement a simple QA system using Pandas and NumPy	**5**

Text Books:

1. Jurafsky Daniel and Martin James H. "Speech and Language Processing" ,3rd Edition, Pearson publications, 2018.
2. Christopher D. Manning and Hinrich Schutze, "Foundations of Natural Language Processing", 6th Edition, The MIT Press Cambridge, Massachusetts London, England, 2003.

Reference Books:

1. Jurafsky Daniel and Martin J. H., "Speech and language processing Natural Language Processing, Computational Linguistics,2nd Edition, Upper Saddle River, NJ: Prentice-Hall, 2008.
2. Jalaj Thanaki, Python Natural Language Processing: Explore NLP with machine Learning and deep learning Techniques, Packt, 2017.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Realize the principles and Processes of Human Languages such as English and other Indian Languages using computers.	1,2,3,7 and 8
2	Describe the concepts of morphology, syntax, semantics, discourse, and pragmatics of natural language.	1,2,3,7 and 8
3	Perform POS tagging for a given natural language and Select a suitable language modelling technique based on the structure of the language.	1,2,3,7 and 8
4	Demonstrate advanced algorithms and techniques for text-based processing with respect to morphology.	1,2,3,7 and 8
5	Develop Statistical Methods for Real World Applications and explore deep learning based NLP	1,2,3,7 and 8

SEMESTER – VI									
Course Title	**Deep Learning**								
Course code	**24BCAO3208R**	**Total credits: 3** **Total hours: 45T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Machine Learning**	**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives (Minimum 3)	1.Introduce major deep learning algorithms. 2.Differentiate between Traditional ML and Deep Learning 3.The problem settings, and their applications to solve real-world problems								
CO1	Understand the basics of artificial neural networks and deep learning.								
CO2	Mathematical intuition of Deep learning networks, and their application.								
CO3	Different Deep Learning Tools in academic research.								
CO4	Identify the deep learning algorithms and tools for real-world applications which are more appropriate for various types of learning tasks in various domains.								
CO5	Analyze deep learning algorithms to solve real-world problems.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction:** Various paradigms of learning problems, Perspectives, and Issues in deep learning framework, review of fundamental learning techniques. **Feed-forward neural network:** Artificial Neural Network, activation function, multi-layer neural network.	**10**	Explain and Understand the fundamentals of Deep learning techniques. Applying with example.	**1, 2, 4**
II	**Training Neural Network:** Risk minimization, loss function, backpropagation, regularization, model selection, and optimization. **Conditional Random Fields:** Linear chain, partition function, Markov network, Belief propagation, Training CRFs, Hidden Markov Model, Entropy.	**10**	Explain Neural Networks and understand the architecture and their applications.	**1, 2, 3**
III	**Deep Learning:** Deep Feed Forward network, regularizations, training deep models, dropouts, Convolutional Neural Network, Recurrent Neural Network, Deep Belief Network.	**8**	Understanding and Analysing the Feedforward and Dropout in CNN	**2, 4**
IV	**Probabilistic Neural Network:** Hopfield Net, Boltzman machine, RBMs, Sigmoid net, Autoencoders.	**10**	Understanding and Application of Probabilistic Neural Network with example.	**2, 3**
V	**Deep Learning research:** Object recognition, sparse coding, computer vision, natural language processing. **Deep Learning Tools:** Caffe, Theano, Torch.	**8**	Understanding the Deep Learning in Computer Vision, NLP etc. Write the code in different Application Tools.	**2, 3**
Practical				
I	Data Manipulation using NumPy and Pandas	**2**	Understand basic operations in NumPy and Pandas for handling datasets.	**2**

I	Implementing Activation Functions using NumPy	**2**	Apply mathematical functions like Sigmoid, ReLU, and Tanh using NumPy.	**3**
I	Creating a Simple Feed-Forward Neural Network	**3**	Build and test a multi-layer perceptron using NumPy.	**3**
II	Implementing Backpropagation from Scratch using NumPy	**3**	Develop and analyze the backpropagation algorithm in a simple neural network.	**4**
II	Loss Function Implementation in Neural Networks	**2**	Compute and compare loss functions like MSE, Cross-Entropy using NumPy.	**3**
II	Regularization Techniques (L1, L2) in Deep Learning Models	**2**	Apply L1/L2 regularization and observe its effect on models.	**4**
II	Hidden Markov Model (HMM) using Pandas & NumPy	**3**	Implement and analyze HMMs for sequence prediction tasks.	**4**
III	Implementing Dropout Regularization in Neural Networks	**3**	Apply dropout and analyze its effect on overfitting using NumPy.	**4**
III	Creating a Simple CNN using NumPy	**3**	Simulate the convolution operation manually to understand CNNs.	**3**
III	Implementing a Recurrent Neural Network from Scratch	**4**	Build a simple RNN using NumPy and analyze sequence data.	**4**
IV	Implementing Hopfield Network using NumPy	**3**	Simulate associative memory using a Hopfield network.	**4**
IV	Building an Autoencoder using NumPy	**3**	Understand dimensionality reduction by implementing autoencoders.	**4**
V	Object Recognition using NumPy Image Processing	**4**	Apply basic object detection using NumPy and OpenCV.	**5**
V	Implementing Sparse Coding for Feature Extraction	**3**	Develop and test sparse coding techniques using NumPy.	**5**
V	NLP-Based Sentiment Analysis using Pandas	**3**	Process text data and perform sentiment analysis using Pandas.	**5**

Text Books:

1. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016..
2. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006..

Reference Books:

1. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.

2. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the basics of artificial neural networks and deep learning.	1,2,3, and 8
2	Mathematical intuition of Deep learning networks, and their application.	1,2,3, and 8
3	Different Deep Learning Tools in academic research.	1,3, and 8
4	Identify the deep learning algorithms and tools for real-world applications which are more appropriate for various types of learning tasks in various domains.	1,2,3,7, and 8
5	Analyze deep learning algorithms to solve real-world problems.	1,2,3,7, and 8

SEMESTER – VI									
Course Title	**PE IV – IT Governance, Risk & Information Security Management**								
Course code	**24BCAO3204R**	**Total credits: 3** **Total hours: 30T+15P**	**L**	**T**	**P**	**S**	**R**	**O/ F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. To help students understand the need for governance in information security 2. To make it possible for students to learn how important concepts of IT governance is integrated with information security 3. To facilitate students to learn key frameworks used in information security and risk management and need for a metrics-based approach for effective information security								
CO1	Analyze and evaluate IT governance principles and their application in aligning IT strategies with business goals, as well as ensuring the security of information systems across an organization.								
CO2	Apply industry-standard frameworks like COBIT, Val-IT, and Risk-IT for developing and managing effective IT governance and risk management strategies that support information security and organizational objectives.								
CO3	Design and develop security strategies that integrate business objectives with IT and security initiatives, ensuring a robust security posture within the organization.								
CO4	Implement risk management programs using structured processes and frameworks to identify, assess, and mitigate risks, ensuring strategic decision-making for IT and information security management.								
CO5	Evaluate and improve information security management programs, leveraging continuous assessment and frameworks such as COBIT to optimize security performance and ensure compliance across organizational levels.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**IT Governance-Part 1** Introduction & Concepts, Origin of Governance, Corporate Governance, Best Practices for IT Governance, Role of Governance in Information Security, Six outcomes of effective Security Governance, benefits of good governance, Cultural aspects in governance.	**8**	Define IT governance concepts, evaluate corporate governance practices, analyze the role of governance in information security, and examine the cultural aspects and benefits of good governance in ensuring effective security outcomes.	1,3, 6
II	**IT Governance-Part 2** IT Governance-Roles and Responsibilities, Role of IT Strategy Committee and Security Steering Committee, Standard IT Balanced Scorecard. Val-IT framework of ISACA, Governance in multi-department and multi-country enterprises, Importance of Governance in establishing a sustainable Security Culture in the organization	**10**	Demonstrate roles and responsibilities in IT governance, apply governance frameworks like Val-IT and the IT Balanced Scorecard, and analyze the impact of governance in establishing a sustainable security culture across multi-departmental and multi-country enterprises.	3,4
III	**Information Systems Strategy** Role of Strategic Planning for IT, Strategic Direction and Alignment of Security Strategy with Business Objectives, Role of CISO, Security Metrics Program.	**8**	Design IT strategies aligned with business objectives, examine the role of CISO in strategic planning, and develop security metrics programs to measure the effectiveness of information security strategies.	3,4, 5

IV	**Risk Management Program** Develop a Risk Management Program. Risk Management Process, Roles and Responsibilities, Risk-IT Framework of ISACA, Strategic Security decisioning using Risk Management	**6**	Develop and implement a risk management program, apply the Risk-IT framework of ISACA, and make strategic security decisions based on comprehensive risk analysis and management processes.	3,5
V	**Information Security Management** Introduction, Performance Optimization, Management Information Security Forum, Segregation of Duties, Description of COBIT and other Frameworks, Security Program Effectiveness, Continuous Assessment and Improvement, In-sourcing versus Out-sourcing, Impact of ISM program across organization	**5**	Manage information security programs, evaluate the effectiveness of security measures, implement continuous improvement processes, and assess the impact of ISM programs across organizations.	3,5, 6
Practical				
Practical 1	Introduction to Incident Response Lifecycle	**2**	Understand and apply the phases of the Incident Response Lifecycle.	**3**
Practical 2	Log Analysis and Preprocessing of Data	**2**	Analyze and preprocess logs to extract actionable insights for security.	**4**
Practical 3	Digital Forensics Techniques	**2**	Apply digital forensics techniques to investigate, analyze, and preserve evidence.	**3**
Practical 4	Network Traffic Analysis	**2**	Analyze network traffic to detect anomalies, threats, and ensure security.	**3**
Practical 5	Malware Analysis Basics	**2**	Identify, analyze, and mitigate malware threats using basic analysis techniques.	**4**
Practical 6	Automated Incident Response Tools	**2**	Use automated tools to streamline incident response and improve efficiency.	**3**
Practical 7	Building a Simple Intrusion Detection Model	**2**	Develop and deploy a basic intrusion detection model for security.	**4**
Practical 8	Real-Time Intrusion Detection	**2**	Implement real-time intrusion detection to identify and mitigate security threats.	**5**
Practical 9	Automating Intrusion Response	**2**	Automate intrusion response processes to quickly detect and mitigate threats.	**4, 5**
Practical 10	Incident Containment Strategies	**2**	Implement effective containment strategies to minimize damage and prevent escalation.	**4**
Practical 11	Developing and evaluating an IR Plan	**2**	Design, implement, and evaluate incident response plans for effective management.	**4**
Practical 12	Introduction to Cloud Incident Response Scenarios	**2**	Understand and respond to cloud-based incidents using appropriate	**5**

			response strategies.	

Text Books:

T1. Information Security Governance by S.H. Solms, RossouwSolms, Springer; 1st Edition. 2nd Printing, 2008 edition (12 December 2008)

T2. IT Governance: How Top Performers Manage IT Decision Rights for Superior Results by Weill, Harvard Business Review Press; First edition (1 June 2004)

T3.ISACA publications

Reference Books:

R1. IT Governance: An International Guide to Data Security and ISO27001/ISO27002 by Alan Calder, Steve Watkins, Kogan Page; 6 edition (3 September 2015)

R2. ISACA publications on COBIT, RiskIT and ValIT

R3. Information Security Governance: Guidance for Information Security Managers by W. KragBrotby and IT Governance Institute, Isaca (2 June 2008)

R4. COBIT 5 Framework Perfect by Isaca, (10 April 2012)

R5. Cobit 5 Foundation-reference and Study Guide by Ana Cecilia Delgado, Create Space Independent Publishing Platform; Stg edition (20 June 2016)

R6. Governance of Enterprise IT Based on COBIT 5: A Management Guide by Geoff Harmer (Author), IT Governance Publishing, (6 February 2014)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate an understanding of fundamental concepts of cybersecurity, threat landscapes, and the importance of incident response management in safeguarding organizational assets.	2,3,4,5,6,7,8
2	Apply techniques for identifying, detecting, and analyzing security incidents, including recognizing patterns of malicious activity and differentiating between false positives and genuine threats.	1,2,3,4,5,6,7,8
3	Develop and evaluate incident response plans, policies, and procedures that align with organizational goals and regulatory compliance standards.	1,2,3,4,5,6,7,8
4	Utilize appropriate tools and frameworks to investigate, mitigate, and remediate cybersecurity incidents effectively, ensuring minimal disruption to business operations.	1,2,3,4,5,6,7,8
5	Analyze post-incident findings to generate reports, improve security policies, and implement measures to prevent future incidents, fostering a proactive cybersecurity posture.	2,3,4,5,6,7,8

SEMESTER – VI										
Course Title	**PE II – Cyber Forensics**									
Course code	**24BCAO3206R**	**Total credits: 3** **Total hours: 30T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Computer Applications**								
Semester	**Fall/ II semester of the second year of the program**								
Course Objectives	1. Introduce students to the fundamental concepts and procedures in computer forensics and cybercrime investigations. 2. Provide an in-depth understanding of storage devices, data acquisition, and recovery techniques. 3. Familiarize students with forensic techniques for Windows, Linux, mobile devices, networks, and email tracking.									
CO1	Describe the principles of computer forensics and implement cybercrime investigation procedures.									
CO2	Analyze storage devices and apply techniques for data recovery and volatile data analysis.									
CO3	Utilize forensic tools and techniques for investigating various platforms, including networks, mobile devices, and email systems.									
CO4	Apply cyber law principles and evidence-handling procedures to ensure legal compliance during investigations.									
CO5	Conduct forensic analysis of web applications, including server logs and security breaches, to identify intruders and vulnerabilities.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Computer Forensics** Introduction to Computer Forensics, Forms of Cyber Crime, First Responder Procedure- Non-technical staff, Technical Staff, Forensics Expert and Computer Investigation procedure, Case Studies.	**8**		
II	**Storage Devices & Data Recover Methods** Storage Devices- Magnetic Medium, Non-magnetic medium and Optical Medium, Working of Storage devices-Platter, Head assembly, spindle motor, Data Acquisition, Data deletion and data recovery method and techniques, volatile data analysis, Case Studies.	**10**		
III	**Forensics Techniques** Windows forensic, Linux Forensics, Network forensics – sources of network-based evidence, other basic technical fundamentals, Mobile Forensics – data extraction & analysis, Steganography, Password	**8**		

	cracking-Brute force, Cross-drive analysis, Live analysis, deleted files, stochastic forensics, Dictionary attack, Rainbow attack, Email Tacking – Header option of SMTP, POP3, IMAP, examining browsers, Case Studies.			
IV	**Cyber Law** Corporate espionage, digital evidences handling procedure, Chain of custody, Main features of Indian IT Act 2008 (Amendment), Case Studies, Incident specific procedures – virus and worm incidents, Hacker incidents, Social incidents, physical incident, Guidelines for writing forensic report.	**6**		
V	**Forensic Analysis of Web Application** Forensic analysis of web server, network analysis of web server compromise, web server log analysis, web application forensic, forensic analysis of web application security, intruder profiling, forensic for code injection attack, Case Studies	**5**		
Practical Component – NIL				

SN	Course Outcome (CO)	Mapped Program Outcome
1	Describe the principles of computer forensics and implement cybercrime investigation procedures.	2,3,5,6,7, and 8
2	Analyze storage devices and apply techniques for data recovery and volatile data analysis.	1,2,3,5,6,7, and 8
3	Utilize forensic tools and techniques for investigating various platforms, including networks, mobile devices, and email systems.	1,2,3,5,6,7, and 8
4	Apply cyber law principles and evidence-handling procedures to ensure legal compliance during investigations.	2,3,4,5,6,7, and 8
5	Conduct forensic analysis of web applications, including server logs and security breaches, to identify intruders and vulnerabilities.	1,2,3,5,6,7, and 8

SEMESTER – VI									
Course Title	**Project II**								
Course code	**24BCAO3207R**	**Total credits: 6**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**8**	**0**	**12**	**0**	**6**
Pre-requisite		**Co-requisite**	NIL						
Programme	**Bachelor of Computer Application (BCA)**								
Semester	**VI**								
Course Objectives	1. To develop the practical skill 2. To enhance Critical Thinking and Problem-Solving Abilities 3. To Improve Communication and Collaboration Skills								
CO1	Demonstrate a sound technical knowledge of the selected project topic.								
CO2	Illustrate problem identification, formulation, and solution.								
CO3	Plan solutions to complex problems utilizing a systems approach.								
CO4	Design the prototype of the solution and analyze the prototype using testing methods.								
CO5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Project Introduction and Planning** Overview of project goals and objectives. Importance of project planning and management. Project Proposal Development: Identifying a project topic or problem Conducting preliminary research Writing a project proposal Setting timelines and milestones Resource allocation (materials, tools, etc.) Risk assessment and management strategies	**24**	Describe, illustrate and explain problem	**2,3**
II	Literature Review and Research Methodology Identifying and reviewing relevant literature Summarizing and synthesizing existing research Highlighting gaps and formulating research questions Choosing appropriate research methods (qualitative, quantitative, or mixed methods) Designing research instruments (surveys, interviews, experiments, etc.) Ethical considerations in research	**24**	Describe, illustrate and Literature Review	**2,3,4**
III	Data Collection and Analysis Gathering primary data (surveys, experiments, interviews, observations) Ensuring data accuracy and reliability Storing and organizing data Using statistical tools and software for data analysis Interpreting data and drawing conclusions Visualizing data (charts, graphs, tables)	**24**	Describe, illustrate and explain data collection and analysis	**2,3,4**
IV	Project Implementation and Monitoring Project Execution: Implementing the project plan Monitoring progress and making adjustments as needed Documentation and Record Keeping: Keeping detailed records of project activities and	**24**	Describe, illustrate and explain project implementation and execution	**6**

	outcomes Troubleshooting and problem-solving during implementation Interim Reporting: Preparing and presenting progress reports Receiving and incorporating feedback			
V	Project Presentation and Evaluation Final Reporting: Writing the final project report (structure, content, formatting) Ensuring clarity, coherence, and completeness Project Presentation: Preparing visual aids (slides, posters, etc.) Practicing presentation skills (public speaking, answering questions) Evaluation and Reflection: Assessing project outcomes against objectives Reflecting on the project process and personal learning Identifying areas for future improvement and potential follow-up projects	**24**	Describe, illustrate and explain testing and documentation	**2,3,5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate a sound technical knowledge of selected project topic.	1, 2 and 4
2	Illustrate problem identification, formulation and solution.	1, and 2
3	Plan solutions to complex problems utilizing a systems approach.	1, 2, 3, 5 and 7
4	Design the prototype of the solution and analyze the prototype using testing methods.	1, 2, 3, 5, 6, 7, and 8
5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.	1, 2, 3, 6 and 8

Assam down town University

Curriculum and Syllabus

Bachelor of Technology
in
Computer Science and Engineering

OUTCOME BASED EDUCATION FRAMEWORK
CHOICE BASED CREDIT SYSTEM
Version: 2.2

FACULTY OF COMPUTER TECHNOLOGY

July, 2024

PREAMBLE

Assam down town University is a premier higher educational institution which offers Bachelor, Master, and Ph.D. degree programmes across various faculties. These programmes, collectively embodies the vision and mission of the university. In keeping with the vision of evolutionary changes taking place in the educational landscape of the country, the university has restructured the course curriculum as per the guidelines of National Education Policy 2020. This document contains outline of teaching and learning framework and complete detailing of the courses. This document is a guidebook for the students to choose desired courses for completing the programme and to be eligible for the degree. This volume also includes the prescribed literature, study materials, texts, and reference books under different courses as guidance for the students to follow.

Recommended by the Board of Studies (BOS) meeting of the Faculty of Computer Technology held on dated 15/07/2024 and approved by the 51st Academic Council (AC) meeting held on dated 26/07/2024

Chairperson, Board of Studies

Member Secretary, Academic Council

Vision

To become a Globally Recognized University from North Eastern Region of India, dedicated to the Holistic Development of Students and Making Society Better

Missions

1. Creation of curricula that address the local, regional, national, and international needs of graduates, providing them with diverse and well-rounded education.

2. Build a diverse student body from various socio-economic backgrounds, provide exceptional value-based education, and foster holistic personal development, strong academic careers, and confidence.

3. Achieve high placement success by offering students skill-based, innovative education and strong industry connections.

4. Become the premier destination of young people, desirous of becoming future professional leaders through multidisciplinary learning and serving society better.

5. Create a highly inspiring intellectual environment for exceptional learners, empowering them to aspire to join internationally acclaimed institutions and contribute to global efforts in addressing critical issues, such as sustainable development, Climate mitigation and fostering a conflict-free global society.

6. To be renowned for creating new knowledge through high quality interdisciplinary research for betterment of society.

7. Become a key hub for the growth and excellence of AdtU's stakeholders including educators, researchers and innovators

8. Adapt to the evolving needs and changing realities of our students and community by incorporating national and global perspectives, while ensuring our actions are in harmony with our foundational values and objectives of serving the community.

Programme Details

Programme Overview (not more than 100 words)

Bachelor of Technology in Computer Science and Engineering is an Under Graduate Degree awarded for the programme of Computer Science and Engineering. As one of the best computer engineering colleges, we intend to create a cohesive learning experience with the latest technological developments to that of industry demand.

I. Specific Features of the Curriculum

The B. Tech CSE curriculum, guided by the outlined Program Specific Outcomes (PSOs), emphasizes a comprehensive educational framework. It prepares students for competitive exams through updated and focused syllabi. It integrates advanced research principles with sustainable technology applications, ensuring graduates possess the necessary engineering and management skills for societal impact. The curriculum fosters effective communication, teamwork, and project management in multidisciplinary settings. Additionally, it incorporates contemporary industrial and research trends, encouraging innovation and entrepreneurial proficiency. This holistic approach equips students to develop novel software engineering solutions, addressing real-world problems with cutting-edge techniques and strategies.

II. Eligibility Criteria:

Minimum 45% with English, Physics & Mathematics in HS.

III. Program Educational Objectives (PEOs):

PEO1	To produce graduates who have strong foundation of knowledge and skills in the field of Computer Science and Engineering.
PEO2	To produce graduates who can provide solutions to challenging problems in their profession by applying Computer Engineering theory and practices.
PEO3	To produce graduates who are employable in industries/public sector/research organizations or work as an entrepreneur, as well as can provide leadership and are effective in multidisciplinary environment.

IV. Program Specific Outcomes (PSOs):

PSO 1: Educational Advancement: Graduates will be prepared for state, national and international competitive examinations with focused and updated syllabi.

PSO 2: Advanced Research: Graduates will have a holistic comprehension of engineering and management principles required for application of sustainable

technologies for societal development, and also will be able to communicate effectively in oral, written, visual and graphic modes as a member and leader in a team, to manage projects in multidisciplinary environments.

PSO 3: Entrepreneurial Proficiency: Graduates will be acquainted with the contemporary trends in industrial/research environments and also will capable of innovating novel solutions to prevailing problems by applying software engineering techniques and strategies.

V. Program Outcome: (8-12)

PO1 Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2 Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4 Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective

presentations, and give and receive clear instructions.

PO11 Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

VI. Total Credits to be Earned:

The total credit to be earned is 173

VII. Career Prospects:

A B. Tech in Computer Science and Engineering (CSE) offers diverse and promising career prospects. Graduates can pursue roles such as software developers, data scientists, cybersecurity analysts, and AI/ML engineers in top tech firms. They are well-suited for positions in research and development, working on cutting-edge technologies. The curriculum also prepares them for competitive exams, enabling careers in public sector and government organizations. Additionally, the strong emphasis on entrepreneurial skills allows graduates to innovate and launch their own start-ups. With rapid advancements in technology, CSE graduates remain in high demand across various industries, ensuring robust career opportunities.

EVALUATION METHODS

The student performance shall be evaluated through In-semester (Sessional) and semester-end examinations. A weightage of 40% or as prescribed by the programme shall be added to the score of the end-semester examination.

A. INTERNAL ASSESSMENT:

The teacher who offers the course shall be responsible for internal assessment by conducting in-semester (sessional) examination and evaluating the performance of the students pursuing that course. The components for internal assessment are illustrated in the table given below.

Table 1: Question paper pattern for End semester examination

SN	Components/ Examinations	Marks Allotted
1.	In-Sem Exam – I (ISE-I) (Written Examination)*	30
2.	In-Sem Exam – II (ISE-II) (Written Examination)*	30
3.	Assignment	10
4.	Presentation (SP)	10
5.	Quiz	5
6.	Class Performance based score*	5

are compulsory

Note: *Total Internal assessment should be out of 40*

INSTRUCTION

1. If a student fails to appear in the any of the component without any valid reason he/she shall be marked zero in that component. However, the course teacher at his discretion may arrange for the missed test on an alternate date for the absentee students after determining ground with genuine/valid reasons for the absent.

2. The report of evaluation of an activity towards the in-semester (sessional) component of a course shall be duly notified by the concerned course teacher within a week of completion.

3. The program coordinators should upload the in-semester marks to the ERP and forward acknowledgement of all the courses of the program to the Controller of Examinations before the start of the End-semester examination.

B. SEMESTER END EXAMINATION:

Time table for end semester examination is published at least 25 days prior to the start of

Examination.

I. Pre-Examination:

Eligibility Criteria for a student to appear in University Examinations:

The student shall only be allowed to appear in a University Examination, if:

i) He/ She is a registered student of the University;

ii) He/ She is of good conduct and character;

iii) He/ She has completed the prescribed Programme of study with minimum percentage of attendance as laid down in the Regulations of the Programme concerned.

Under special cases, a student may be allowed to appear for an examination without being registered in the University but the result of the said student will be kept on hold till the registration of the concerned student is completed.

II. Admit Card:

Admit card for the examination may be downloaded through ERP where the system will generate a Unique ID Cards through online.

The University shall have the right to cancel admission for examination of any candidate on valid grounds.

III. Pattern of Question Papers:

The question paper shall follow the principles of Bloom's Taxonomy.

Table 2: Question paper pattern for End semester examination

S. N.	Level	Questions /verbs for test
1	Remember	List, Define, tell, describe, recite, recall, identify, show who, when, where, etc.
2	Understand	Describe, explain, contrast, summarize, differentiate, discuss, etc.
3	Apply	Predict, apply, solve, illustrate, determine, examine, modify
4	Analyze	Classify, outline, categorize, analyze, diagrams, illustrate, infer, etc.
5	Evaluate	Assess, summarize, choose, evaluate, recommend, justify, compare etc.
6	Create	Design, Formulate, Modify, Develop, integrate, etc.

Note: No course is to be evaluated on basis of **all 6 knowledge levels.**

The format of the question paper across all the program follow a unique pattern and the total marks is 60

Table 3: Question paper pattern for End semester examination

Sl no	Question pattern	Total marks
1	MCQs (10 Questions)	10
2	2 Marks questions (10 Questions)	20
3	4 Marks questions (5 Questions)	20
4	10 Marks questions (1 Question)	10

IV. Examination Duration:

Each paper of 60 marks shall ordinarily be of two hours duration.

V. Practical Examinations, Viva-Voce etc.:

i) Practical examination shall be conducted in the presence of one external expert and one or more internal examiners.

ii) Viva-Voce, Oral examinations of the Project report, Dissertation etc. shall be undertaken by a Board of Examiners constituted by the respective Dean of Program with the advice of Supervisor(s).

VI. Procedure of Expulsion:

If any candidate is found to be using any unfair-means during the examination, the invigilator may cease his/her answer sheet and report it directly to the Officer-in-Charge. The Office-in-Charge of the center may take appropriate decisions as per the rules and procedure of the examination. The Officer-in-Charge may allow the students to write the exam with new answer sheet or may expel the student from appearing the paper depending on the nature of unfair-means. In case of Computer based test, the students may be directed to write an apology letter and sign in the prescribe expulsion form. The student may not be allowed to write that examination.

VII. Instruction to the Students:

(i) The students shall not bring to the Examination Hall, any electronic gadget used as a means of communication or record except electronic calculator, if required.

(ii) The students shall not receive any book or printed or hand written or photo copy (Xerox) or blank-paper from any other person while he/she is in the examination-room or in laboratory or in any other place to which he/she is allowed to have access during course of examination.

(iii) The students shall not communicate with any other candidate in the examination room or with any other person in and outside the examination-room.

(iv) The students shall not see, read or copy anything written by any other candidate, nor shall he/she knowingly or negligently permit any other candidate to see, read or copy anything written by him/her or conveyed by him/her.

(v) The students shall not write anything on the Question Paper or in other paper or materials during the examination, or pass any kind of paper to any other candidate in the examination-room, or to any person outside the room.

(vi) The students shall not disclose his/her identity to the examiner by writing his/her name or putting any sign / symbol in any part of his answer-script.

(vii) The students shall not use any abusive language or write any objectionable remark or make any appeal to examiner by writing in any part of his answer-script.

(viii) The students shall not detach any page from the answer-script or insert any authorized or unauthorized loose sheet into it. He /she shall also not insert any other answer-script / loose sheet by removing the pins of the origin answer-scripts and re-fixing it.

(ix) The students shall not resort to any disorderly conduct inside the examination-room or misbehave with the invigilator or any other examination official.

VIII. Provision for an Amanuensis (writer):

(i) A candidate may be provided with an Amanuensis (writer) to write down on dictation on his / her behalf on ground of his / her physical disability to write down by himself / herself due to accident or any other reason. The amanuensis may be provided till he / she recovers from the physical disability. The physical disability to write down by himself / herself must be supported by Medical Certificate from a competent Medical Officer.

(ii) The qualifications of the amanuensis so provided must not be equal or higher than that of the candidate. This is also to be supported by Certificate from the Faculty of Study where the Amanuensis is provided.

(iii) Such candidates are to be accommodated in a separate room under the supervision of an invigilator so that the fellow candidates are not disturbed in the process.

C. Credit Point:

It is the product of grade point and number of credits for a course, thus, CP = GP x CR

i. Credit:

A unit by which the course work is measured. It determines the number of hours of instructions required per week. 'Credit' refers to the weightage given to a course, usually in terms of the number of instructional hours per week assigned to it. Credits assigned for

a single course always pay attention to how many hours it would take for an average learner to complete a single course successfully.

ii. Grade Point:

Grade Point is a numerical weight allotted to each Grade Letter on a 10-point scale.

iii. Letter Grade:

Letter Grade is an index of the performance of students in a said paper of a particular course. Grades are denoted by letters O, A+, A, B+, B, C, P, F and Abs. Student obtaining Grade F / Grade Abs shall be considered failed/ absent and, will be required to appear in the subsequent ESE. The UGC recommends a 10-point grading system with the following (Table: 4) Letter Grades:

(i) A Letter Grade shall signify the level of qualitative/quantitative academic achievement of a student in a Course, while the Grade Point shall indicate the numerical weight of the Letter Grade on a 10-point scale.

(ii) There shall be 08 (eight) Letter Grades bearing specific Grade Points as listed in Table 4, where the Letter Grades 'O' to 'P' shall indicate successful completion of a course.

(iii) Apart from the 08 (eight) regular Letter Grades listed in Table 4, there shall be 03 (three) additional Letter Grades, which shall be awarded if a Course is withdrawn or spanned over the next Semester or remains incomplete.

Table 4: Letter Grades and Grade Points

Letter Grade	Grade Points	Description
O	10	Outstanding
A+	9	Excellent
A	8	Very Good
B+	7	Good
B	6	Above Average
C	5	Average
P	4	Pass
F	0	Fail
Abs	0	Absent
UFM	0	Unfair Means

iv. Grade Point Average:

a. SGPA (Semester Grade Point Average)

The SGPA of a student in a Semester shall be the weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered in that Semester, irrespective of whether he/she could or could not complete the Courses. More specifically, the calculation of SGPA shall take into account the Courses graded with Letter Grades 'O' to 'F' as given in Table 4.

$$\text{SGPA} = \frac{\Sigma_{i=1}^{n} C_i G_i}{\Sigma_{i=1}^{n} C_i} \qquad (1.1)$$

The SGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.1) up to two decimal places, where n is the total number of Credit Courses registered by the student in that Semester, Gi is the Grade Point secured in the i^{th} registered Course and Ci is the Credit (weight) of that Course.

b. CGPA (Cumulative Grade Point Average)

(i) The CGPA of a student in a Semester of a Programme shall be the accumulated weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered and successfully completed so far starting from the enrolment in the Programme. In other words, taking into account all the Courses graded with 'O' to 'P' as given in Table 4, generally the CGPA of a student shall be calculated starting from the first Semester of his/her enrolled Programme, while the CGPA of a lateral-entry student shall be calculated starting from the Semester of his/her enrolment.

(ii) The CGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.2) up to two decimal places, where N is the total number of Credit Courses registered and successfully completed so far by the student, Gi is the Grade Point secured in the ith completed Course and Ci is the Credit (weight)of that Course.

$$\text{CGPA} = \frac{\Sigma_{i=1}^{N} C_i G_i}{\Sigma_{i=1}^{N} C_i} \qquad (1.2)$$

(iii) The CGPA shall be convertible into equivalent percentage of marks using Equation Conversion of CGPA to percentage marks: = CGPA*10

D. Post-Examination

i. Transcript or Grade Card or Certificate:

A marking certificate shall be issued to all the registered students after every Semester. The Semester mark sheet will display the course details (code, title, number of credits, grade secured) along with total credit earned in that Semester.

ii. Grievance Readdress Mechanism:

Students with any dissatisfaction or grievance regarding the marks awarded in any of the Papers / Courses may appeal to the Controller of Examinations for remedial action such as Re-evaluation within 10 days of the declaration of result.

(i) A student has options to appeal for re-evaluation of his /her answer script to the Controller of Examination.

(ii) Application for re-evaluation / re-scrutiny of answer scripts shall be made in the definite proforma available with the Examination Office through the head of the respective departments within 10 days of declaration of the results of the respective examinations.

(iii) The Controller of Examination may appoint an examiner for re-evaluation and will consider and recognize the evaluation done by a University appointed examiner.

(iv) There shall be no provision for re-evaluation of the Practical Papers, Project Work, and Dissertation etc. However, the students fail in practical examination or viva voce and wish to appear again may apply to be evaluated can do so with the next schedule.

(v) After screening the application for re-evaluation, the CoE may send the answer scripts of the student to the examiners appointed by the CoE with the approval of Vice Chancellor.

(vi) The marks/grades achieved by the students after the re-evaluation shall be final and binding.

(vii) Fresh Marks – sheets / Grade Card shall be issued only if the candidate secures pass marks / passing grade in the re-evaluated paper.

(viii) Revaluation of answer scripts shall be deemed to be an additional facility provided to the students with a view to improving upon their results at the preceding examination result for any reason whatsoever shall not confer any right upon them for admission to next higher class which matters always be regulated in accordance with the relevant rules or regulations framed by the University.

(ix) If as a result of revaluation of the candidate attracts the provision of condonation of deficiency, the same may be applied to his/her only for fresh attempt.

INSTRUCTION TO TEACHERS AND STUDENTS

(Teaching and Learning Methods)

In all the courses the teacher has to select topics for teacher-method which should not be less than 20 percent. The approach will be direct classroom teaching through a series of lectures delivering concepts using ITC facilities, white or blackboard. Notes may also be circulated to the students; however, the students are to be involved in the preparation of the notes. The teacher will be responsible for selecting the best note for circulation. The teacher-centric methodology has recently fallen out of favour because this strategy for teaching is seen to favour passive students.

1. **Student- centric / Constructivist Approach:**

The topics of the courses may be selected at the start of the class and assigned one topic to each of the students for studying by themselves, prepare presentations, notes, etc., and present at respective class time after consultation and discussion with the course teachers. The teacher facilitates the learning of the students by guiding and providing input and explaining concepts. 60 percent of the course contents may be selected for this purpose. To avoid behaviour problems, teachers must lay a lot of groundwork in student-centric classrooms. Typically, it involves instilling a sense of responsibility in students. In addition, students must learn internal motivation.

a. Project-Based Learning: The teacher may select 5 percent of topics for the purpose and may conduct visits to the laboratory for experiments or field surveys. The selection of the topic may be done considering the available facility for the purpose. However, in the final semester of each of the programme the student has to undergo project-based learning at least 4 months duration. This approach will help the student to think critically, evaluate, analyze, make decisions, collaborate, and more.

b. Inquiry-Based Learning: The teacher/ students are supposed to list at least five questions in each contact hour and student solve these question or search for answer which becomes the home work for the students "question-driven" learning approach. The teacher may look for the correctness of the solution or the best possible answer and discuss in the successive class. This will help in the preparation for various competitive examination and develop a habit for search for solutions.

c. Flipped Classroom: About 10 percent of the course content has to be completed by this method. In this approach the students are asked to watch video or lecture prepared by the teacher or any video available (relevant to the course). A set of questions may be given to the students for searching answers by the students. The idea is that students should have more time in-classroom focusing on achieving these higher levels of thinking and learning. The Flipped classroom is also an acronym. The letters FLIP represent the four pillars included in this type of learning: Flexible environment, Learning culture shift, Intentional content, and Professional educator. As you can see, the second pillar refers to a culture shift from the traditional approach where students are more passive to an approach where students are active participants. As a result, this approach is also a student- centric teaching method.

d. Cooperative Learning: The remaining five percent has to be completed by cooperative learning approach. In this approach, the students are allotted problems. During library hours the students along with the teacher visit the library and search for probable solutions for the assigned problem. The same has to be done in groups so that the students discuss among themselves for the appropriate answers. Essentially, cooperative learning believes that social interactions can improve learning. In addition, the approach recreates real-world work situations in which collaboration and cooperation are required.

The percentage categorization for the completion of a theory course

Teacher-centric or Direct Classroom Teaching: Delivery by series of lectures	20%
Student-centric Approach, Students present and deliver lectures in the presence of teacher and supervised by teacher	60%
Students visit fields or perform experiments or teachers perform demonstration	05%
Flipped Classroom approach	10%
Cooperative learning approach	05%

Inquiry-based approach has to be followed in all of the classes

The teacher has to distribute the topics to be considered for teaching by the above-mentioned approaches and prepare a lesson plan for execution and maintain a file.

B.Tech CSE Curriculum 2024-28 (AdtU)

First Semester

Sl.No	Course Title	Course Code	Course Category	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Linear Algebra and Univariate Calculus	24BTCS1101R	DSC (Minor)	4	3	1	0	0	0	0	40	60	0	100
2	Problem Solving Using C	24BTCS1102R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	Engineering Physics	24BTCS1103R	MDC	3	2	0	2	0	0	0	40	60	100	200
4	Basic Electrical and Electronics Engineering	24BTCS1104R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
5	Workshop for Engineers	24BTCS1105R	DSC (Minor)	1	0	0	2	0	0	0	0	0	100	100
6	Introduction to Java	24MOCS1101R 24MOCS1102R	VAC	2	0	0	0	8	0	0	100	0	0	100
7	Introductory English	24UBPD1104R	AEC	2	0	0	4	0	0	0	0	0	100	100
8	Co Curricular Activities	24UBCC1101	Co Curricular	1	0	0	0	4	0	0	100	0	0	100
Total Credit in 1 st Semester				**20**	**10**	**1**	**12**	**12**	**0**	**0**	**360**	**240**	**500**	**1100**

2nd Semester

Sl.No	Course Title	Course Code	Course Category	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Ordinary Differential Equations and Multivariate Calculus	24BTCS1201R	DSC (Minor)	4	3	1	0	0	0	0	40	60	0	100
2	Object Oriented Programming using JAVA	24BTCS1202R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	Data Structure and Algorithms	24BTCS1203R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Environmental Science	24UBES1201R	MDC	3	3	0	0	0	0	0	40	60	0	100
5	1. Introduction To Git And Github 2. React Basics	24MOCS1201R 24MOCS1202R	VAC	2	2	0	0	0	0	0	100	0	0	100
6	Field-based Learning	24BTCS1204R	Field Visit	1	0	0	0	0	0	16	100	0	0	100
7	PDP	24UBPD1204R	AEC	2	0	0	4	0	0	0	0	0	100	100
8	Extra Curricular Activities	24UBEC1201	Extra Curricular Activities	1	0	0	0	4	0	0	100	0	0	100
Total Credit in 2nd Semester				**21**	**14**	**1**	**8**	**4**	**0**	**16**	**460**	**240**	**300**	**1000**

3rd Semester														
Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Digital Electronics	24BTCS2101R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
2	Functional Programming in Python	24BTCS2102R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	Discrete Mathematics & Graph Theory	24BTCS2103R	DSC (Minor)	3	3	0	0	0	0	0	40	60	0	100
4	Computer Organization and Architecture	24BTCS2104R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	Field-based Learning	24BTCS2105R	Field Visit	1	0	0	0	0	0	16	0	100	0	100
6	Financial Literacy (FL)	24UBPD2104R	MDC	1	0	0	2	0	0	0	0	0	100	100
7	Basic Life Saving Skills	24UULS2102R	MDC	1	0	0	2	0	0	0	0	0	100	100
8	1. Spring Framework Specialization 2. Javascript Basics	24MOCS2101R 24MOCS2102R	MDC	1	1	0	0	0	0	0	0	100	0	100
9	PDP	24UBPD2103R	AEC	2	0	0	4	0	0	0	0	0	100	100
10	Logical Reasoning for Computer Science	24BTCS2107R	SEC	2	0	0	4	0	0	0	0	0	100	100
11	Mini Project I	24BTCS2106R	DSC (Minor)	1	0	0	0	4	0	0	0	0	100	100
Total Credit in 3rd Semester				**23**	**12**	**0**	**18**	**4**	**0**	**16**	**160**	**440**	**800**	**1400**

4th Semester														
Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Database Management System	24BTCS2201R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Operating Systems	24BTCS2202R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	Probability and Statistics	24BTCS2203R	DSC (Minor)	3	2	1	0	0	0	0	40	60	0	100
4	Formal Language and	24BTCS2204R	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100

	Automata Theory													
5	PE - I	24BTCS2205R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
6	PDP	24UBPD2201R	AEC	2	0	0	4	0	0	0	0	0	100	100
7	Quantitative Aptitude For Computer Science	24BTCS2209R	SEC	2	1	0	2	0	0	0	0	0	100	100
8	Digital Literacy	24UCDL1002R	VAC	1	0	0	2	0	0	0	0	0	100	100
9	Basic Acclimitizing Skills	24UULS2201R	VAC	1	0	0	2	0	0	0	0	0	100	100
10	Mini Project II	24BTCS2208R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100
Total Credit in 4th Semester				**25**	**15**	**1**	**16**	**4**	**0**	**0**	**300**	**300**	**700**	**1300**

5th Semester

Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Data Communication and Computer Network	24BTCS3101R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Signals and Systems	24BTCS3102R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
3	Cloud Computing	24BTCS3103R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Software Engineering	24BTCS3104R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	PE - II	24BTCS3105R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
6	PDP	24UBPD3104R	AEC	2	0	0	4	0	0	0	0	0	100	100
7	Design Thinking and Entrepreneurship	24BTCS3106R	SEC	2	1	0	2	0	0	16	0	0	100	100
8	MOOCS	24MOCS3101R 24MOCS3102R	VAC	1	1	0	0	0	0	0	0	0	100	100
9	Essence of Indian Traditional Knowledge	24BTCS3107R	VAC	2	2	0	0	0	0	0	40	60	0	100
10	Mini Project-III	24BTCS3108R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100

Total Credit in 5th Semester				**27**	**18**	**0**	**16**	**4**	**0**	**16**	**340**	**360**	**800**	**1500**
6th Semester														
Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Web Technologies	24BTCS3201R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Compiler Design	24BTCS3202R	DSC (Major)	4	3	1	0	0	0	0	40	60	0	100
3	Design and Analysis of Algorithm	24BTCS3203R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	PE-III	24BTCS3203R 24BTCS3204R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	Statistical Methods and Modelling	24BTCS3205R	DSC (Minor)	3	3	0	2	0	0	0	40	60	100	200
6	Mini Project-IV	24BTCS3206R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100
7	PDP	24UBPD3205R	AEC	2	0	0	4	0	0	0	0	0	100	100
Total Credit in 6th Semester				**22**	**15**	**1**	**12**	**4**	**0**	**0**	**300**	**300**	**500**	**1100**
7th Semester														
Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	PE-III	24BTCS4101R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Project-I	24BTCS4102R	DSC (Major)	6	0	0	0	16	12	0	100	200	0	300
3	PE-IV	24BTCS4103R	DSC (Major)	4	3	0	2	4	0	0	40	60	100	200
4	Summer Internship	24BTCS4104R	Internship	4	0	0	0	16	0	0	0	100	0	100
Total Credit in 7th Semester				**18**	**6**	**0**	**4**	**36**	**12**	**0**	**180**	**420**	**200**	**800**
8th Semester														
Sl.No	**Course Title**	**Course Code**	**Course Category**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Cryptography & Network Security	24BTCS4201R	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100

2	Machine Learning	24BTCS4205R	DSC (Major)	4	3	0	2	4	0	0	40	60	100	200
3	Image Processing and Pattern Recognition	24BTCS4202R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Project-II	24BTCS4203R	DSC (Major)	6	0	0	0	12	18	0	100	200	0	300
Total Credit in 8th Semester				**17**	**9**	**0**	**4**	**16**	**18**	**0**	**220**	**380**	**200**	**800**

***IA: Internal Assessment,**
***SEE: Semester End Examination,**
***PE: Practical Examination**

List of Program Electives (PE)Paper		
Semester	**No.**	**Course Name**
IV	**PE-I**	**1. Artificial Intelligence** **2. Introduction to Internet of Things**
V	**PE-II**	**1. Data Mining and Analytics** **2. Information Retrieval Systems**
VI	**PE-III**	**1. Machine Learning** **2. Cloud Computing**
VII	**PE-IV**	**1. Fundamentals of Cyber Security** **2. Foundation of Data Science**

Semester	Credit	L	T	P	S	R	O	Maximum Marks			Total
								IA	SEE	PE	
1st Semester	20	10	1	12	12	0	0	360	240	500	1100
2nd Semester	21	14	1	8	4	0	16	460	240	300	1000
3rd Semester	23	12	0	18	4	0	16	160	440	800	1400
4th Semester	25	15	1	16	4	0	0	300	300	700	1300
5th Semester	27	18	0	16	4	0	16	340	360	800	1500
6th Semester	22	15	1	12	4	0	0	300	300	500	1100
7th Semester	18	6	0	4	36	12	0	180	420	200	800
8th Semester	17	9	0	4	16	18	0	220	380	200	800
Total	**173**	**99**	**4**	**90**	**84**	**30**	**48**	**2320**	**2680**	**4000**	**9000**

List of MOOCS				
Sl. No	**MOOCS Course No**	**Name of the Course**	**Course Code**	**Semester**
1	MOOCS I	Introduction to Java	24MOCS1101R	I
2		International Leadership And Organizational Behaviour	24MOCS1102R	
1	MOOCS II	Introduction To GIT And GITHUB	24MOCS1201R	II
2		React Basics	24MOCS1202R	
1	MOOCS III	Spring Framework Specialisation	24MOCS2101R	III
2		Javascript Basics	24MOCS2102R	
1	MOOCS IV	Introduction to Mobile Development	24MOCS3101R	V
2		Introduction to the Internet of Things and Embedded Systems	24MOCS2102R	

SEMESTER – I									
Course Title	**Linear Algebra and Univariate Calculus**								
Course code	**24BTCS1101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 48T	**3**	**1**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Mathematics	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. To make understand to evaluate definite and improper integrals. Apart from these some special functions like Beta and Gamma function must be introduced. 2. To provide the application of differential and integral calculus. 3. To make understand the convergence and divergence of sequence and series								
CO1	The students will be able to demonstrate the ability to understand the concepts of mathematics, logic, reasoning and coding								
CO2	Students will be able to apply of applications of concepts in other disciplines such as engineering, computer science, physics, etc.								
CO3	Solve and devise solutions to a range of elementary real-world problems in mathematics and programming								
CO4	Explore and apply key concepts in logical thinking to business problems.								
CO5	Enable students to critically analyze information to evaluate evidence and construct reasoned arguments.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Calculus:** Evaluates and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions.	10	Evaluate definite and improper integrals and understand their properties. Apply Beta and Gamma functions in various contexts. Use definite integrals to calculate surface areas and volumes of revolution.	1,2,3
II	**Calculus:** Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; indeterminate forms and L'Hospital's rule; Maxima and minima.	10	Understand and apply Rolle's Theorem and the Mean Value Theorem to determine properties of functions. Utilize Taylor and Maclaurin series expansions with remainders for function approximation. Evaluate indeterminate forms using L'Hospital's Rule, and identify and analyze local maxima and minima of functions.	3,4
III	**Sequences and series:** Convergence of sequence and series, tests for convergence; Power series, Taylor's series, series for exponential, trigonometric and logarithm functions; Fourier series: Half range sine and cosine series, Parseval's theorem.	10	Analyze the convergence of sequences and series using various tests. Develop and utilize power series, including Taylor's series, to represent functions such	3,4

			as exponential, trigonometric, and logarithmic functions. Apply Fourier series to decompose functions into their sine and cosine components and understand Parseval's theorem in the context of these series.	
IV	**Multivariable Calculus (Differentiation):** Limit, continuity and partial derivatives, directional derivatives, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers; Gradient, curl and divergence.	9	Understand and apply the concepts of limits, continuity, and partial derivatives, including directional and total derivatives. Analyze and solve problems involving tangent planes, normal lines, and optimization techniques such as maxima, minima, saddle points, and the method of Lagrange multipliers. Compute and interpret the gradient, curl, and divergence of vector fields.	4,5
V	**Matrices:** Inverse and rank of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Determinants; Eigenvalues and eigenvectors; Diagonalization of matrices; Cayley- Hamilton Theorem, and Orthogonal transformation.	9	Understand and calculate the inverse and rank of matrices, and apply the rank-nullity theorem. Solve systems of linear equations using matrix methods. Identify and work with symmetric and skew-symmetric matrices.	3,5

TEXT BOOKS:

T1: G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint,2002.

T2: Erwin kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley &Sons, 2006.

T3: Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi,11th Reprint, 2010

EFERENCE BOOKS:

R1:N.P. Bali and Manish Goyal, A text book of Engineering Mathematics,Laxmi Publications, Reprint,2008.

R2: B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition,2010.

OTHER LEARNING RESOURCES:
O1: Coursera online platform for Mathematical learning
O2: Math kind Quality *Math Education* portal

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	The students will be able to demonstrate the ability to understand the concepts of mathematics, logic, reasoning and coding.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
2	Students will be able to apply of applications of concepts in other disciplines such as engineering, computer science, physics, etc.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
3	Solve and devise solutions to a range of elementary real-world problems in mathematics and programming.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
4	Explore and apply key concepts in logical thinking to business problems.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
5	Enable students to critically analyze information in order to evaluate evidence and construct reasoned arguments.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12

A

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BTCS 1101R	**Linear Algebra and Univariate Calculus**	**CO1**	3	2	0	0	3				2	0	0	2
		CO2	3	3	0	0	2				0	0	0	2
		CO3	0	3	3	3	0				2	0	0	0
		CO4	2	0	3	0	0				0	3	2	1
		CO5	2	3	2	0	0				2	2	2	1
SEMESTER – I														

Course Title	Problem Solving Using C								
Course code	24BTCS1102R	Total credits:4 Total hours:40T+30P	L	T	P	S	R	O/F	C
			3	0	2	0	0	0	4
Pre-requisite	Nil	Co-requisite	Nil						
Programme	Bachelor of Technology in Computer Science and Engineering								
Semester	Summer/ I semester of the first year of the Programme								
Course Objectives (Minimum 3)	1. To formulate simple algorithms for arithmetic and logical problems. 2. To test and execute the programs and correct syntax and logical errors. 3. Apply C in Real-World Applications.								
CO1	Understand the concept of computer system, analyze a given problem, develop an algorithm, fundamental programming constructs, identify data representation formats, and describe operators and their precedence, associativity								
CO2	Apply branching and loop statements in problem solving.								
CO3	Apply the concepts of homogeneous derived data types, heterogeneous data types, strings and functions in programming.								
CO4	Understand the concept of pointers and apply it in programming.								
CO5	Apply the concept of file handling in C programming.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Introduction to Programming Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.) Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples. From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code.	8	Identify and describe the function of key components of a computer system, including disks, memory, processor, and how they interact with each other. Explain where and how a program is stored and executed within a computer system. Understand the role of an operating system in managing computer hardware and software resources.	1, 2, 3
II	Arithmetic expressions and precedence Conditional Branching and Loops: Writing and evaluation of conditionals and consequent branching, Iteration and loops Arrays: Arrays (1-D, 2-D), Character arrays and Strings Basic Algorithms: Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example	12	Understand and apply the rules of arithmetic operations and operator precedence to construct and evaluate expressions. Write and evaluate conditional statements and implement control flow through branching and iterative loops. Utilize 1-D and 2-D arrays, including character arrays and strings, for data storage and manipulation.	3, 4

	programs (no formal definition required)			
III	Function: Functions (including using built in libraries), Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference Recursion: Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quick sort or Merge sort.	**12**	Understand how to define functions, use built-in libraries, and pass parameters by value or reference. Explore recursion as a technique for solving problems, with examples like finding factorial, Fibonacci series, and Ackermann function. Learn about efficient sorting methods such as Quick sort or Mergesort, emphasizing their implementation and efficiency.	**3, 4**
IV	Structure: Structures, Defining structures and Array of Structures Pointers: Idea of pointers, Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation)	**6**	Understand how to define and use structures to organize related data elements in a program, enhancing clarity and organization. Grasp the concept of pointers, their definition, and application in structures, especially in self-referential structures, laying the foundation for understanding dynamic data structures like linked lists. Comprehend how arrays of structures allow efficient storage and manipulation of multiple related data sets, enabling scalable and organized data management in programs.	**4**
V	File handling (only if time is available, otherwise should be done as part of the lab)	**6**	Students will grasp the concepts of reading from and writing to files, including different modes such as read, write, and append. They will learn how to manipulate file data, including searching, updating, and deleting records within a file. Students will practice error handling techniques and understand best practices for file handling to ensure data integrity and security.	**4**
Practical Component				
Practical 1	a) Write a C program to find sum and average of three numbers.	**2**	Understand basic input/output operations, arithmetic operations in C, and how to compute simple statistics.	1, 2

	b) Write a C program to find the sum of individual digits of a given positive c) Write a C program to generate the first n terms of the Fibonacci sequence			
Practical 2	a) Write a C program to generate prime numbers between 1 to n. b) Write a C program to Check whether given number is Armstrong Number or Not.	**2**	Learns to implement a basic algorithm using loops and conditional statements to identify prime numbers within a specified range.	1, 3
Practical 3	a) Write a C program to evaluate algebraic expression (ax)/ (ax-b). b) Write a C program to check whethergiven number is perfect number or Not	**2**	Gains knowledge of iterative looping structures to sum proper divisors, use of conditional statements for validation, and understanding the concept of perfect numbers in number theory.	2, 3
Practical 4	a). Write a C program to check whether given number is strong number or not.	**2**	Understand the concept of strong numbers in programming. Demonstrate proficiency in C programming syntax and logic.	3
Practical 5	a) Write a C program to find the roots of a quadratic equation. b) Write a C program perform arithmetic operations using switch statement.	**2**	Learns to calculate roots using the quadratic formula, handling different cases (real, imaginary roots).	3, 4
Practical 6	a) Write a C program to find factorial of a given integer using non-recursive Function. b) Write a C program to find factorial of a given integer using recursive function.	**2**	Students will demonstrate the ability to write a C program that calculates the factorial of a given integer using iterative control structures and function calls, understanding fundamental concepts of loops, conditionals, and function definitions in C programming.	3, 5
Practical 7	a) Write C program to find GCD of two integers by using recursive function. b) Write C program to find GCD of two integers using non-recursive function.	**2**	By implementing a C program to find the GCD of two integers using recursion, learners understand recursion's application in solving mathematical problems efficiently, demonstrating proficiency in recursive function design and understanding of basic number theory concepts.	3, 4
Practical 8	a) Write a C program to find both the largest and smallest number in a list of Integers	**2**	Learn to iterate through an array, compare elements, and track both maximum and minimum values efficiently.	4, 5

	b) Write a C Program to Sort the Array in an Ascending Order. c) Write a C Program to find whether given matrix is symmetric or not.			
Practical 9	a) Write a C program to perform addition of two matrices. b) Write a C program that uses functions to perform Multiplication of Two Matrices.	**2**	Understands matrix representation in C, handling multidimensional arrays, and implementing basic arithmetic operations.	4, 3
Practical 10	a) Write a C program to use function to insert a sub-string in to given main string from a given position. b) Write a C program that uses functions to delete n Characters from a give position in a given string.	**2**	Develop proficiency in C programming by implementing a function that inserts a substring into a main string at a specified position, demonstrating effective use of string manipulation and function abstraction.	3, 5
Practical 11	a) Write a C program using user defined functions to determine whether the given string is palindrome or not. b) Write a C program that displays the position or index in the main string S where the sub string T begins, or - 1 if S doesn't contain T	**2**	By writing a C program using user-defined functions to check if a string is a palindrome, students will understand string manipulation, function definition, and logic implementation to verify symmetry in textual data.	2, 3
Practical 12	a) Write C program to count the number of lines, words and characters in a given text. b) Write a C program to find the length of the string using Pointer	**2**	Developed proficiency in C programming by implementing a program to accurately count lines, words, and characters in a provided text using efficient file handling and string manipulation techniques.	3, 4
Practical 13	a) Write a C program to Display array elements usingcalloc( ) function. b) Write a C Program to Calculate Total and Percentage marks of a student using structure.	**2**	Understand dynamic memory allocation in C using calloc() to allocate memory for arrays, and learn how to iterate through and display array elements	4, 5
Practical 14	a) Write a C program that uses functions and structures to perform the following operations: i) Reading a complex number ii) Writing a complex number	**2**	Utilize input functions to capture real and imaginary parts separately into a structure. Output the stored complex number using formatted printing with proper notation.	2, 3

	iii) Addition of two complex numbers iv) Multiplication of two complex numbers b). Write a C program to display the contents of a file			
Practical 15	a) Write a C program to copy the contents of one file to another. b) Write a C program to merge two files into a third file. Write a C program to reverse the first n characters in a file	**2**	Developed proficiency in handling file operations in C, demonstrating the ability to read from and write to files, ensuring accurate data transfer between file streams	3, 4

TEXT BOOKS:

T1: E. Balaguruswamy, Programming in ANSI C, TataMcGraw-Hill.
T2: Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.

REFERENCE BOOKS:
R1: Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill

OTHER LEARNING RESOURCES:
O1: Coursera online platform for Mathematical learning
O2: Mathkind Quality *Math Education* portal

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the concept of computer system, analyze a given problem, develop an algorithm, fundamental programming constructs, identify data representation formats, and describe operators and their precedence, associativity	PO1,PO2,PO3,PO4,PO5,PO11
2	Apply branching and loop statements in problem solving.	PO1,PO2,PO3,PO4,PO5,PO11
3	Apply the concepts of homogeneous derived data types, heterogeneous data types, strings and functions in programming.	PO1,PO2,PO3,PO4,PO5,PO11
4	Understand the concept of pointers and apply it in programming.	PO1,PO2,PO3,PO4,PO5,PO11
5	Apply the concept of file handling in C programming.	PO1,PO2,PO3,PO4,PO5,PO11

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS1102R	Problem Solving Using C	CO1	3	3	2	1	2						1	
		CO2	3	3	2	1	2						1	
		CO3	3	3	3	1	2						1	
		CO4	3	3	2	1	1						1	
		CO5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**Engineering Physics**								
Course code	**24BTCS1103R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:40T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Basic knowledge of Physics	**Co-requisite**	Mathematics						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. To understand the theories of physics 2. To apply the concepts in practical problems 3. To understand the physics of any process								
CO1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.								
CO2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.								
CO3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.								
CO4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.								
CO5	Understand the basic characteristics of transformers and electrical machines.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Electrostatics in vacuum	**8**	Understand and apply Coulomb's Law to calculate the electrostatic force between point charges in a vacuum. Analyze electric fields and potentials created by various charge distributions using principles of electrostatics.	**2**
II	Magneto statics	**8**	Magnetostatics studies magnetic fields in systems with steady currents, focusing on the behavior and interactions of magnetic fields in static conditions. It explains phenomena like the magnetic field produced by a current-carrying conductor and the forces between magnetic dipoles.	**3**
III	Faraday's law	**8**	Faraday's law of electromagnetic induction states that a change in magnetic flux through a circuit induces an electromotive force (EMF) in the circuit. This induced EMF is proportional to the rate of change of the magnetic flux.	**3,5**
IV	Displacement current, Magnetic field due to time-dependent electric field and Maxwell's equations.	**8**	Understand the concept of displacement current as a term added to Ampere's law to account for the changing electric field in non-conductive regions. Recognize its role in ensuring the continuity of current in Maxwell's equations and the propagation of electromagnetic waves.	**3**

V	Electromagnetic waves	**8**	Electromagnetic waves are transverse waves composed of mutually perpendicular electric and magnetic fields oscillating perpendicular to the direction of wave propagation. They encompass a wide range of frequencies, from radio waves to gamma rays, each characterized by its wavelength and frequency.	**5**
Practical Component				
Practical 1	To verify the ohm's law and hence determine the unknown resistance of the given material of the wire.	**4**	Verification of Ohm's Law through experimental data to determine the resistance of a given wire material	3
Practical 2	To find the value of a given resistance by using meter bridge.	**5**	Understanding how to determine the resistance of a given resistor using a meter bridge based on null point detection	4
Practical 3	To convert the galvanometer in to voltmeter and hence calibrate it with a standard resistance and ammeter.	**5**	To convert a galvanometer into a voltmeter, learn to connect a high resistance in series with the galvanometer and calibrate it using a standard resistance and ammeter to measure the voltage accurately.	5
Practical 4	To determine the internal resistance of a cell by using potentiometer.	**6**	Understand and apply the principle of a potentiometer to accurately measure the internal resistance of a cell	3
Practical 5	To determine the frequency of a tuning fork by Melde's apparatus.	**5**	Determine the frequency of a tuning fork using Melde's apparatus by observing the formation of stationary waves on a stretched string and 2calculating it from the wave parameters.	2
Practical 6	To determine the moment of inertia of a body about an axis passing through its center of gravity and perpendicular to its length.	**2**	To determine the moment of inertia of a body about an axis passing through its center of gravity and perpendicular to its length, the learning outcome is: "Calculate the moment of inertia using integral calculus, considering the mass distribution and geometric properties of the body."	4
Practical 7	To determine the ECE of copper by using copper voltmeter and ammeter.	**3**	To determine the Electrochemical Equivalent (ECE) of copper using a copper voltmeter and ammeter, students will measure the mass of copper deposited during electrolysis and relate it to the charge passed through the solution to calculate the ECE.	5

TEXT BOOKS:

T1: Introduction to Electrodynamics. David Griffiths. Prentice Hall, Upper Saddle River, New Jersey, 07458

T2: Basic Laws of Electromagnetism. IE IRODOV

REFERENCE BOOKS:

R1: Principles of physics. Halliday Resnick OTHER LEARNING RESOURCES

R2: https://www.sciencedirect.com/science/article/pii/S0951832022005142

OTHER LEARNING RESOURCES:

O1: Coursera Physics Courses

O2: edX Physics Courses

O3: MIT Open Course Ware Physics

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
5	Understand the basic characteristics of transformers and electrical machines.	PO1, PO2,PO3,PO4,PO5,PO10,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS1 103R	Engineering Physics	CO1	3	3	2	1	2						1	
		CO2	3	3	2	1	2						1	
		CO3	3	3	3	1	2						1	
		CO4	3	3	2	1	1						1	
		CO5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**Basic Electrical and Electronics Engineering**								
Course code	**24BTCS1104R**	**Total credits:3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives	1. To introduce the fundamental principles and concepts of electrical and electronic systems. 2. To develop the ability to analyze and solve basic electrical and electronic circuits. 3. To familiarize students with the working principles of electrical machines and semiconductor devices. 4. To provide an understanding of digital logic concepts and their applications in designing digital systems. 5. To impart knowledge of measurement techniques and instruments used in electrical and electronic systems.								
CO1	Understand and apply the basic principles of electrical circuits and AC/DC analysis.								
CO2	Demonstrate knowledge of the construction and working principles of electrical machines and devices.								
CO3	Analyze and design basic electronic circuits using diodes, transistors, and operational amplifiers.								
CO4	Apply digital logic techniques to solve problems and design simple digital circuits.								
CO5	Select appropriate measuring instruments for electrical and electronic applications and explain their working principles.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Basics of Electrical Engineering :** Fundamental Concepts: Charge, Current, Voltage, Power, and Energy. DC Circuits: Ohm's Law, Kirchhoff's Laws, Series and Parallel Circuits, Mesh and Nodal Analysis. AC Fundamentals: Sinusoidal Waveforms, RMS and Average Values, Power Factor, Single-Phase and Three-Phase Circuits	**7**	Understand the fundamental electrical concepts and laws. Analyze and solve basic DC and AC circuit problems.	**2,3**
II	**Electrical Machines:** Transformers: Principle, Construction, EMF Equation, and Efficiency. DC Machines: Construction, Working Principle, and Applications of DC Generators and Motors. Induction Motors: Basics of Single-Phase and Three-Phase Induction Motors.	**8**	Explain the working principles of electrical machines. Apply knowledge of electrical machines to real-world applications.	**2, 3**
III	**Basics of Electronics:**	**7**	Understand the behaviour and characteristics of basic semiconductor devices.	**2, 3,4**

	Semiconductor Theory: P-N Junction Diode, V-I Characteristics, Zener Diode. Transistors: Bipolar Junction Transistor (BJT) - Configuration, Characteristics, and Applications. Basics of Integrated Circuits and Operational Amplifiers (Op-Amps): Working and Applications.		Utilize semiconductor devices in simple circuits.	
IV	**Digital Electronics:** Number Systems: Binary, Octal, Decimal, and Hexadecimal. Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR - Truth Tables and Applications. Boolean Algebra and Simplification of Boolean Expressions. Basic Digital Circuits: Half Adder, Full Adder, Multiplexers, and Flip-Flops.	**7**	Understand and apply digital logic concepts in basic circuits. Design simple digital circuits for computational tasks.	**2,3,4**
V	**Electrical and Electronic Measurements:** Measurement Basics: Measurement of Current, Voltage, Power, and Energy. Instruments: Working Principles of Ammeters, Voltmeters, Watt meters, and Multi meters	**7**	Explain the working principles of measurement devices. Select appropriate instruments for various electrical and electronic measurements.	**2,3,4**
Practical Component				
Practical 1	Verification of Ohm's Law	**2**	Verify Ohm's Law and understand its application in DC circuits.	1,2
Practical 2	Series and Parallel Circuit Analysis	**2**	Analyze series and parallel resistive circuits and compute voltage, current, and power.	3,4
Practical 3	Kirchhoff's Voltage and Current Laws	**2**	Validate Kirchhoff's Laws and apply them to solve electrical networks.	2,3
Practical 4	Measurement of Power and Power Factor in AC Circuits	**3**	Measure power and power factor in single-phase AC circuits using appropriate instruments.	3
Practical 5	Open-Circuit and Short-Circuit Tests on a Transformer	**3**	Perform OC and SC tests on a transformer and calculate efficiency and voltage regulation.	3,4
Practical 6	Speed Control of a DC Motor	**3**	Analyze the speed control methods of DC motors using armature and field control techniques.	1,2
Practical 7	Characteristics of PN Junction and Zener Diode	**3**	Understand the V-I characteristics of PN junction and Zener diodes and their applications.	1,3

Practical 8	Input and Output Characteristics of a BJT in CE Configuration	**3**	Analyze the input and output characteristics of a BJT and determine its parameters in CE mode.	2,3
Practical 9	Design and Implementation of Half-Wave and Full-Wave Rectifiers	**3**	Design and analyze rectifier circuits for AC to DC conversion and understand ripple factor.	3
Practical 10	Realization of Logic Gates Using Universal Gates	**2**	Construct and validate the truth tables of basic gates using NAND and NOR gates.	3,4
Practical 11	Implementation of Boolean Functions Using Logic Gates	**2**	Simplify and implement Boolean expressions using basic logic gates.	3
Practical 12	Measurement of Electrical Quantities Using Multi-meter	**2**	Use a digital multi-meter to measure voltage, current, resistance, and continuity.	3,4

Textbooks:

T1 : Hughes, E., Smith, I. M., Hiley, J., & Brown, K. "Electrical and Electronic Technology", 12th Edition, Pearson Education, 2016.

T2: Boylestad, R. L., & Nashelsky, L. "Electronic Devices and Circuit Theory", 11th Edition, Pearson Education, 2015.

T3: Mittle, V. N., & Mittal, A. "Basic Electrical Engineering", McGraw-Hill Education, 3rd Edition, 2017.

Reference Books:

R1 : Hayt, W. H., Kemmerly, J. E., & Durbin, S. M. "Engineering Circuit Analysis", 8th Edition, McGraw-Hill Education, 2012.

R2: Sedra, A. S., & Smith, K. C. "Microelectronic Circuits", 7th Edition, Oxford University Press, 2014.

R3: Floyd, T. L. "Digital Fundamentals", 11th Edition, Pearson Education, 2015.

R4: Sawhney, A. K. "A Course in Electrical and Electronic Measurements and Instrumentation", Dhanpat Rai & Co., 2015.

Other Learning Resources:

O1: NPTEL Lectures on Basic Electrical Circuits by Prof. L. Umanand, IISc Bangalore.
O2: NPTEL Lectures on Digital Circuits by Prof. S. Srinivasan, IIT Madras.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand Kirchoff's laws to solve complex DC circuits, employing Thevenin, Norton, and Superposition theorems effectively.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
2	Analyze single-phase AC circuits, applying phasor representation and power concepts with precision and understanding.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
3	Evaluate transformer performance, considering losses, regulation, and efficiency, and apply knowledge to three-phase transformer connections.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12

4	Critically analyze three-phase induction motor operation, interpreting torque-slip characteristics, losses, and efficiency for practical applications.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
5	Understand the design of power converters and demonstrate expertise in electrical installations, including safety devices and battery types.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS1104R	**Basic Electrical and Electronics Engineering**	CO1	3	3	3	2	2	1			1	2	1	3
		CO2	3	3	3	2	2	1			1	2	1	3
		CO3	3	3	3	2	2	1			1	2	1	3
		CO4	3	3	3	2	2	1			1	2	1	3
		CO5	3	3	3	2	2	1			1	2	1	3

SEMESTER – I									
Course Title	**Workshop for Engineers**								
Course code	**24BTCS1105R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15T+ 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. Learning this course will lead you to understand basic concepts of workshop and manufacturing 2. Apply fundamental knowledge of workshop and manufacturing in day-to-day life 3. Recognize components using different materials.								
CO1	Introduction to various manufacturing methods like casting, Forming, machining etc.								
CO2	Application of computer coding in automation of Machines.								
CO3	Introduction to Carpentry & fitting operations and its application in industries.								
CO4	Different machining operations like turning, milling.								
CO5	Learning the different types of welding and its field of application.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Manufacturing Methods:** Casting, forming, machining, joining, advanced manufacturing methods	3	Manufacturing methods include casting, forming, machining, and joining, which involve shaping and assembling materials into final products. Advanced manufacturing methods leverage modern technologies such as additive manufacturing, laser machining, and automation to enhance efficiency, precision, and product quality.	1,2,3
II	**CNC machining, Additive Manufacturing: o**verview of CNC machining process, overview of additive manufacturing.	3	CNC Machining: An automated manufacturing process that uses pre-Programmed computer software to control machinery and tools to shape, cut, and drill materials with high precision. Additive Manufacturing: A process of creating objects by adding material layer by layer, commonly known as 3D printing, allowing for complex geometries and reduced material waste.	3,4
III	**Carpentry & Fitting operations:** Carpentry tools, carpentry operations, fitting tools, fitting operations	3	Understand the use of various carpentry tools and perform fundamental carpentry operations. Gain proficiency in using fitting tools and executing essential fitting operations.	3,4
IV	**Machining operations:** Turning, milling, turning processes, milling processes	3	Understand the principles, equipment, and techniques involved in turning and milling operations. Gain proficiency in identifying and performing specific processes related to turning and milling, including	4

			tool selection, setup, and operation.	
V	**Welding:** Arcwelding & gas welding, brazing	3	Understand the principles, techniques, and applications of arc welding and gas welding. Develop skills in brazing, including proper material selection and joint preparation for effective metal joining.	4
Practical Component				
Practical 1	**A brief introduction of workshop:** Machine shop, Fitting shop, carpentry shop, welding shop	5	Gain practical skills and foundational knowledge in machine operations, fitting, carpentry, and welding techniques essential for manufacturing and construction industries.	1,2
Practical 2	**Machine shop:** Plain turning, Taper turning, Step turning	10	Understand and execute the basic turning operations on a lathe, including plain turning, taper turning, and step turning, ensuring precision and adherence to specified dimensions. Develop skills in setting up and operating a lathe, selecting appropriate tools and parameters for each operation, and maintaining safety protocols throughout the machining process.	2,3
Practical 3	**Carpentry shop:** Dovetail joint, T-lap joint, Cross-lap joint, Corner-lap joint	5	Gain practical skills in creating dovetail, T-lap, cross-lap, and corner-lap joints, essential for woodworking projects. Understand the structural benefits and appropriate applications of each joint type in carpentry.	3,4
Practical 4	**Welding shop:** Arcwelding (Butt joint,T-joint, Lap joint, Corner joint)	5	The welding shop training aims to develop skills in performing arc welding on various joint types, including butt joints, T-joints, lap joints, and corner joints. Trainees will learn proper techniques, safety practices, and quality control measures to produce strong, reliable welds in different configurations.	4
Practical 5	**Fitting shop:** Angle fitting, square fitting, corner fitting	5	Participants will gain proficiency in identifying and selecting appropriate angle, square, and corner fittings for various construction needs. Participants will demonstrate competence in assembling and integrating angle, square, and	5

			corner fittings into structural frameworks effectively.	

TEXT BOOKS:

T1: Elements of Workshop Technology, Vol. I 2008 and Vol. II 2010,
T2: Manufacturing Engineering and Technology

REFERENCE BOOKS:

R1: Principles Manufacturing Technology – I, Pearson Education, 2008

R2: Processes and Materials of Manufacture, Roy A. Lindberg 4th edition, Prentice Hall India, 1998

OTHER LEARNING RESOURCES:

O1: Coursera workshop & Manufacturing Courses
O2: edX manufacturing technology Courses

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.	PO1,PO2,PO3,PO5,PO7
2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.	PO1,PO5,PO11,PO12
3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.	PO1,PO3,PO9
4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.	PO1,PO2,PO5,PO9
5	Understand the basic characteristics of transformers and electrical machines.	PO1,PO2,PO3,PO5,PO6

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS1105R	Workshop for Engineers	CO1	3	3	2	1	2						1	
		CO2	3	3	2	1	2						1	
		CO3	3	3	3	1	2						1	
		CO4	3	3	2	1	1						1	
		CO5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**MOOCS I: Introduction to Java**								
Course code	**24MOCS1101R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 40T	**0**	**0**	**0**	**8**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To introduce students to the fundamental concepts of object-oriented programming using Java. 2. To familiarize learners with the syntax and basic constructs of the Java programming language. 3. To enable students to understand and apply key programming concepts such as variables, data types, operators, and control flow statements. 4. To provide hands-on experience in writing, compiling, and executing Java programs. 5. To develop problem-solving skills by implementing basic programming logic using Java.								
CO1	Understand the basic structure and syntax of Java programming language.								
CO2	Write simple Java programs to perform basic input and output operations.								
CO3	Apply fundamental programming concepts such as variables, data types, and control structures in Java.								
CO4	Develop small programs using loops and conditional statements in Java.								
CO5	Create and run basic Java programs using an Integrated Development Environment (IDE).								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Java :** Introduction to Java and its features, Overview of the Java Virtual Machine (JVM), Java Development Kit (JDK) and Integrated Development Environment (IDE) setup, Writing, compiling, and running your first Java program	**8**	Understand the evolution, features, and basic structure of Java programs. Set up the Java environment and execute simple programs.	**1,2,3**
II	**Basics of Java Programming:** Java syntax and data types, Variables, constants, and type casting, Operators (arithmetic, relational, logical, assignment, etc.), Input and output operations in Java	**8**	Describe the fundamental syntax and elements of Java programming. Apply basic operations using operators and variables in Java programs.	**2,3**
III	**Control Flow Statements:** Decision-making statements (if, if-else, switch), Looping constructs (for, while, do-while loops), Nested loops and the use of break/continue statements	**8**	Explain and use decision-making and looping constructs in Java programs. Write programs to solve problems using control flow statements.	**3,4**

IV	**Object-Oriented Programming in Java:** Introduction to Object-Oriented Programming (OOP) principles, Classes and objects in Java, Constructors and method overloading, Access modifiers and the this keyword	**8**	Describe the principles of OOP and their implementation in Java. Develop programs using classes, objects, and constructors.	**4**
V	**Arrays and Strings in Java :** Co Introduction to arrays (single-dimensional and multi-dimensional arrays), Basic operations on arrays (initialization, traversal, and manipulation), Strings in Java and common string operations, Introduction to the StringBuffer and StringBuilder classes	**8**	Demonstrate the use of arrays and strings in Java programs. Write programs to perform operations on arrays and strings.	**5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basic structure and syntax of Java programming language.	PO1,PO2,PO3,PO4,PO5,PO12
2	Write simple Java programs to perform basic input and output operations.	PO1,PO2,PO3,PO4,PO5,PO12
3	Apply fundamental programming concepts such as variables, data types, and control structures in Java.	PO1,PO2,PO3,PO4,PO5,PO12
4	Develop small programs using loops and conditional statements in Java.	PO1,PO2,PO3,PO4,PO5,PO12
5	Create and run basic Java programs using an Integrated Development Environment (IDE).	PO1,PO2,PO3,PO4,PO5,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24MOCS1101R	Introduction to Java	CO1	2	2	3	1	1							1
		CO2	2	1	3	1	1							1
		CO3	2	1	2	1	1							1
		CO4	2	1	2	1	1							1
		CO5	2	2	2	1	1							1

SEMESTER – I									
Course Title	**PDP (Introductory English)**								
Course code	**24UBPD1104R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. To capacitate the students with mastery over Basic English grammar. 2. To enable the students to communicate confidently with a focus on listening a speaking skills. 3. With the help of the basics of Phonetics, the students will be able to pronoun words correctly. 4. To interact successfully and with decorum.								
CO1	Analyze and apply the rules of Parts of Speech in constructing grammatically correct sentences.								
CO2	Evaluate sentence structures, employ varied sentence types, and demonstrate effective comprehension skills								
CO3	Assess listening skills, identify factors influencing listening, and implement strategies for improved listening.								
CO4	Demonstrate effective speaking skills, including self-introduction, pronunciation, and extempore speech delivery.								
CO5	Apply communication theories, recognize barriers, and enhance skills for diverse communication situations.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar** 1. Parts of Speech 2. Articles 3. Auxiliary Verbs 4. Affirmative and Negative Sentences	**5**	You'll learn how different parts of speech function in sentences, including articles and auxiliary verbs, enabling you to construct both affirmative and negative sentences effectively.	**1**
II	**Grammar** 1. Determiners 2. Sentence Construction 3. Types of Sentences (Assertive, Imperative, etc.) 4. Degree of Comparison 5. Comprehension Exercises	**7**	Master determiners, sentence construction, and types of sentences (assertive, imperative, etc.), along with understanding degrees of comparison. Practice comprehension exercises to reinforce understanding and application of these concepts effectively.	**3**
III	**Listening Skills** 1. What is listening? 2. The Process of Listening 3. Factors that adversely affect Listening 4. Difference between	**6**	By studying listening skills, you will understand the distinction between listening and hearing, recognize factors that hinder effective listening, appreciate the importance of active listening, and	**5**

	Listening and Hearing, 5. Purpose and Importance of Effective Listening 6. How to Improve Listening Process.		learn strategies to enhance the listening process	
IV	**Speaking Skills** 1. Introducing yourself 2. Self-discovery 3. Basics of Phonetics, pronunciation 4. Extempore speech 5. Video Recording for Self reflection	**6**	Developing speaking skills through self-introduction, phonetics, and extempore speaking, enhanced by video reflections for self-improvement and self-discovery.	**4**
V	**Communication Skills** 1. Introduction to Communication, 2. Importance of Communication Skills, 3. Purpose of Communication, 4. Types of Communication, 5. Formal and informal communication 6. Importance of Communication, 7. Barriers to Communication, 8. How to improve/ tips to improve Communication skills. 9. Responding to different questions in various situations(formal/informal)	**6**	By studying communication skills, learners gain a comprehensive understanding of its importance, types, barriers, and strategies for improvement, enabling effective interaction in various contexts.	**3**

TEXT BOOKS:

T1: Chaturvedi, P.D., ChaturvediMukesh, 2011.Business Communication: Concepts, Cases and Applications, second edition, Pearson,Noida.

T2: Alex K., Chand, S, 2009. Soft Skills: Know Yourself and Know the World, first edition, S. Chand & Company Ltd.: NewDelhi

REFERENCE BOOKS:

R1: Quirk, Randolp. (2010) A Comprehensive Grammar of the English Language by Randolph Quirk, Sidney Greenbaum, Pearson EducationIndia

R2: Marks, Jonathan. (2017) IELTS Advantage Speaking and Listening Skills: A step-by-step guide to a high IELTS speaking and listening score. Book + CD-ROM, Delta Publishing by Klett

OTHER LEARNING RESOURCES:
O1: Clapingo english Courses
O2: English partner India's Largest Online English Learning Platform

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Analyze and apply the rules of Parts of Speech in constructing grammatically correct sentences.	PO10
2	Evaluate sentence structures, employ varied sentence types, and demonstrate effective comprehension skills	PO2,PO10
3	Assess listening skills, identify factors influencing listening, and implement strategies for improved listening.	PO9,PO10
4	Demonstrate effective speaking skills, including self-introduction, pronunciation, and extempore speech delivery.	PO10
5	Apply communication theories, recognize barriers, and enhance skills for diverse communication situations.	PO6,PO10

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24UBPD1104R	**Introductory English**	CO1										3		
		CO2		2								3		

	for engineers	CO 3									2	3		
		CO 4										3		
		CO 5						2				3		

SEMESTER – I									
Course Title	**Co-curricular Activities**								
Course code	**24UBCC1101**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30S	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. It is to develop the social and soft skills and to promote a holistic development of the learners. 2. Participating in debate clubs, drama, or public speaking events enhances verbal and non-verbal communication skills. 3. Taking leadership roles in student organizations or event planning committees fosters leadership and teamwork abilities.								
CO1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.								
CO2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.								
CO3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.								
CO4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.								
CO5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Discover the exciting world of engineering through hands-on projects and workshops. Gain practical skills in robotics, coding, and innovation, preparing for future STEM success.	30	Explore the dynamic field of engineering through interactive projects and workshops. Develop practical skills in robotics, coding, and innovation. Prepare for future success in STEM disciplines.	1,2,3,4,5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Connect and adapt cultural diversity among communities.	PO1, PO6, PO8, PO10, PO12
2	Enhance team for working toward a shared vision	PO6, PO9, PO10, PO12
3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals	PO2, PO4, PO10, PO12

4	Learn to effectively communicate, delegate responsibilities and motivate team members.	PO2, PO6, PO10, PO12
5	Develop strong teamwork and collaboration skills by engaging in group activities.	PO1, PO2, PO6, PO8, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24UBCC1101R	Co-Curricular Activities	CO1	2					3		1		2		1
		CO2						1			3	3		1
		CO3		3		3						2		2
		CO4		3				3				2		2
		CO5	2	3				3		1				2

SEMESTER – II									
Course Title	**Ordinary Differential Equations and Differential Calculus**								
Course code	**24BTCS1201R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**3**	**1**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science & Engineering**								
Semester	**Winter/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. To make students understand and solve first order and second order ordinary differential equations. 2. To help understand and solve higher order differential equations. 3. To provide the application on differential calculus. 4. To provide an overview of partial derivatives and the different notions attached to them. 5. To help understand the concept of multiple integrals and the ways to find the area of different types of curves.								
CO1	Enabling problem solving skills of ordinary differential equations of various order.								
CO2	Enable to find the partial order derivative of functions of two or more variables.								
CO3	Analyse the complex analysis and solutions.								
CO4	Solve problems related to differentiation under integral sign.								
CO5	Analyze and find the area of two curves using multiple integrals								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**First order ordinary differential equations:** Exact, linear and Bernoulli's equations- Euler's equations- Equations not of first degree: equations solvable for p- equations solvable for y- equations solvable for x and Clairaut's type.	**5**	Solve initial value problems and understand their significance in applications. Interpret the existence and uniqueness of solutions based on initial conditions.	**2,3**
II	**Ordinary differential equations of higher orders:** Second order linear differential equations with variable coefficients- method of variation of parameters, Cauchy-Euler equation- Power series solutions- Legendre polynomials- Bessel functions of the first kind and their properties.	**7**	Formulate real-world problems as first-order ODEs (e.g., population dynamics, cooling/heating problems, mixing problems). Interpret and analyze the solutions in the context of the modelled phenomena.	**2,3,4**
III	**Partial Differentiation-I** Introduction to partial derivatives- its meaning- standard rules of finding partial derivatives of functions of two or more variables- Introduction to homogeneous functions- To check whether a given function is homogeneous or not- Introduction to Euler's homogeneous function- application of Euler's homogeneous function.	**6**	Understand and interpret direction fields (slope fields). Analyze the behaviour of solutions without solving the equation explicitly. Determine the stability of equilibrium solutions.	**2,3,4,5**

IV	**Partial differentiation-II** Jacobian- properties of Jacobian- Jacobian of Implicit Functions- Taylor's theorem for function of two variables- Maclaurin's series for function of two variables- Maxima and Minima of functions of two variables- Conditions for f(x, y) to be Maximum or Minimum of two variables without constraints and Lagrange's method- Differentiation under integral sign.	**6**	Solve first-order ODEs using separation of variables. Apply integrating factor methods to solve linear first-order ODEs. Utilize exact equations and understand the conditions for exactness.	**2,3,4,5**
V	**Multiple integrals** Double and Triple integrals – Change of order of integration in double integrals – Change of variables to polar, cylindrical and spherical coordinates.	**6**	Evaluate double integrals by iterated integration (Fubini's Theorem). Change the order of integration in double integrals. Compute double integrals in polar coordinates.	**2,3,4,5**

TEXT BOOKS:

T1: C Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley and Sons, 2006
T2: N. P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.

REFERENCE BOOKS:

R1: S. L. Ross, Differential Equations, 3rd Ed. Wiley India, 1984.
R2: Dr. V. P. Mishra, Dr. Jyoti Sinha, Pratibha Mishra, Advanced Engineering Mathematics, Bhavya Books, Delhi.

OTHER LEARNING RESOURCES:
O1: Coursera: Platforms like Coursera offer courses such as "Mathematics for Engineers" which cover topics ranging from calculus to differential equations and linear algebra.
O2: edX: Courses like "Engineering Mathematics" provide an in-depth study of mathematical techniques important in engineering disciplines

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enabling problem solving skills of ordinary differential equations of various order.	PO1,PO2,PO3,PO12
2	Enable to find the partial order derivative of functions of two or more variables.	PO1,PO2,PO3
3	Analyse the complex analysis and solutions.	PO1,PO2,PO3
4	Solve problems related to differentiation under integral sign.	PO1,PO2,PO3
5	Analyze and find the area of two curves using multiple integrals	PO1,PO2,PO3

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS1201R	**Ordinary Differential Equations and Differential Calculus**	CO1	3	3	1									1
		CO2	3	2	2									
		CO3	3	2	2									
		CO4	3	3	1									
		CO5	3	3	2									

SEMESTER – II									
Course Title	**Object Oriented Programming using JAVA**								
Course code	**24BTCS1202R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science & Engineering**								
Semester	**Winter/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1.To enable students to understand the basic object oriented programming concepts and apply them in problem solving. 2.Additionally, to illustrate among students the inheritance concepts for reusing the program and also to enable students. 3.To develop GUI-based applications using AWT, Swing and Event handling.								
CO1	Understand object-oriented programming concepts and execute them proficiently in Java.								
CO2	Apply building blocks of OOPs language, including inheritance, packages, and interfaces, analysing real-world problems in their context.								
CO3	Utilize multithreading, exception handling, and other OOPs concepts effectively in object-oriented programs.								
CO4	Apply various exception handling methods in programming, demonstrating competence in error management.								
CO5	Develop interactive and GUI-based Java applications through project-based learning, showcasing practical application of Java programming skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Java Overview:** Java Introduction, Platform Independence, JVM & JDK, Data types, Operators, If, else statement, Switch condition, while, do-while, for loop, break and continue statement. **Array and String:** Single Array & Multidimensional Array, Library Classes-String, String Buffer & Wrapper Class, Command line arguments and Various String Operations.	**5**	Describe the concept of platform independence in Java and how the Java Virtual Machine (JVM) enables it. Differentiate between the Java Virtual Machine (JVM) and the Java Development Kit (JDK), and understand their roles in Java application development. Define and declare single-dimensional arrays in Java, understanding array initialization, accessing elements, and iterating through array elements.	**1,2,3**
II	**Classes, Objects and Methods:** Class and Object, Object reference, Constructor: Constructor Overloading,	**7**	Define classes and objects in Java, understanding the concept of encapsulation and how classes serve as blueprints for creating objects. Declare and define classes in Java, including instance	**3,4,5,6**

	Method: Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize () method, Import statement, Static import, Access control, Nested class, Inner class, Anonymous inner class.		variables, constructors, methods, and access modifiers (public, private, protected). Create objects of a class using constructors and understand the process of object initialization, including default constructors and parameterized constructors.	
III	**Inheritance and Interfaces in Java:** Overview of Inheritance, inheritance in constructor, Inheriting Data members and Methods, Multilevel Inheritance – method overriding Handle multilevel constructors Explain super keyword, Stop Inheritance, Explain Final keywords, Creation and Implementation of an interface, Interface reference, instance of operator, Interface inheritance, Dynamic method dispatch, Abstract class, Comparison between Abstract Class and interface, inside of System.out.println – statements.	**6**	Differentiate between types of inheritance in Java, including single inheritance, multiple inheritance through interfaces, and hierarchical inheritance. Implement method overriding in subclasses to provide specialized implementations of methods inherited from superclasses, understanding runtime polymorphism.	**2,3,4,5**
IV	**Exception Handling in Java:** Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. JAVA File Handling: Overview of Different Stream (Byte Stream, Character stream), Readers and Writers class, FileClass, File Input Stream, File Output Stream, Input Stream Reader and Output Stream Writer class, File reader and writer class,	**6**	Define exceptions in Java and understand their role in handling runtime errors and abnormal conditions that may occur during program execution. Implement try-catch blocks to handle exceptions gracefully, preventing program termination and providing alternative behaviors or error messages.	**1,3,4,5**

	File Writer, Buffered Reader class.			
V	**Applet, AWT and Swing:** MVC Architecture, Applet: Applet Fundamental, Applet Architecture, Applet Skeleton, Requesting Repainting, Event Handling: various event handling mechanisms, Delegation Event Model, Events, Event Sources, Event Listeners, various classes related to event sources and event listeners, AWT: window fundamentals, creating frames, Adding removing various controls, Layout managers, Introduction To Swing, Applications and Pluggable look and feel, Basic swing components: Text Fields, Buttons, Toggle Buttons, Checkboxes, and Radio Buttons.	**6**	Define what Java applets are and understand their role in web-based applications, particularly in the context of early web development. Describe the lifecycle of an applet, including initialization (init()), start (start()), stop (stop()), and destruction (destroy()), understanding how applets interact with web browsers. Implement event handling in Java applets to respond to user interactions such as mouse clicks, keyboard input, and window events, using event listeners and adapters.	**4**
Practical Component				
Practical 1	Hands-on Introduction to Object-Oriented Programming: Practical exercises on creating classes and objects, Implementing encapsulation through practical examples, Coding exercises on constructors and destructors, Applying access modifiers (public, private, protected) in practical scenarios.	**6**	Define object-oriented programming (OOP) concepts such as classes, objects, inheritance, polymorphism, and encapsulation, understanding their role in software development. Create classes in Java, defining attributes (fields) and behaviors (methods) that represent real-world entities or concepts, and instantiate objects from these classes.	**1,2,4,5**
Practical 2	Practical Inheritance and Polymorphism: Coding exercises on implementing different types of inheritance (single, multi-level, and multiple), Practical examples of method overriding, Creating and using abstract classes and interfaces, Implementing	**8**	Implement different types of inheritance in Java: single inheritance (one subclass inherits from one superclass), multi-level inheritance (a subclass extends another subclass), and multiple inheritance (a subclass inherits from multiple superclasses using interfaces).	**2,4,5**

	static and dynamic binding through practical examples.			
Practical 3	Advanced OOP Concepts in Practice: Practical exercises on creating and using packages and namespaces, Implementing exception handling in real-world scenarios, Creating and managing multithreaded applications, Practical exercises on generics and collections.	**8**	Organize Java classes into packages to manage code structure and namespace, demonstrating how packages encapsulate related classes and provide access control. Develop practical exercises that involve creating, importing, and utilizing packages in Java applications, ensuring modularization, and facilitating code reuse and maintenance.	**1,2,3**
Practical 4	Software Design and Development in Action: Hands-on Object-Oriented Analysis and Design (OOAD) exercises, Implementing design patterns (e.g., Singleton, Factory, Observer) in practical scenarios, Software testing and debugging exercises, Code optimization and refactoring practical sessions.	**8**	Conduct hands-on exercises on OOAD principles, including identifying objects, defining classes, specifying relationships (inheritance, composition), and modeling behavior using UML diagrams (class diagrams, sequence diagrams, etc.). Implement commonly used design patterns (e.g., Singleton, Factory, Observer, Strategy) in practical scenarios, demonstrating their application to solve recurring design problems and improve code flexibility and maintainability.	**2,3,4,5**

TEXT BOOKS:

T1: Herbert Schildt and Dale Skrien, ‖Java Fundamentals – A comprehensive Introduction‖, McGraw Hill.

T2: Herbert Schildt, ―Java the complete reference, McGraw Hill, Osborne.

T3: T. Budd, ―Understanding Object-Oriented Programming with Java‖, Pearson Education.

REFERENCE BOOKS:

R1: P.J. Dietel and H. M. Dietel,―Java How to program ‖,Prentice Hall.

R2: P. Radha Krishna, ―Object Oriented programming through Java‖, CRC Press.

OTHER LEARNING RESOURCES:

O1: http://java.sun.com

O2: http://www.oracle.com/technetwork/java/index.html)

O3: http://java.sun.com/javase

O4: http://www.oracle.com/technetwork/java/javase/overview/index.html

O5: http://download.oracle.com/javase/7/docs/api/index.htmlE-Text Books:

O6: http://docs.oracle.com/javase/tutorial/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand object-oriented programming concepts and execute them proficiently in Java.	PO1,PO2,PO3,PO4,PO5,PO12
2	Apply building blocks of OOPs language, including inheritance, packages, and interfaces, analysing real-world problems in their context.	PO1,PO2,PO3,PO4,PO5,PO12
3	Utilize multithreading, exception handling, and other OOPs concepts effectively in object-oriented programs.	PO1,PO2,PO3,PO4,PO5,PO12
4	Apply various exception handling methods in programming, demonstrating competence in error management.	PO1,PO2,PO3,PO4,PO5,PO12
5	Develop interactive and GUI-based Java applications through project-based learning, showcasing practical application of Java programming skills.	PO1,PO2,PO3,PO4,PO5,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS1202R	Object Oriented Programming using JAVA	CO1	2	2	3	1	1							1
		CO2	2	1	3	1	1							1
		CO3	2	1	2	1	1							1
		CO4	2	1	2	1	1							1
		CO5	2	2	2	1	1							1

SEMESTER – II									
Course Title	**Data Structure and Algorithms**								
Course code	**24BTCS1203R**	**Total credits: 4** **Total hours:** **45T+30P**	**L**	**T**	**P**	**S**	**R**	**O / F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Fundamentals of programming logic	**Co-requisite**	Basics of C Programming						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. To impart the basic concepts of data structures and algorithms. 2. To understand basic concepts about stacks, queues, lists, trees and graphs. 3. To understand concepts about searching and sorting techniques								
CO1	Understand analysis of algorithms using asymptotic notations, and learn search technique								
CO2	Analyse algorithms on stacks and queues and their applications.								
CO3	Implement and analyse operations on linked lists and its variations and their applications.								
CO4	Apply tree terminologies and operations on different types of trees, with a focus on algorithmic analysis and practical applications.								
CO5	Evaluate and compare various sorting algorithms, hashing techniques, and graph theoretic concepts along with their complexity analysis.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction:** Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.	**8**	Understanding fundamental data structures, algorithm analysis, and search techniques, including complexities, for efficient problem-solving and resource management in computing.	1,2,3
II	**Stacks and Queues**: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.	**10**	Learning outcomes: Mastery of stack and queue ADTs, including operations and complexities, and their practical applications in expression conversion, evaluation, and various queue types with corresponding algorithms and analyses.	3,4
III	**Linked Lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked	**10**	Gain expertise in linked list structures: singly linked lists, doubly linked lists, circular linked lists, stack and queue implementations, and algorithmic complexities.	3,4

	Lists: all operations their algorithms and the complexity analysis.			
IV	**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.	**6**	Learning outcomes: Proficiency in basic tree terminology, operations on binary, threaded, AVL trees, and B/B+ trees, including algorithms and complexity analysis, with applications in various domains.	3,5
V	**Sorting and Hashing:** Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing. **Graph:** Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.	**6**	Master sorting (Selection, Bubble, Insertion, Quick, Merge, Heap), hashing principles, and graph operations (search, traversal) with complexity analyses.	3,4,5
Practical Component				
Practical 1	Array Operations- Program to perform basic array operations (insertion, deletion, search, update).	3	Learning outcome: Proficiency in performing array operations (insertion, deletion, search, update) essential for efficient data manipulation and algorithm implementation in programming.	2,3
Practical 2	Linked List Operations- Program to implement singly linked list operations (insertion, deletion, traversal). Program to implement doubly linked list operations (insertion, deletion, traversal). Circular Linked List- Program to implement circular linked list operations (insertion, deletion, traversal).	3	Mastery of implementing and manipulating singly, doubly, and circular linked lists through operations like insertion, deletion, and traversal in programming contexts.	2,3,4
Practical 3	Stacks and Queues- Stack Using Array- Program to implement stack operations using arrays (push, pop, peek). Stack Using Linked List- Program to implement stack operations using linked lists. Infix to Postfix Conversion	3	Proficiency in implementing stack operations using arrays and linked lists, and converting infix expressions to postfix using stacks, enhancing algorithmic	3,4

	Program to convert infix expression to postfix expression using stack.		understanding and programming skills.	
Practical 4	Queue Using Array- Program to implement queue operations using arrays (enqueue, dequeue, front, rear). Queue Using Linked List- Program to implement queue operations using linked lists. Circular Queue- Program to implement circular queue using arrays.	3	Mastery in implementing queue operations using arrays and linked lists, including circular queue implementation, crucial for efficient data management and algorithmic proficiency in programming.	4,5
Practical 5	Trees Binary Tree Creation Program to create a binary tree and perform pre-order, in-order, and post-order traversals. Binary Search Tree (BST) Operations c) Program to implement BST operations (insertion, deletion, search).	3	Ability to create and traverse binary trees using preorder, in-order, and post-order methods, and proficiency in implementing BST operations (insertion, deletion, search) for efficient data organization and manipulation in programming.	2,3,4
Practical 6	AVL Tree Implementation Program to implement AVL tree operations (insertion with rotations). Heap Implementation c) Program to implement a max-heap or min-heap and perform heap operations (insert, delete, heap if y).	3	Mastering AVL tree operations with rotations for balance and implementing heap structures with insertions, deletions, and maintenance in programming contexts.	2,3,4
Practical 7	Sorting Programs- Merge Sort Implementation: Write a C program to implement the merge sort algorithm to sort an array of integers. Quick Sort Implementation: Implement the quick sort algorithm in C to sort an array of integers. Heap Sort Implementation: Write a C program to implement the heap sort algorithm to sort an array of integers.	3	Proficiency in implementing and understanding merge sort, quick sort, and heap sort algorithms for efficient array sorting in programming.	3,5
Practical 8	Hash Table with Chaining: Implement a hash table using chaining for collision resolution in C. Include functions to insert, search, and display elements.	3	Learning to implement a hash table with chaining for collision resolution, including functions for insertion, searching, and displaying elements, crucial for efficient data management in C programming.	3,5
Practical 9	Hash Table with Linear Probing: Write a C program to implement a hash table using open addressing with	3	Proficiency in implementing a hash table using linear probing for collision	3,4

	linear probing for collision resolution. Include functions to insert, search, and display elements.		resolution, including functions for insertion, searching, and displaying elements in C programming, essential for efficient data management and retrieval.	
Practical 10	Hash Table with Quadratic Probing: Implement a hash table using open addressing with quadratic probing for collision resolution in C. Include functions to insert, search, and display elements	3	Learning to implement hash tables with quadratic probing for collision resolution, including insert, search, and display functions in C.	4,5

TEXT BOOKS:

T1: "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.

T2: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein**,** "Introduction to Algorithms" by Cormen et al.

T3: Seymour Lipschutz , "Data Structures, with C" by McGraw-Hill Education, Schaum's Outlines series.

REFERENCE BOOKS:

R1: Algorithms, "Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company

R2: "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

OTHER LEARNING RESOURCES:

O1: Coursera: "Data Structures and Algorithms" by UC San Diego and National Research University Higher School of Economics

O2: Udemy: "Master the Coding Interview: Data Structures + Algorithms" by Andrei Neagoie

O3: GeeksforGeeks: Offers tutorials, explanations, and problem-solving sessions on data structures and algorithms.

O4: Tutorialspoint: Provides comprehensive tutorials on data structures and algorithms.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand analysis of algorithms using asymptotic notations, and learn search technique	PO1, PO2, PO3, PO10, PO12
2	Analyse algorithms on stacks and queues and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Implement and analyse operations on linked lists and its variations and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

4	Apply tree terminologies and operations on different types of trees, with a focus on algorithmic analysis and practical applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Evaluate and compare various sorting algorithms, hashing techniques, and graph theoretic concepts along with their complexity analysis.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS1203R	**Data Structure & Algorithms**	**CO1**	2	2	1							1		1
		CO2	3	3	2	2	1					1		2
		CO3	3	3	3	1	1					1		2
		CO4	3	3	2	2	1					1		2
		CO5	3	3	2	2	1					1		2

SEMESTER – II										
Course Title	**Environmental Science**									
Course code	**24UBES1201R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science & Engineering**								
Semester	**Winter/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. Understanding Ecological Systems and Processes. 2. Analysing Environmental Issues and Solutions. 3. Promoting Environmental Stewardship and Sustainable Practices.								
CO1	Understand the relationships between natural and man-made systems.								
CO2	Develop critical thinking for shaping strategies (scientific, social, economic, administrative, and legal) for environmental protection, conservation of biodiversity, environmental equity, and sustainable development.								
CO3	Understand the consequences of human actions on the web of life, global economy, and quality of human life.								
CO4	Understand about Environment Pollution, Global Environment issues, Impact of Human Population and modern civilization on the Environment								
CO5	Analyse various aspects of human population, and the impact of the population growth on the environment.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Environment:** Definition and scope, components of environment, atmosphere, hydrosphere, lithosphere and biosphere, structure and composition, Life systems, pro and eukaryotic organizations, Metabolic principles; types of plants and animals. Producers, consumers and decomposers.	**7**	Identify and describe the major components of the environment: atmosphere (air), hydrosphere (water), lithosphere (earth's crust), and biosphere (living organisms), understanding their interactions and interdependencies. Define "environment" in the context of ecological and environmental sciences, exploring its scope as encompassing natural, built, and social environments.	**1,2,3**
II	**Ecology:** Terminology and approach, ecosystem, types of ecosystems; structure and function, mineral cycling, energy flow and trophic chains. Development and evolution.	**7**	Define key ecological terms such as ecosystem, community, population, niche, habitat, and biodiversity, understanding their roles in studying interactions between organisms and their environment. Explain the concept of an ecosystem as a functional unit consisting of biotic (living organisms) and abiotic (non-living environment) components interacting within a defined area.	**3,4**
III	**Environmental Pollution:** Sources, causes, assessment,	**7**	Identify and categorize sources of pollution such as	**3,4**

	effect, prevention and control of water pollution, air pollution noise and land pollution. Strategies of management, concept of sustainability. Energy, e n v i r o n m e n t , and their relationship with human activities. Water Resources and utilization, forest resources.		industrial emissions, agricultural runoff, urban waste, and vehicular emissions, understanding their contributions to environmental degradation. Conduct assessments of environmental pollution using scientific methods and tools, including monitoring techniques, data analysis, and environmental impact assessments (EIAs).	
IV	**Global Environmental Problems:** Human health, settlements, management of rivers, lakes, forests, wild life and catchments. Role of society, NGO and Govt. agencies. Concept of urbanization and green cities Global Warming, greenhouse causes and effects, carbon Sequestration.	**7**	Analyze the impact of environmental pollution and degradation on human health, including respiratory diseases, waterborne illnesses, and exposure to hazardous substances. Discuss the challenges and opportunities of urbanization, including urban sprawl, infrastructure development, resource consumption patterns, and sustainable urban planning practices.	**3,5**
V	**International agreements and protocols**, National forest policy and Environmental laws and acts. EIA.	**7**	Explain the purpose and significance of international environmental agreements and protocols, such as the Kyoto Protocol, Paris Agreement, Convention on Biological Diversity (CBD), and Montreal Protocol, analyzing their goals, implementation mechanisms, and global impact.	**3,5**

TEXT BOOKS:

T1: Sudhir Andrews (2013) Food and Beverage Service: A Training Manual, Tata McGraw Hill, 2013.

REFERENCE BOOKS:

R1: PradipDey, ManasGhosh, "Programming in C", 2nd Edition, 2018, Oxford University

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping

SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the relationships between natural and man-made systems.	PO1,PO2,PO3,PO4,PO7,PO12
2	Develop critical thinking for shaping strategies (scientific, social, economic, administrative, and legal) for environmental protection, conservation of biodiversity, environmental equity, and sustainable development.	PO1,PO2,PO3,PO4,PO7,PO12
3	Understand the consequences of human actions on the web of life, global economy, and quality of human life.	PO2,PO7,PO12
4	Understand about Environment Pollution, Global Environment issues, Impact of Human Population and modern civilization on the Environment	PO2,PO3,PO7
5	Analyse various aspects of human population, and the impact of the population growth on the environment.	PO1,PO2,PO3,PO4,PO7,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24UBES1201R	Environmental Science	CO1	1	2	2	2			3					3
		CO2	2	2	2	2			3					3
		CO3		3					3					3
		CO4		1	2				3					
		CO5	1	2	2	2			3					3

SEMESTER – II									
Course Title	**Field-based Learning**								
Course code	**24BTCS1204R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 16P	**0**	**0**	**0**	**0**	**0**	**16**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings. 4. To foster collaboration and communication skills through group projects and professional interactions. 5. To cultivate an appreciation for the complexities and challenges of professional practice in the field.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyze and interpret the university data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								
Unit-No.	**Content**		**Contact Hour**		**Learning Outcome**				**BL**
I	Field-Based Learning is designed to provide students with hands-on, practical experience in real-world environments related to Networking. This course emphasizes experiential learning through site visits, fieldwork, projects, and interactions with professionals in the field. Students will apply theoretical knowledge to practice, develop critical thinking skills, and gain a deeper understanding of their discipline.		**16**		Field-Based Learning provides students with hands-on experience in real-world networking environments. It emphasizes experiential learning through site visits, fieldwork, projects, and professional interactions. Students will apply theoretical knowledge, develop critical thinking skills, and gain a deeper understanding of networking.				**1,2,3,4,5**

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS1204R	Field-Based Training	CO1	3				2		1			1		1
		CO2		3		3	2	1						2
		CO3		3	3	2					2		1	
		CO4						1		1	3	3	2	
		CO5						3	3	2			1	2

SEMESTER – II									
Course Title	**PDP (Effective English for Engineers)**								
Course code	**24UBPD1204R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 40T	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**NIL**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. This course will enable the students to acquire the important knowledge on grammar like the formation of sentences. 2. To enable the students to use vocabulary meaningfully for a successful conversation. 3. To establish Reputation and Rapport, a dress code session is much needed 4. The 3 P's (Planning, Prioritizing, and Performing) of Time Management will be taught to the students.								
CO1	Understand and identify common errors in English writing.								
CO2	Acquire skill of report writing.								
CO3	Develop the ability as critical readers and writers.								
CO4	Improve speaking ability in English both in terms of fluency and comprehensibility.								
CO5	Understand the correct usage of English grammar in writing and speaking.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar** Interchange of Interrogative and Assertive Sentences, Interchange of Exclamatory and Assertive Sentences, Analysis of Sentences, Types of Tenses Exercises on Tense	8	Identify the structural differences between interrogative and assertive sentences. Convert assertive sentences into interrogative sentences and vice versa without altering the original meaning. Apply correct punctuation and syntax rules while transforming sentences from one form to another.	1,2,3
II	**Understanding Harmony in the Vocabulary** Synonyms, Antonyms, Homonyms	12	Define and understand what synonyms are and their role in language. Identify and provide examples of synonyms for given words. Use synonyms appropriately in various contexts to avoid repetition and enhance writing and speaking	3,4
III	**Reading Skills** Techniques of Effective Reading, Gathering ideas and information from a text, The SQ3RTechnique Interpret the text	8	Improve the ability to understand and interpret written texts across various genres and disciplines. Expand vocabulary by encountering and learning	4,5

			new words through reading diverse texts.	
IV	**Dress Code Ethics** Introduction to Dress Code Ethics, Purpose and Importance, How to Make first impression, What to Wear During Interviews or Any Other Formal Meetings.	6	Define dress code ethics and explain their relevance in various professional and social settings. Recognize how dress codes can vary across different cultures and the importance of being culturally sensitive in dress choices.	3,5
V	**Time-Management Skills** Introduction To Time Management, Purpose And Importance of Time Management,Basic Tips to Maintain Time.	5	Define time management and explain its key principles. Explore the historical development of time management concepts and techniques.	3,5

TEXT BOOKS:

T1: Wren, P.C and Martin,H. 1995. High School English Grammar and Composition, S Chand Publishing.
T2: Barrett, Grant 2016 Perfect English Grammar: The Indispensable Guide to Excellent Writing and Speaking, Zephyros Press

REFERENCE BOOKS:

R1: Carthy. (2008) English Vocabulary in Use Upper - Intermediate with CD-ROM, Cambridge University Press
R2: Tracy, Brian. (2018) Time Management: The Brian Tracy Success Library, Manjul Publishing House

OTHER LEARNING RESOURCES: NIL

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and identify common errors in English writing.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
2	Acquire skill of report writing.	PO1,PO2,PO3,PO4,PO5,PO10,PO12

3	Develop the ability as critical readers and writers.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
4	Improve speaking ability in English both in terms of fluency and comprehensibility.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
5	Understand the correct usage of English grammar in writing and speaking.	PO1,PO2,PO3,PO4,PO5,PO10,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24UBPD1204R	**PDP (Effective English for Engineers)**	**CO1**	1	1	1	1	1					3		2
		CO2	1	1	1	1	1					3		2
		CO3	1	1	1	1	1					3		2
		CO4	1	1	1	1	1					3		2
		CO5	1	1	1	1	1					3		2

SEMESTER – II									
Course Title	**Extra-Curricular Activities**								
Course code	**24UBEC1201**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. It is to develop the social and soft skills and to promote a holistic development of the learners. 2. Participating in debate clubs, drama, or public speaking events enhances verbal and non-verbal communication skills. 3. Taking leadership roles in student organizations or event planning committees fosters leadership and teamwork abilities.								
CO1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.								
CO2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.								
CO3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.								
CO4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.								
CO5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Discover the exciting world of engineering through hands-on projects and workshops. Gain practical skills in robotics, coding, and innovation, preparing for future STEM success.	30	Explore the dynamic field of engineering through interactive projects and workshops. Develop practical skills in robotics, coding, and innovation. Prepare for future success in STEM disciplines.	1,2,3,4,5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.	PO9
2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.	PO11,PO12
3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.	PO12
4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.	PO10
5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.	PO9

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24UBEC1201	Extra-Curricular Activities	CO1									2			
		CO2											3	2
		CO3												3
		CO4										2		2
		CO5									1			

SEMESTER – III									
Course Title	**Digital Electronics**								
Course code	**24BTCS2101R**	**Total credits: 3** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To familiarize with the different number systems, logic gates, and combinational and sequential circuits utilized in the different digital circuits and systems. 2. The course will help in design and analysis of the digital circuit and system.								
CO1	Understand Boolean algebra, analyze digital logic families, demonstrate IC interfacing.								
CO2	Design logic functions, implement digital circuits, showcase MSI chip expertise								
CO3	Analyse, design sequential circuits, demonstrate flip-flop and counter proficiency								
CO4	Evaluate digital-to-analog converters, understand quantization, A/D converter types.								
CO5	Analyse semiconductor memories, demonstrate proficiency in memory technology and PLDs.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Fundamentals of Digital Systems and logic families:** Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Boolean algebra, examples of IC gates, number systems-binary, signed binary, octal hexadecimal number, binary arithmetic, one's and two's complements arithmetic, codes, error detecting and correcting codes, characteristics of digital ICs, digital logic families, TTL, Schottky TTL and CMOS logic, interfacing CMOS and TTL, Tri-state logic.	**12**	Understand digital signals, circuits, logic gates, Boolean algebra, binary arithmetic, number systems, IC gates, logic families, and interfacing techniques.	**1,2,3**
II	**Combinational Digital Circuits:** Standard representation for logic functions, K-map representation, simplification of logic functions using K-map, minimization of logical functions. Don't care conditions, Multiplexer, De-Multiplexer/Decoders, Adders, Subtractors, BCD arithmetic, carry look ahead adder, serial adder, ALU, elementary ALU design, popular MSI chips, digital comparator, parity checker/generator, code converters,	**10**	Understanding digital logic fundamentals including Boolean algebra, K-maps, multiplexers, adders, ALUs, and MSI chips for designing and optimizing digital circuits.	**1,2,3**

	priority encoders, decoders/drivers for display devices, Q-M method of function realization.			
I	**Sequential circuits and systems**: A 1-bit memory, the circuit properties of Bistable latch, the clocked SR flip flop, J- K-T and D types flip flops, applications of flip flops, shift registers, applications of shift registers, serial to parallel converter, parallel to serial converter, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, counters design using flip flops, special counter IC's, asynchronous sequential counters, applications of counters.	**8**	Mastering digital memory elements (latches, flip-flops), shift registers, counters, and their applications for sequential logic design in digital systems.	**1,2,3,4,6**
IV	**A/D and D/A Converters:** Digital to analog converters: weighted resistor/converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter ICs, sample and hold circuit, analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D	**7**	Gaining expertise in digital to analog and analog to digital converters, encompassing specifications, operational principles, and practical applications in digital systems design.	**1,2,3,4,5**
V	**Semiconductor memories and Programmable logic devices:** Memory organization and operation, expanding memory size, classification and characteristics of memories, sequential memory, read only memory (ROM), read and write memory(RAM), content addressable memory (CAM), charge de coupled device memory (CCD), commonly used memory chips, ROM as a PLD, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).	**8**	Developing comprehensive knowledge of memory organization, types (ROM, RAM, CAM), memory characteristics, and applications in digital systems and programmable logic devices like PLDs and FPGAs.	**1,2,3,4,6**
Practical Component				

Practical 1	To study and verify the truth table of logic gates.	**2**	To effectively study and validate the truth tables of logic gates, ensuring proficiency in understanding their operational principles and applications in digital circuit design.	**2,3**
Practical 2	To design and implementation using NAND gate and NOR gate as Universal Gate.	**2**	Designing and implementing circuits using NAND and NOR gates as universal gates, enhancing versatility in digital logic synthesis and problem-solving.	**3,4,5**
Practical 3	To study about Ex-OR gates and verify their Truth Table.	**2**	Studying XOR gates and verifying their truth table enhances understanding of their unique logic behavior critical for digital circuit design.	**2,3**
Practical 4	To study about adder circuits, half adder circuits and verify their truth table	**2**	Studying adder circuits, including half adders, and verifying their truth tables enhances proficiency in binary arithmetic fundamental to digital logic design.	**2,3**
Practical 5	To study about adder circuits, full adder circuits and verify their truth table.	**2**	Studying full adder circuits and verifying their truth tables strengthens understanding of binary addition, critical for designing complex arithmetic units in digital systems.	**2,3,4**
Practical 6	To study half subtractor using basic gates.	**2**	Studying half subtractors using basic gates enhances understanding of binary subtraction fundamentals, essential for digital logic design and arithmetic operations.	**2,3,4**
Practical 7	To study full subtractor using basic gates.	**2**	Studying full subtractors using basic gates enhances	**2,3,4**

			understanding of binary subtraction with borrow, crucial for designing efficient arithmetic circuits in digital systems.	
Practical 8	To design and set up a 4:1 and 2:1 Multiplexer.	**2**	Designing and setting up 4:1 and 2:1 multiplexer enhances understanding of data selection and routing in digital circuits, critical for signal processing applications.	**2,3,4,5,6**
Practical 9	To design and set up a 1:4 Demultiplexer (DE-MUX).	**2**	Designing and setting up a 1:4 demultiplexer enhances understanding of data distribution and signal routing, crucial for digital communication and control systems.	**2,3,4,5,6**
Practical 10	Implementation and verification of decoder and encoder using logic gates.	**2**	Implementing and verifying decoders and encoders using logic gates enhances understanding of signal decoding and encoding techniques crucial for digital communication and control systems.	**2,3,4,5,6**
Practical 11	Mini-Project			

TEXTBOOKS:

T1: J.B Gupta, Electronic Device and Circuits, 2nd Ed., Katson Books 6th edition 2018.
T2: Dr. Sanjay Sharma, Digital Electronics and Logic Design, 6th Ed., Katson Books 2010.

REFERENCE BOOKS:

R1: Floyd, Electronic Devices, Pearson Education, 2nd edition, 9th edition, 2012.
R2: RP Jain, Modern Digital Electronics, Tata Mc Graw Hill, 3rd Edition, 2007.
R3: Frenzel, Communication Electronics: Principles and Applications, Tata Mc Graw Hill, 3rd Edition, 2001.
R4: Kumar, Fundamentals of Digital Circuits, Prentice Hall India, 2016.

OTHER LEARNING RESOURCES:

O1: NPTEL: Platforms like NPTEL offer courses such as "Digital Circuits and Systems" which cover topics ranging from digital circuits to system design.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand Boolean algebra, analyse digital logic families, demonstrate IC interfacing.	PO1, PO2, PO3, PO10, PO12
2	Design logic functions, implement digital circuits, showcase MSI chip expertise	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Analyse, design sequential circuits, demonstrate flip-flop and counter proficiency	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Evaluate digital-to-analog converters, understand quantization, A/D converter types.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Analyse semiconductor memories, demonstrate proficiency in memory technology and PLDs.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2101R	Digital Electronics	CO1	2	2	1							1		1
		CO2	3	3	2	2	1					1		2
		CO3	3	3	3	1	1					1		2
		CO4	3	3	2	2	1					1		2
		CO5	3	3	2	2	1					1		2

SEMESTER – III									
Course Title	**Functional Programming in Python**								
Course code	**24BTCS2102R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To introduce the core concepts of functional programming and their implementation in Python. 2. To develop proficiency in using Python's functional constructs such as lambda functions, map, filter, and reduce. 3. To enable students to write modular, reusable, and efficient code using functional programming paradigms. 4. To explore the use of recursion, higher-order functions, and immutability in functional programming. 5. To apply functional programming techniques to solve real-world computational problems effectively.								
CO1	Understand about the history of Programming Languages, types of programming languages, Basic Syntax of a few programming languages.								
CO2	Understand the fundamentals of Python programming language and its syntax, enabling them to write basic to intermediate level programs.								
CO3	Demonstrate proficiency in using data types and conditional statements to make simple Python programs and manipulate data effectively.								
CO4	Understand about Loops, functions and various packages in Python.								
CO5	Understand Lambda and user defined functions in Python.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**History of Programming Languages** Overview of Programming Languages, Basic Syntax of a few programming languages, Advantages and disadvantages of Python, Real-life examples of application of Python	**6**	Understanding Python's versatility through its syntax, advantages in ease of use, and real-world applications in diverse fields like data science and automation.	**1,2**
II	**Introduction to Python Programming** Introduction to Python Language, Writing and Compiling Python Programs, Basic Structure of a Python Program, Data Types and Constants	**10**	Developing proficiency in Python basics: syntax, program structure, data types, constants, enabling effective programming and problem-solving skills.	**2,3**
I	**Data types and Variables** Data types, Variable declaration and use, Typecasting in Python, Operators in Python: Assignment, Logical, Arithmetic, Conditional statements: If statement, If else statement, Python	**10**	Mastering Python essentials: data types, variables, typecasting, operators, conditional statements, and for robust programming proficiency.	**1,2,3**

	Collections: List, Tuple, Sets and Dictionary			
IV	**Loops and Functions** Loops in Python: For Loop, While Loop and Nested Loops, User Defined functions	**7**	Proficiency in Python loop and user-defined functions, enabling efficient problem-solving and scalable software development.	**1,2,3**
V	**Lambda Functions** Types of functions, Lambda functions	**7**	Understanding types of functions (built-in, user-defined) and lambda functions in Python for versatile programming and concise code implementation.	**1,2,3**
Practical Component				
Practical 1	**Introduction to Python** Overview of Python Programming ● Setting up Python environment a) Installing Python b) Installing an IDE (e.g., PyCharm, VS Code, Jupyter Notebook) ● Writing your first Python program ● Basic syntax, comments, and documentation	**3**	Develop foundational skills in Python: setup environment, write code, apply basic syntax, comments, and documentation for effective programming.	**1,2,3**
Practical 2	**Basic Data Types and Variables** ● Variables and data types ○ Integers, floats, strings, and Booleans ● Type casting ● Basic input and output	**3**	Mastering Python basics: variables (integers, floats, strings, booleans), type casting, and basic input/output operations for practical programming proficiency.	**1,2,3**
Practical 3	**Operators and Expressions** ● Arithmetic operators ● Comparison operators ● Logical operators ● Assignment operators ● Bitwise operators	**3**	Understanding Python operators: arithmetic, comparison, logical, assignment, and bitwise, essential for comprehensive programming and problem-solving in diverse applications.	**3,4**
Practical 4	**Control Structures**	**3**	Mastering control structures in Python:	**1,2,3**

	• Conditional statements a) if, else if, else • Loops a) for loops b) while loops • Break and continue statements		conditional statements, loops, break, and continue statements for efficient program flow and logic implementation.	
Practical 5	**Functions** • Defining functions • Calling functions • Parameters and arguments • Return values • Scope of variables • Lambda functions	**3**	Proficiency in Python functions: defining, calling, handling parameters/arguments, return values, variable scope, and utilizing lambda functions for concise and efficient code implementation.	**1,2,3**
Practical 6	**Data Structures** • Lists a) Creating and accessing elements b) List methods c) List comprehensions • Tuples • Dictionaries a) Creating and accessing elements b) Dictionary methods Sets	**3**	Proficiency in Python data structures: lists, tuples, dictionaries, and sets.	**1,2,3**
Practical 7	**Strings** • String operations • String methods • String formatting • Regular expressions	**3**	Mastering string operations, methods, formatting, and regular expressions in Python for robust text processing and pattern matching capabilities.	**1,2,3**
Practical 8	**File Handling** • Opening and closing files • Reading and writing files • Working with file modes • Using with statement	**3**	Proficiency in file handling in Python: opening, closing, reading, writing files, understanding file modes, and utilizing the with statement for resource management.	**1,2,3**
Practical 9	**Exception Handling** • Understanding exceptions • try, except, finally blocks • Handling multiple exceptions • Custom exceptions	**3**	Mastering exception handling in Python: understanding exceptions, using try-except-finally blocks, managing multiple exceptions, and creating custom exceptions for robust	**1,2,3**

			error management in programs.	
Practical 10	**Modules and Packages** • Importing modules • Creating modules • Using standard libraries	**3**	Proficiency in module management in Python: importing modules, creating custom modules, and utilizing standard libraries like os, sys, math, datetime for enhanced functionality in applications.	**1,2,3**

TEXT BOOKS:

T1: BALAGURUSAMY, 1st Edition, Introduction To Computing And Problem-Solving Using Python, McGraw Hill, ISBN 9789352602582

REFERENCE BOOKS:

R1: Andrew Ngo, Introduction to Python Programming: Beginner to Advanced, Practical Guide, Tips and Tricks, Easy and Comprehensive, 1st Edition, 2017, Kindle Books

R2: Venugopal and Prasad, Python: The Complete Reference, 4th Edition, 2018, Tata McGraw Hill.

OTHER LEARNING RESOURCES:

O1: youtube: The youtube tutorial "Python Tutorial for Beginners" offers python programming tutorials for beginners, including functional programming principles.

O2: geeksforgeeks: Offers tutorials, including functional programming principles.

O3: Coursera: the course "Programming for Everybody (Getting Started with Python)" offered helps Gain a foundational understanding and learn new concepts from industry experts

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand about the history of Programming Languages, types of programming languages, Basic Syntax of a few programming languages.	PO1, PO2, PO3, PO5, PO11
2	Understand the fundamentals of Python programming language and its syntax, enabling them to write basic to intermediate level programs.	PO1, PO2, PO3, PO5, PO11
3	Demonstrate proficiency in using data types and conditional statements to make simple Python programs and manipulate data effectively.	PO1, PO2, PO3, PO5, PO11
4	Understand about Loops, functions and various packages in Python.	PO1, PO2, PO3, PO5, PO11
5	Understand Lambda and user defined functions in Python.	PO1, PO2, PO3, PO5, PO11

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2102R	Functional Programming in Python	CO1	3	1	1		2						1	
		CO2	1	2	2		2						1	
		CO3	1	1	1		2						1	
		CO4	1	2	1		2						1	
		CO5	2	1	1		2						1	

SEMESTER – III									
Course Title	**Discrete Mathematics and Graph Theory**								
Course code	**24BTCS2103R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To provide students with a solid understanding of the fundamental concepts in discrete mathematics and graph theory. 2. To develop students' ability to solve problems related to discrete structures and graph algorithms. 3. To enable students to apply discrete mathematical concepts and graph theory techniques to computer science and related fields.								
CO1	Understand set operations, analyse relations, and demonstrate problem-solving in mathematics.								
CO2	Evaluate propositional logic, demonstrating analytical reasoning and deduction skills.								
CO3	Analyse algebraic structures, showcasing proficiency in groups, rings, and fields.								
CO4	Apply counting techniques, demonstrate problem-solving capabilities in mathematics.								
CO5	Analyse graph properties, apply graph theory concepts effectively in problem-solving.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Foundations of Discrete Mathematics** • Set Theory: Sets, subsets, operations on sets, Venn diagrams, Cartesian products. • Logic: Propositional logic, predicates, quantifiers, logical connectives, truth tables. • Proof Techniques: Direct proof, proof by contradiction, induction, contrapositive.	**4**	In Foundations of Discrete Mathematics, students will understand the basics of set theory, including sets, subsets, operations, Venn diagrams, and Cartesian products. They will learn propositional logic, predicates, quantifiers, logical connectives, and truth tables, along with mastering proof techniques such as direct proof, proof by contradiction, induction, and contrapositive.	**1,2,3**
II	**Combinatorics and Discrete Probability** • Combinatorics: Permutations, combinations, Pigeonhole principle, inclusion-exclusion principle. • Discrete Probability: Probability theory, conditional probability, Bayes' theorem, expected value.	**8**	Students will understand permutations, combinations, the Pigeonhole principle, and the inclusion-exclusion principle. They will also learn probability theory, conditional probability, Bayes' theorem, and how to calculate the expected value.	**2,5**

I	**Graph Theory Basics** • Graphs: Definitions, types of graphs, subgraphs, isomorphism. • Graph Representation: Adjacency matrix, adjacency list, incidence matrix. Trees: Properties, spanning trees, minimum spanning trees.	**10**	Students will learn the definitions and types of graphs, subgraphs, and graph isomorphism. They will understand graph representation using adjacency matrices, adjacency lists, and incidence matrices, and explore tree properties, spanning trees, and minimum spanning trees.	**1,2,3,4**
IV	**Advanced Graph Theory** • Graph Algorithms: Breadth-first search (BFS), depth-first search (DFS), Dijkstra's algorithm, Floyd-Warshall's algorithm. • Planarity and Colouring: Planar graphs, Euler's formula, graph colouring, chromatic number. Network Flows: Max-flow min-cut theorem, Ford-Fulkerson algorithm.	**7**	In Advanced Graph Theory, students will master graph algorithms such as BFS, DFS, Dijkstra's, and Floyd-Warshall. They will understand planar graphs, Euler's formula, graph colouring, chromatic number, and network flows including the max-flow min-cut theorem and the Ford-Fulkerson algorithm.	**2,4,6**
V	**Applications of Discrete Mathematics** • Applications in Computer Science: Data structures, algorithms, complexity. • Applications in Network Theory: Internet graphs, social networks, communication networks. • Additional Applications: Cryptography, coding theory, combinatorial optimization.	**7**	In Applications of Discrete Mathematics, students will explore its use in computer science for data structures, algorithms, and complexity, and in network theory for internet graphs, social networks, and communication networks. They will also understand applications in cryptography, coding theory, and combinatorial optimization.	**2,3**

TEXT BOOKS:

T1: Discrete Mathematics and Its Applications" by Kenneth H. Rosen

REFERENCE BOOKS:

R1: "Introduction to Graph Theory" by Douglas B. West.

OTHER LEARNING RESOURCES:

O1*:* *Discrete Mathematics for Computer Science Specialization* on Coursera: Offered by UC San Diego and National Research University Higher School of Economics.

O2: MIT Open Course Ware's lecture notes on Discrete Mathematics and Graph Theory.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand set operations, analyse relations, and demonstrate problem-solving in mathematics.	PO1, PO2,PO12
2	Evaluate propositional logic, demonstrating analytical reasoning and deduction skills.	PO1, PO2,PO12
3	Analyse algebraic structures, showcasing proficiency in groups, rings, and fields.	PO1, PO2,PO12
4	Apply counting techniques, demonstrate problem-solving capabilities in mathematics.	PO1, PO2,PO12
5	Analyse graph properties, apply graph theory concepts effectively in problem-solving.	PO1, PO2,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS2103R	**Discrete Mathematics and Graph Theory**	**CO1**	1	3										1
		CO2	2	3										2
		CO3	2	3										2
		CO4		3										2
		CO5	2	3										2

SEMESTER – III									
Course Title	**Computer Organization and Architecture**								
Course code	**24BTCS2104R**	**Total credits: 4** **Total hours: 45T + 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Programming For Problem Solving	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. Discuss the organization and architecture of computer systems and electronic computers. 2. Analyse the basic components of computer systems besides the computer arithmetic. 3. Discuss input-output organization, memory organization and management, and pipelining.								
CO1	Understand the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes.								
CO2	Analyse the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers.								
CO3	Analyse the control unit design approaches, memory design technologies and I/O transfers.								
CO4	Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.								
CO5	Summarize the concepts of memory organization with mapping functions and replacement algorithms.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Functional blocks of a computer:** CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU – registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs.	**8**	Understanding computer architecture components: CPU, memory, I/O subsystems, control unit. Mastering CPU's instruction set architecture, addressing modes, and interpreting instructions for diverse applications.	**1,2,3**
II	**Data representation:** signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic. Introduction to x86 architecture.	**8**	Understanding data representation (signed numbers, fixed/floating-point, character encoding) and computer arithmetic (addition, multiplication, division techniques, floating-point operations) for comprehensive computer system understanding.	**1,2,3**
I	**CPU control unit design:** hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU. Memory system design: semiconductor memory	**10**	Understanding CPU control unit design (hardwired vs micro-programmed), memory system architecture, I/O subsystems, interfaces,	**1,2**

	technologies, memory organization. Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB		interrupts, and exceptions for comprehensive computer system understanding and design.	
IV	**Pipelining:** Basic concepts of pipelining, throughput and speedup, pipeline hazards. Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.	**8**	Mastering pipelining fundamentals: concepts, throughput, speedup, and handling pipeline hazards. Understanding parallel processors, concurrent memory access, and cache coherency for advanced computational efficiency and performance.	**1,2**
V	**Memory organization:** Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.	**8**	Understanding memory organization: interleaving, hierarchical structure, cache memory, mapping techniques, replacement algorithms, and write policies for optimized memory access and performance in computer systems.	**1,2,3**

Practical Component				
Practical 1	**Exploring CPU Components:** Simulate the functional blocks of a CPU (ALU, registers, and control unit) using a simulator.	**3**	Understand the role and functionality of various CPU components. Analyse how instructions are executed within a CPU.	**1,2,4**
Practical 2	**Instruction Set Simulation:** Implement a subset of a CPU's instruction set (e.g., arithmetic and logical instructions) using a simulated environment.	**3**	Identify and explain the operation of basic instructions in a CPU. Simulate and interpret the results of instruction execution cycles.	3,4

Practical 3	**Data Representation and Conversion:** Write a program to convert between signed number representations (e.g., two's complement) and perform fixed and floating-point arithmetic.	**3**	Interpret different number systems and their representations. Solve arithmetic operations involving fixed and floating-point numbers.	**3,4**
Practical 4	**Designing Adders:** Design and simulate a ripple carry adder and a carry look-ahead adder using a hardware description language (HDL) like Verilog or VHDL.	**3**	Understand the concept of integer addition and analyse the performance of different adder designs.	**2,3**
Practical 5	**Hardwired vs. Microprogrammed Control Unit Design:** Design and compare a hardwired control unit and a microprogrammed control unit for a simple hypothetical CPU.	**3**	Understand the differences between hardwired and microprogrammed control units. Analyse the trade-offs in terms of speed and flexibility.	**4,5**
Practical 6	**Memory Hierarchy Simulation:** Simulate the organization of hierarchical memory (cache, main memory) and analyse the impact of cache size, block size, and replacement policies.	**3**	Proficiency in Python data structures: lists, tuples, dictionaries, and sets.	**1,2,3**
Practical 7	**Booth's Multiplier Implementation:** Implement Booth's multiplication algorithm in a programming language like Python or C++.	**3**	Understand and apply Booth's algorithm for efficient integer multiplication. Evaluate the advantages of Booth's method over basic shift-and-add techniques.	**3,5**
Practical 8	**Pipelining Simulation:** Simulate a basic instruction pipeline using a programming language or simulation tool to understand pipeline hazards (data, control, structural).	**3**	Understand pipelining concepts, hazards, and their impact on performance. Analyse how hazards affect throughput and suggest techniques to mitigate them.	**2,4**
Practical 9	**Exploring x86 Architecture:** Use an x86 assembler (e.g., NASM) to write and execute simple programs	**3**	Explore the x86 instruction set and its addressing modes.	**1,2,3**

	demonstrating arithmetic, logical, and control flow instructions.		Develop basic assembly language programs.	
Practical 10	**I/O Device Interface:** Simulate an I/O transfer mechanism using program-controlled, interrupt-driven, and DMA modes, and analyse their performance.	**3**	Understand the differences between program-controlled, interrupt-driven, and DMA-based I/O transfers. Evaluate the efficiency of different I/O modes.	**2,5**

TEXT BOOKS:

T1: David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", 5th Edition, Elsevier.

REFERENCE BOOKS:

R1: "Computer Organization and Embedded Systems", 6th Edition by CarlHamacher, McGraw Hill Higher Education.
R2: "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill.
R3: "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

OTHER LEARNING RESOURCES:

O1: javatpoint: the tutorial "Computer Architecture and Organization"
O2: NPTEL: "**Computer Organization and Architecture"** by Prof. Mainak Chaudhuri (IIT Kanpur)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes.	PO1, PO2, PO3, PO10, PO12
2	Analyse the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Analyse the control unit design approaches, memory design technologies and I/O transfers.	PO1, PO2, PO3, PO10, PO12
4	Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.	PO1, PO2, PO3, PO4, PO10, PO12
5	Summarize the concepts of memory organization with mapping functions and replacement algorithms.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2104R	Computer Organization and Architecture	CO1	2	3	1							1		3
		CO2	3	3	3	2	2					1		3
		CO3	2	2	2							1		3
		CO4	2	2	3	2						1		3
		CO5	2	2	2	2	1					1		3

SEMESTER – III									
Course Title	**Field-Based Learning**								
Course code	**24BTCS2105R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 16	**0**	**0**	**0**	**0**	**0**	**16**	**1**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings. 4. To foster collaboration and communication skills through group projects and professional interactions. 5. To cultivate an appreciation for the complexities and challenges of professional practice in the field.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyse and interpret the university data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	The organization will provide the students one field visit for this program. The students will be taken to a specific blocks of the university for learning purposes. The students will visit all department block site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners of IT department of the university. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.	16	Enhanced understanding through immersive field visits, interaction with IT professionals, and reflective journaling to integrate theoretical knowledge with practical insights effectively.	1,2,3,4,5,6

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.	PO1, PO5, PO7, PO10, PO12
2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.	PO2, PO4, PO5, PO6, PO12
3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.	PO2, PO3, PO4, PO9, PO11
4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.	PO6, PO8, PO9, PO10, PO11
5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.	PO6, PO7, PO8, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS2105R	**Field-Based Training**	CO1	3				2		1			1		1
		CO2		3		3	2	1						2
		CO3		3	3	2					2		1	
		CO4						1		1	3	3	2	
		CO5						3	3	2			1	2

SEMESTER – III									
Course Title	**PDP (English For Employability For Engineers)**								
Course code	**24UBPD2103R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To introduce the types of sentences and their significance. 2. To strengthen the vocabulary of the students to enhance student' vocabulary to enhance their speaking and writing skills it the importance of dress codes in various organizations. 3. To overcome fear of Public Speaking 4. To understand the process of Effective Reading techniques and Listening skills.								
CO1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.								
CO2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.								
CO3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.								
CO4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.								
CO5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Reading Skills:** Exercise 1: Interpreting different texts Exercise 2: Importance and significance of Punctuation Exercise 3: Story -reading Exercise 4: News-reading Exercise 5: Comprehending texts of different genres	**6**	Enhancing reading skills: interpreting texts, understanding punctuation, reading stories and news, and comprehending various genres for improved literacy.	**1,2,3**
II	**Non-Verbal Communication:** i. Understanding Non-Verbal Communication and Body Language Types of Body Language ii. Importance and Impact of Body Language and introduction to Haptics, Kinesics and Proxemics	**6**	Understanding non-verbal communication: types and impact of body language, and introduction to haptics, kinesics, and proxemics for effective interaction.	**1,2,3**
I	**Grammar (Flipped Classroom):** i. Types of Sentences (using Assertive, Imperative, exclamatory in respective contexts), ii. Degrees of Comparison	**4**	Understanding grammar through flipped classroom: identifying types of sentences (assertive, imperative, exclamatory) and mastering degrees of comparison in context.	**2,3**
IV	**Public Speaking Skills:** i. Introduction to public speaking	**7**	Developing public speaking skills: preparation (scripts, non-verbal cues), overcoming	**2,3**

	Preparation for Public speaking (scripts, non-verbal cues) ii. Understanding and overcoming Fear of Public Speaking Tips Public Speaking		fear, and applying effective public speaking tips for confident presentations.	
V	**Listening Skills:** i. Process of listening ii. Understanding listening barriers I. Difference between Listening and Hearing, iv. Importance of Effective Listening Practice Session: Listening to podcasts, Lectures, Audio Books etc.	**5**	Enhancing listening skills: understanding the listening process, identifying barriers, distinguishing listening from hearing, and practicing with podcasts, lectures, and audiobooks.	**2,3,4**

TEXT BOOKS:

T1: Wren,P.C and Martin,H. 1995. High School English Grammar and Composition, S Chand Publishing.

T2: Barrett,Grant.2016.PerfectEnglishGrammar:TheIndispensibleGuidetoExcellent Writing and Speaking, Zephyros Press

REFERENCE BOOKS:

R1: Carthy. (2008) English Vocabulary in Use Upper - Intermediate with CD-ROM, Cambridge University Press

R2: Tracy, Brian. (2018) Time Management: The Brian Tracy Success Library, Manjul Publishing House

OTHER LEARNING RESOURCES:

O1: NPTEL- the course "Technical English for Engineers" by Prof. Aysha Iqbal, IIT Madras

O2: youtube- "Your body language may shape who you are" Amy Cuddy ,TED

O3: Coursera- "Training and Practicing in English Public Speaking" by Shanghai Jiao Tong University

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.	PO1, PO5, PO9, PO10
2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.	PO6, PO9, PO10, PO12
3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.	PO6, PO9, PO10, PO11

4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.	PO5, PO10, PO12
5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.	PO6, PO8, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24UBPD2103R	English For Employability for Engineers	CO1	1				1				1	3		
		CO2						2			1	3		1
		CO3						2			3	2	1	
		CO4					1					2		3
		CO5						3		1		3		2

SEMESTER – III									
Course Title	**Financial Literacy**								
Course code	**24UBPD2104R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	Basic Financial knowledge	**Co-requisite**	**NIL**						
Programme	**Bachelor of Technology and Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. The course would offer an inclusive approach to understand the relevant concepts of money, borrowing, lending, taxes and their application to financial planning. 2. Assess the personal financial planning process, the life cycle of financial plans, and methods of goal achievement. 3. Formulate a budget, record-keeping system, and tax planning strategy based on current financial goals.								
CO1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.								
CO2	Design a diversified investment portfolio that addresses several different investment objectives.								
CO3	Differentiate between open and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.								
CO4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.								
CO5	Apply financial knowledge and skills to make informed decisions, ensuring long-term financial stability and security.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Fundamentals of Financial Planning - i. Functions of money; ii. Inflation- Meaning, causes, how it can be controlled; I.process official planning, iv. Time value of money-simple and compound interest; v.Net Present Value and Future value, vi. Power of Compounding; vii. Doubling period and Rule of 72.	**8**	Fundamentals of Financial Planning covers understanding money's functions, inflation causes and control, financial planning processes, time value of money including simple and compound interest, Net Present Value (NPV) and Future Value (FV), compounding power, doubling periods, and the Rule of 72 for effective financial decision-making.	**1,2, 3**
II	Income Tax Planning- i. Meaning of Income, ii. Direct and Indirect Taxes, Taxable Income, various heads of Income for tax Calculation, I. Non-taxable Income, iv. Tax evasion and tax avoidance, v.GST, Tax Planning Strategies.	**8**	Learning Outcome: Gain proficiency in understanding income, taxes (direct, indirect), taxable and non-taxable income, tax evasion, avoidance, GST, and effective tax planning strategies.	**3,4**
I	Entrepreneurial planning - i. Meaning of Entrepreneurship, prerequisites for becoming an entrepreneur, ii. Entrepreneurship Support Systems in India,	**10**	Understand entrepreneurship fundamentals, support systems in India, institutional and financial support mechanisms (venture capital, business angels, government	**3,4**

	I. Institutional support systems for entrepreneurs, iv. Financial support systems for entrepreneurs; v. Venture Capital, Business Angels, vi. Assistant of Government, vii. Commercial Bank Loans and Overdraft.		assistance, commercial bank loans), enabling effective entrepreneurial planning and implementation.	
IV	Planning for investing in securities market - i. Investment avenues offered by Securities Markets.. Primary Market and Secondary Market, ii . Stock market- meaning, features, functions of NSE, BSE DEMAT trading account, I. Security repository, stock brokers. Operational aspects of securities markets: placement of orders, contract note, pay-in and pay-out, trading and settlement cycle, iv. Various risks involved in investing in securities markets; Role of Financial Intermediaries; Stock indices. v. Mutual Funds- meaning concept, definition, types, importance and drawbacks of mutual funds, mutual funds in India, investing in mutual funds, vi. Systematic Investment Plan (SIP) and its advantages.	**10**	Gain knowledge of investment avenues in securities markets, primary and secondary markets, stock market functions (NSE, BSE), DEMAT trading, security repositories, stock brokers, operational aspects (order placement, contract notes, trading cycles), risks, financial intermediaries, stock indices, mutual funds (types, importance, drawbacks, investing in India), and advantages of Systematic Investment Plans (SIPs).	**4**
V	Planning for debts and Retirement i. Consumer credit Introduction to consumer credit; choosing a source of credit, the cost of credit alternatives, ii. Consumer Legal Protection; I. Housing Decision: Factors and Finance: Vehicle Decisions. iv. Retirement planning Meaning of cost of living; retirement need analysis; development of retirement plan, various retirement schemes. v. Estate Planning; Pension and Medicare Planning; Wills.	**9**	Develop skills in consumer credit management, including choosing credit sources and evaluating costs. Understand consumer legal protections and factors influencing housing and vehicle decisions. Master retirement planning, including cost-of-living considerations, retirement needs analysis, retirement scheme evaluation, estate planning, pension, Medicare planning, and wills.	**4**

TEXT BOOKS:

T1: Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems andThe Million-Dollar Financial Advisor: Powerful Lessons and Proven Strategies from Top Producers by David J. Mullen Jr
T2: Personal Finance and Planning by Dr. Rajni
T3: Peaceful Personal Finance: A Short Read on the Basics of Personal Finance and Planning Kindle Edition by Hema Singh

REFERENCE BOOKS:

R1: Be Your Own Financial Advisor: Financial Planning, Investment Options, Risk Management, Tax Management, Succession Planning Kindle Edition y Sushil Bali
R2: The Dumb Things Smart People Do with Their Money: Thirteen Ways to Right Your Financial Wrongs Kindle Edition y Jill Schlesinger

OTHER LEARNING RESOURCES:

O1: Financial Planning Association (FPA):Offers resources, workshops, and events for financial planners and individuals seeking financial advice.
O2: Local Community Resources: Check local libraries, community centers, and universities for workshops or seminars on personal financial planning.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	PO2, PO9, PO10, PO11
2	Design a diversified investment portfolio that addresses several different investment objectives.	PO2, PO5, PO11, PO12
3	Differentiate between open and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.	PO1, PO2, PO6, PO7
4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.	PO2, PO6, PO8, PO11, PO12
5	Apply financial knowledge and skills to make informed decisions, ensuring long-term financial stability and security.	PO2, PO6, PO8, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24UBPD2104R	**Financial Literacy**	**CO1**		2							1	1	3	
		CO2		3			2						3	1
		CO3	2	2				1	1					
		CO4		2				1		1			3	2
		CO5												

SEMESTER – III									
Course Title	**Basic Life Saving Skills**								
Course code	**24UULS2102R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To equip students with essential knowledge of emergency response protocols. 2. To develop skills for administering CPR and first aid in critical situations. 3. To familiarize students with the use of automated external defibrillators 4. To train students in recognizing and responding to medical emergencies effectively. 5. To promote confidence and preparedness in handling life-threatening scenarios.								
CO1	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives								
CO2	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life								
CO3	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders, and anyone involved in emergency care								
CO4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate ,delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.								
CO5	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction(heart attack),stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Basic Life Support(BLS)** Introduction of BLS Chain of survival ABCs Assessment CPR and Ventilation Technique AED Choking for adult and children	5	Understanding chain of survival, ABCs assessment, CPR techniques, ventilation, AED usage, and choking management for adults and children.	**1,2,3**
II	**Soft skills** ● Introduction ● Communications Skills ● Situational Skills ● Team WorK	5	Enhanced communication abilities, improved situational awareness, strengthened teamwork skills, and mastery of other interpersonal competencies crucial for professional success.	**3,4**

I	**Trauma emergencies** • Scene safety • Primary assessment • Bleeding control • Helmet removal • Care of amputated body part • Extrication of victims and safe transfer • Cervical spine stabilizatio n • Cervical collar application • Splinting of broken Limbs	10	Mastering scene safety assessment, prioritizing initial approaches in pre-hospital care such as primary assessment and bleeding control, and executing procedures like helmet removal, care of amputated body parts, extrication, cervical spine stabilization and collar application, and splinting of broken limbs for effective patient management and safety.	**3,4**
IV	**Triage system** • Introduction • Flow chart approach of Triage • Triage of Multiple Casualties in Pre-Hospital setting • Triage of Single casualty	5	Understanding the introduction to triage, mastering the flowchart approach for triage, effectively triaging multiple casualties in a pre-hospital setting, and appropriately triaging a single casualty to optimize emergency response and patient care outcomes.	**1,2,3,4**
V	**Medical emergencies** • Introduction • Victim centred approach in medical emergency • Diabetic emergencies • emergency child birth • stroke recovery position	5	Learning victim-centered medical emergency approaches, managing seizures, heart attacks, asthma, diabetic emergencies, emergency childbirth, and stroke recovery positioning for effective patient care.	**2,3,4**

TEXT BOOKS:

T1: Nancy Caroline, Emergency Care in the streets, Seventh edition by Jones and Bartlett

T2: LC Gupta, First Aid book

REFERENCE BOOKS:

R1: Advance Cardio vascular life support and Basic life support provider manual, American Heart Association (AHA)

R2: Wikipedia - Basic life support

OTHER LEARNING RESOURCES:

O1: youtube- "Basic Life Support (BLS) CPR and AED Training" by the American Heart Association

O2: **youtube**- **"First Aid Training"** by St John Ambulance
O3: **Coursera**- **"Introduction to First Aid"** by the University of Michigan
O4: **Udemy**- **"Complete BLS Certification Course - Basic Life Support"** by First Aid For Free

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives	PO1, PO6, PO8, PO10, PO12
2	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life	PO6, PO9, PO10, PO12
3	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders, and anyone involved in emergency care	PO2, PO4, PO10, PO12
4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate ,delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.	PO2, PO6, PO10, PO12
5	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction(heart attack),stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.	PO1, PO2, PO6, PO8, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
23UULS211R	Basic Life Saving Skills	CO 1	2					3		1		2		1
		CO 2						1			3	3		1
		CO 3		3		3						2		2
		CO 4		3				3				2		2
		CO 5	2	3				3		1				2

SEMESTER III										
Course Title	**Logical Reasoning for Computer Science**									
Course code	**24BTCS2107R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the Second year of the Programme**								
Course Objectives	1. To introduce the fundamental concepts of logical reasoning and its importance in problem-solving for computer science. 2. To develop mathematical, analytical, and abstract reasoning skills for computational thinking. 3. To enable students to analyze critical thinking scenarios and evaluate solutions for data sufficiency problems. 4. To enhance students' ability to recognize patterns and relationships for effective decision-making. 5. To apply logical reasoning techniques in programming, debugging, and optimization of algorithms.									
CO1	Understand the foundational concepts of logical reasoning and their applications in computer science.									
CO2	Apply mathematical and analytical reasoning skills to solve complex problems.									
CO3	Analyze and evaluate critical thinking problems and data sufficiency scenarios.									
CO4	Recognize and utilize abstract reasoning and pattern recognition for logical problem-solving.									
CO5	Integrate logical reasoning techniques in programming and algorithm optimization for real-world applications.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to logical reasoning, types, and applications in computer science. Number Series, Alphabet Series, Coding-Decoding, and Direction Sense problems. Logical Deductions: Syllogisms, Statements, and Assumptions.	**8**	Understand the fundamentals of logical reasoning and apply them to basic problems.	**1,2,3**
II	Blood Relations, Seating Arrangements (Linear and Circular), Puzzles. Basic Arithmetic Problems: Percentage, Ratio and Proportion, Profit and Loss. Analytical Reasoning: Ranking, Order, and Arrangement Problems.	**12**	Apply mathematical and analytical reasoning skills to solve computational problems.	**3,4**

I	Data Sufficiency Problems: Understanding assumptions and evaluating statements. Critical Thinking: Logical Cause and Effect, Evaluating Arguments, Identifying Summeracies. Logical Connectives: AND, OR, NOT, and implications in real-world applications.	**12**	Analyze data sufficiency scenarios and solve critical thinking problems for decision-making.	**3,4,5**
IV	Pattern Completion, Figure Series, Mirror and Water Images, and Shape Analogy. Logical Problems Involving Matrices, Odd One Out, and Logical Grouping of Objects. Identifying Relationships in Structured and Unstructured Data.	**12**	Develop abstract reasoning skills to identify patterns and relationships in problem-solving.	**3,4**
V	Problem Solving with Algorithms: Understanding Flowcharts, Decision Trees, and Logical Conditions in Programming. Real-World Scenarios: Scheduling, Network Flow, Resource Allocation, and Optimization Problems. Case Study: Application of logical reasoning in debugging and optimizing computational problems.	**12**	Apply logical reasoning techniques in programming, algorithms, and problem-solving for computer science.	**4,5**

Textbooks

T1: Aggarwal, R. S. - "A Modern Approach to Logical Reasoning", S. Chand Publishing, 2021.

Reference Books

R1: Sharma, A. - "Logical Reasoning and Data Interpretation for CAT", McGraw-Hill, 2022.
R2: Skiena, S. S. - "The Algorithm Design Manual", Springer, 3rd Edition, 2020.

Other Learning Resources

O1: Online Platforms:
- Brilliant.org: Interactive modules on logical reasoning and problem-solving.
- GeeksforGeeks: Logical reasoning for competitive programming and interviews.

O2: Video Tutorials:
- NPTEL: "Problem Solving and Programming" series by IITs.
- YouTube Channels: "Unacademy Logical Reasoning" for practical applications.

O3: Practice Tools:
- LeetCode: Logical reasoning problems in coding.
- HackerRank: Challenges on data sufficiency and pattern recognition.

SEMESTER – III									
Course Title	**Mini Project-I**								
Course code	**24BTCS2106R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 48	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To provide hands-on experience in identifying, analyzing, and solving real-world problems using the knowledge acquired during the course. 2. To enhance project planning, execution, and time management skills through teamwork and collaboration. 3. To enable students to apply theoretical concepts and technical skills to develop a functional solution or prototype. 4. To improve research, critical thinking, and problem-solving skills in a practical setting. 5. To strengthen communication and documentation skills by preparing project reports and presenting project outcomes effectively.								
CO1	Apply IT principles for real-world problem-solving using programming languages and network solutions.								
CO2	Contribute to complex projects as a team member, demonstrating effective personal and team management.								
CO3	Develop programming skills to create and assess software, hardware, and network solutions.								
CO4	Apply personal and team management skills as a professional software developer.								
CO5	Cultivate employability skills and uphold professionalism in software development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a) theoretical studies b) computer simulations c) hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;	**48**	Learning about independent research, technical expertise application, and effective communication through comprehensive project execution and presentation in specialized technical areas.	**1,2, 3,4, 5,6**

	4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BTCS2106R	Mini Project I	CO1	2	2	2	1	2	2				2		2
		CO2	2	2	2	1	2	3				2		2
		CO3	1	1	2	1	2	2				3		3
		CO4	2	2	2	1	2	2				2		2
		CO5	1	1	1	1	2	1				1		3

SEMESTER – IV									
Course Title	**Database Management Systems**								
Course code	**24BTCS2201R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	File Systems	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To understand and use data manipulation language to query, update, and manage a database. 2. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency 3. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.								
CO1	Understand the fundamental concepts of database management systems with DBMS languages and data models.								
CO2	Apply the concepts of query languages such as DDL and DML for designing a relational database.								
CO3	Apply the ACID properties and concurrency control schemes to perform transaction processing with database recovery.								
CO4	Apply security to the database by checking for authentication and authorization								
CO5	Apply advanced topics of data warehousing and data mining, distributed databases and web databases.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Database system architecture:** Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). **Data models:** Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.	**6**	In studying Database System Architecture, students will understand data abstraction, data independence, and the roles of DDL and DML. They will also learn various data models, including entity-relationship, network, relational, and object-oriented models, and comprehend integrity constraints and data manipulation operations.	**1,2**
II	**Relational query languages:** Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. **Relational database design:** Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. **Query processing and optimization:** Evaluation of relational algebra expressions,	**10**	Students will gain proficiency in relational query languages (relational algebra, tuple and domain relational calculus, SQL3), understand DDL and DML constructs across MySQL, Oracle, DB2, and SQL Server. They will also learn effective relational database design (domain and data dependency, Armstrong's axioms, normal forms,	**2,3,4**

	Query equivalence, Join strategies, Query optimization algorithms.		dependency preservation, lossless design) and optimize query processing through evaluation, equivalence, join strategies, and algorithms.	
III	**Transaction processing:** Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Multi-version and optimistic, Concurrency Control schemes, Database recovery.	**8**	Students will comprehend transaction processing concepts such as concurrency control, ACID properties, serializability of scheduling, and locking mechanisms in timestamp-based and multi-version concurrency control schemes. They will also master database recovery techniques essential for maintaining data integrity and availability.	**2,3,4,5**
IV	**Database Security:** Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.	**6**	Students will understand database security fundamentals including authentication, authorization, DAC, MAC, and RBAC models, as well as techniques for intrusion detection and prevention of SQL injection attacks.	**2,3**
V	**Advanced topics:** Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.	**5**	Students will explore advanced topics including object-oriented and object-relational databases, logical databases, web databases, distributed databases, and data warehousing, alongside principles and techniques of data mining for comprehensive data management solutions.	**1,2,3**
Practical Component				
Practical 1	Designing and implementing a relational database schema based on business requirements, ensuring normalization up to 3NF.	**2**	Students will achieve proficiency in designing and implementing relational database schemas tailored to business requirements, ensuring data	2,3

			organization adheres to normalization principles up to the third normal form (3NF).	
Practical 2	Writing and executing SQL queries to retrieve, update, and delete data from a relational database, using DML commands such as SELECT, INSERT, UPDATE, DELETE.	**2**	Students will master the skill of writing and executing SQL queries to efficiently retrieve, update, and delete data from relational databases, employing DML commands (SELECT, INSERT, UPDATE, DELETE) to manipulate and manage data effectively.	3,5
Practical 3	Implementing concurrency control mechanisms like locking and timestamp-based protocols to manage simultaneous access to data in multi-user environments.	**2**	Students will develop the capability to implement concurrency control mechanisms such as locking and timestamp-based protocols, ensuring consistent and reliable management of data access in multi-user environments to prevent conflicts and maintain data integrity.	3,4
Practical 4	Performing query optimization by analyzing query plans, selecting appropriate join strategies, and applying optimization algorithms (e.g., cost-based optimization).	**2**	Students will learn to optimize query performance by analyzing query plans, selecting optimal join strategies, and applying algorithms such as cost-based optimization, enhancing database efficiency and responsiveness.	4,5
Practical 5	Securing database access through authentication mechanisms and defining authorization roles using RBAC models to enforce access controls.	**2**	Students will gain proficiency in securing database access through authentication mechanisms and implementing RBAC models to define and enforce granular authorization roles, ensuring data confidentiality and integrity in diverse organizational settings.	4,5
Practical 6	Managing transactions to ensure ACID properties (Atomicity, Consistency, Isolation, Durability) using transaction management techniques and protocols.	**2**	Students will develop expertise in managing transactions to guarantee ACID properties (Atomicity, Consistency, Isolation, Durability),	5,6

			employing robust transaction management techniques and protocols for reliable data operations in database systems.	
Practical 7	Designing and deploying distributed database systems to handle data across multiple nodes or locations while ensuring consistency and fault tolerance.	**2**	Students will learn to design and deploy distributed database systems, ensuring data consistency and fault tolerance across multiple nodes or locations, crucial for scalable and resilient data management in modern applications.	3
Practical 8	Implementing data integrity constraints such as primary keys, foreign keys, and check constraints to maintain data accuracy and reliability.	**2**	Students will gain proficiency in implementing data integrity constraints like primary keys, foreign keys, and check constraints, ensuring data accuracy and reliability in relational database systems.	3
Practical 9	Designing and optimizing database indexes to enhance query performance and facilitate efficient data retrieval.	**2**	Students will master designing and optimizing database indexes to improve query performance and streamline data retrieval operations, enhancing overall database efficiency.	3
Practical 10	Implementing database recovery techniques such as backup and restore procedures to ensure data availability and resilience against failures.	**2**	Students will learn to implement database recovery techniques, including backup and restore procedures, to ensure data availability and resilience against failures, maintaining database integrity and continuity.	3
Practical 11	Exploring object-oriented and object-relational database features for modeling complex data structures and relationships using inheritance and encapsulation.	**2**	Students will explore object-oriented and object-relational database features to effectively model complex data structures and relationships using inheritance and encapsulation, enhancing	3

			flexibility and scalability in database design.	
Practical 12	Utilizing data warehousing techniques to integrate and consolidate data from heterogeneous sources for analytical processing and decision support.	**2**	Students will learn to utilize data warehousing techniques to integrate and consolidate data from diverse sources, enabling effective analytical processing and decision support in organizational contexts.	3
Practical 13	Applying data mining algorithms to extract patterns and insights from large datasets stored in databases, supporting business intelligence and predictive analytics.	**2**	Students will apply data mining algorithms to extract valuable patterns and insights from large databases, empowering business intelligence and facilitating predictive analytics for informed decision-making.	3
Practical 14	Developing logical database designs to map conceptual models (e.g., ER diagrams) to physical database schemas optimized for performance and storage.	**2**	Students will develop proficiency in translating conceptual models (e.g., ER diagrams) into optimized physical database schemas, ensuring efficient performance and storage in logical database designs.	3
Practical 15	Managing and optimizing database performance by monitoring resource usage, tuning database parameters, and analyzing execution plans for optimization opportunities.	**2**	Students will learn to manage and optimize database performance through effective monitoring of resource usage, tuning database parameters, and analyzing execution plans, ensuring efficient operation and responsiveness in database systems.	3

TEXT BOOKS:

T1: . "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F.Korth, S. Sudarshan, McGraw-Hill.

REFERENCE BOOKS:

R1: "Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.

OTHER LEARNING RESOURCES:

O1: https://www.jbiet.edu.in/coursefiles/cse/HO/cse2/DBMSI-III.pdf

O2: https://cs.ulb.ac.be/public/_media/teaching/infoh303/dbmsnotes.pdf

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental concepts of database management systems with DBMS languages and data models.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
2	Apply the concepts of query languages such as DDL and DML for designing a relational database.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
3	Apply the ACID properties and concurrency control schemes to perform transaction processing with database recovery.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
4	Apply security to the database by checking for authentication and authorization	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
5	Apply advanced topics of data warehousing and data mining, distributed databases and web databases.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS2201R	**Database Management Systems**	**CO1**	2	2	3	2	2				3		3	3
		CO2	3	2	3	2	2				2		3	3
		CO3	3	2	3	3	3				3		3	2
		CO4	3	2	3	1	3				2		3	2
		CO5	3	2	3	3	3				2		2	2

SEMESTER – IV									
Course Title	**Operating Systems**								
Course code	**24BTCS2202R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To learn the mechanisms of OS to handle processes and threads and their communication and the mechanisms involved in memory management in contemporary OS. 2. To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols 3. To know the components and management aspects of concurrency management.								
CO1	Understand operating system evolution, types, services, structures (UNIX, Windows).								
CO2	Analyse process, thread concepts, CPU scheduling, relationships, and states effectively.								
CO3	Evaluate IPC solutions, evaluate deadlock conditions, implement prevention, detection, recovery.								
CO4	Analyse memory management, mapping, allocation, paging, virtual memory, page replacement.								
CO5	Understand I/O hardware, controllers, DMA, secondary storage, disk scheduling, file management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction:** Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**7**	Students will grasp the concepts of operating systems, including their generations, types, services, system calls, and structural architectures (layered, monolithic, microkernel), as well as understand the concept of virtual machines for efficient resource management and isolation.	**1,2**
II	**Processes:** Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. **Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, **Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling	**8**	Students will understand processes and threads in operating systems, including their definitions, states, relationships, and management mechanisms like context switching and PCBs, alongside exploring process scheduling foundations, criteria, algorithms, and multiprocessor considerations for efficient resource utilization and system responsiveness.	**2,3,4**

	algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.			
III	**Inter-process Communication:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution **Deadlocks:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.	**8**	Students will master inter-process communication concepts including critical sections, race conditions, mutual exclusion, and hardware-based solutions, alongside understanding deadlocks, their conditions, prevention strategies like the Banker's algorithm, and techniques for detection and recovery in operating systems.	**2,3,4,5**
IV	**Memory Management:** Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging. **Virtual Memory**: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).	**8**	Students will comprehend memory management in operating systems, covering concepts like logical and physical address mapping, memory allocation strategies, including contiguous and paging methods, and virtual memory essentials such as locality of reference, demand paging, and page replacement algorithms like FIFO, LRU, and optimal for efficient resource utilization and system performance.	**2,3**
V	**I/O Hardware:** I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure.. **File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management. **Disk Management:** Disk structure, Disk scheduling, Disk reliability, Disk formatting, Boot-block, Bad blocks.	**8**	Students will grasp I/O hardware principles including devices, controllers, and DMA, understand I/O software goals such as interrupt handling and device drivers, explore file management concepts like access methods, types, operations, and directory structures, and learn about disk management covering structure, scheduling, reliability, and formatting for efficient data storage and retrieval in operating systems.	**1,2,3**

Practical Component				
Practical 1	Basic Linux Commands and Overview.	**4**	Gain proficiency in fundamental Linux commands and system navigation.	2,3
Practical 2	Write Shell Script for followings a) To find the global complete path for any file. b) To broadcast a message to a specified user or a group of users logged on any terminal. c) To copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories. d) To compare identically named files in two different directories and if they are same, copy one of them in a third directory e) To delete zero sized files from a given directory (and all its sub-directories). f) To display the name of those files (in the given directory) which are having multiple links. g) To display the name of all executable files in the given directory. h) Write a script to display the date, time and a welcome message (like Good Morning etc.). The time should be displayed with "a.m." or "p.m." and not in 24 hours notation. i) Write a script to display the directory in the descending order of the size of each file.	**10**	Develop proficiency in shell scripting through practical exercises in file management, system administration, and user interaction on Linux systems	2,3,5
Practical 3	Implementation of FCFS (First Come First Serve) CPU Scheduling.	**2**		3
Practical 4	Implementation of SJF (Shortest Job First) CPU Scheduling.	**2**		3
Practical 5	Implementation of Round Robin (RR) CPU Scheduling.	**2**		3
Practical 6	Implementation of Priority CPU Scheduling Algorithm.	**2**		3

Practical 7	Implementation of FIFO Replacement Algorithm.	**2**		3
Practical 8	Implementation of Optimal Page Replacement Algorithm.	**2**		3
Practical 9	Implementation of LRU Page Replacement Algorithm by Stack method	**2**		3
Practical 10	Implement the producer-consumer problem using threads	**2**		3

TEXT BOOKS:

T1: Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.

T2: Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

REFERENCE BOOKS:

R1: Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing.

R2: Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley

OTHER LEARNING RESOURCES:

O1: "Operating System Concepts" by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne.

O2: "Modern Operating Systems" by Andrew S. Tanenbaum.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand operating system evolution, types, services, structures (UNIX, Windows).	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Analyse process, thread concepts, CPU scheduling, relationships, and states effectively.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Evaluate IPC solutions, evaluate deadlock conditions, implement prevention, detection, recovery.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Analyse memory management, mapping, allocation, paging, virtual memory, page replacement.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Understand I/O hardware, controllers, DMA, secondary storage, disk scheduling, file management.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2202R	Operating Systems	CO1	3	2	2	3	2					3		2
		CO2	3	2	2	2	2					2		1
		CO3	3	3	3	3	2					3		1
		CO4	2	2	3	2	3					2		2
		CO5	2	2	2	3	2					3		1

SEMESTER – IV									
Course Title	**Probability and Statistics**								
Course code	**24BTCS2203R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T	**2**	**1**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Mathematics**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To make understand to solve first and second order ordinary differential equation. 2. To provide conceptual understanding on higher order ODE with constant coefficients. 3. To provide an overview of probability to engineers. 4. To understand the concept of basic statistical measurements. 5. To analyse the data by applying statistical methods.								
CO1	Enabling solving skills of ordinary differential equations of various order.								
CO2	Enabling solving skills of higher order ODE with constants coefficients.								
CO3	Enable to formulate and solve problems involving random variables.								
CO4	Enable to solve problems of various probability distribution.								
CO5	Analysing experimental data by applying statistical methods.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**First order ordinary differential equations:** Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.	**7**	Understanding first-order ODEs: exact, linear, Bernoulli's, Euler's equations, and equations solvable for p, y, x, and Clairaut's type.	**1,2**
II	**Ordinary differential equations of higher orders:** Second order linear differential equations with variable coefficients, method of variation of parameters, Cauchy-Euler equation, Higher order ODE with constant coefficients.	**9**	Understanding higher-order ODEs: second-order linear equations with variable coefficients, variation of parameters, Cauchy-Euler equation, and constant coefficient ODEs.	**2,3**
III	**Probability:** Basic definitions on Probability, conditional probability, independence; Discrete random variables, Independent random variables., Probability distributions: Binomial, Poisson and Normal, Poisson approximation to the binomial distribution.	**8**	Understanding probability basics: definitions, conditional probability, independence, discrete random variables, and distributions (Binomial, Poisson, Normal) including Poisson approximation.	**2,3,5**
IV	**Basic Statistics:** Measures of central tendency: Moments, skewness and Kurtosis, Correlation and regression.	**8**	Mastering basic statistics: measures of central tendency, moments, skewness, kurtosis, and understanding correlation and regression for data analysis.	**2,3,4**

V	**Applied Statistics:** Test of significance: Large sample test for single proportion. Difference of proportions, single mean, difference of means and difference of standard deviations.	**8**	Understanding applied statistics: significance testing for large samples, including tests for single proportion, difference of proportions, means, and standard deviations.	**2,3**

TEXT BOOKS:

T1: Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
T2: N. P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.

REFERENCE BOOKS:

R1: S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.
R2: S. L. Ross, Differential Equations, 3rd Ed. Wiley India, 1984.

OTHER LEARNING RESOURCES:
O1: tutorialspoint- offers a course "Statistics-Probability" for all beginners
O2: youtube- Statistics and Probability full course.
O3: NPTEL- a video lecture on "Probability and Statistics" full course by Prof. Somesh Kumar, IIT Kharagpur.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enabling solving skills of ordinary differential equations of various order.	PO1, PO2, PO5, PO12
2	Enabling solving skills of higher order ODE with constants coefficients.	PO1, PO2, PO5, PO12
3	Enable to formulate and solve problems involving random variables.	PO1, PO2, PO4, PO6, PO12
4	Enable to solve problems of various probability distribution.	PO1, PO4, PO5, PO10, PO12
5	Analysing experimental data by applying statistical methods.	PO2, PO4, PO5, PO11, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2203R	Probability and Statistics	CO1	3	3			2							1
		CO2	3	3			2							2
		CO3	2	2		2		1						1
		CO4	2			2	2					1		1
		CO5		3		3	2						1	2

SEMESTER – IV									
Course Title	**Formal Language and Automata Theory**								
Course code	**24BTCS2204R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To gain a solid understanding of formal languages, grammars, automata, and their theoretical foundations in computer science. 2. Learn to design, analyse, and compare different computational models such as finite automata, pushdown automata, and Turing machines. 3. Develop skills to apply formal methods and automata theory concepts to solve complex problems in language processing, compiler design, and algorithm development.								
CO1	Understand the fundamental characteristics of formal languages and formal grammars, showcasing foundational linguistic knowledge.								
CO2	Recognize the similarity between deterministic and non-deterministic finite automata, demonstrating comparative analysis skills.								
CO3	Evaluate the minimization processes of both deterministic and non-deterministic finite automata, applying critical thinking skills.								
CO4	Analyse the resemblance between non-deterministic push-down automata and context-free grammars, demonstrating advanced understanding and synthesis of concepts.								
CO5	Examine the fundamental characteristics of Turing machines and their computational applications, showcasing analytical skills in computing theory.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction** Definitions: Language, Grammar, Automata, Relation between language, Grammar and automata, Importance of automata theory. **Finite Automata (FA)** Introduction of FA and its example. DFA: Definition, Processing strings, Transition functions, Language of a DFA, NDFA: Non-determinism, Definition, Processing strings, Extended transition functions, Language of a NFA, Kleene's theorem, Epsilon transitions, Applications of FA in text search.	**8**	To grasp finite automata theory, design deterministic/nondeterministic automata, understand regular languages, and apply these concepts to problem-solving.	**1, 2,3**

II	**Regular expressions and regular languages** Memory required to recognize a language, Regular expressions, Regular expression to finite automata & vice versa, Algebraic laws for regular expressions, Applications of regular expressions, Criterion for regularity, Regular languages, Properties of Regular languages	**8**	learn to construct and interpret regular expressions, understand regular languages, and apply these concepts to pattern matching.	**2,3**
III	**Context Free Grammars and Languages** Definition, Left most and right most grammars, Parse trees, Ambiguity: Ambiguous grammar, Removing ambiguity, Normal forms, Applications of context free grammars: Parsers	**10**	To comprehend context-free grammars, parse trees, Chomsky hierarchy, and apply these concepts to analyze and generate context-free languages.	**2,4**
IV	**Pushdown automata (PDA) and context free languages (CFL)** Definition & representation of pushdown automata, Acceptance by PDA: By final state, By empty stack, Deterministic PDA, Equivalence of PDA and CFL, Pumping Lemma for CFL Closure properties of CFL, Testing membership of context free, Decision problems for CFLs.	**10**	To master pushdown automata theory, understand context-free languages, parse trees, and apply these concepts to language recognition and parsing.	**2,3**
V	**Turing machines** Definition, Language of a Turing machine, Programming Turing machines, The Church-Turing thesis, A simple programming language, Extensions of the basic Turing machine. **Recursively enumerable languages & Undesirability** The halting problem, The post correspondence problem, Time and space complexity of Turing machines, Complexity classes	**9**	To grasp Turing machine theory, recursively enumerable languages, undecidability, and apply these concepts to analyze computability and algorithmic problems.	**2,3,4**

Textbooks:

T1: John. E. Hopcroft, Rajeev Motwani, Jeffry Ullman,‖ Introduction to Automata Theory, Languages and Computation‖, Pearson Education.

T2: John Martin,―Introduction to Languages and the Theory of Computation, Tata McGraw

Hill

Reference Books:

R1: Adesh K. Pandey,―An Introduction to Automata Theory & Formal Languages‖, Katson Books.
R2: Peter Linz,―An Introduction to Formal Languages and Automata‖, Narosa.
R3: Zvi Kohavi and Niraj K. Jha,―Switching and Finite Automata Theory‖, Tata McGraw Hill.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental characteristics of formal languages and formal grammars, showcasing foundational linguistic knowledge.	CO1, CO2, CO3, CO10, CO12
2	Recognize the similarity between deterministic and non-deterministic finite automata, demonstrating comparative analysis skills.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Evaluate the minimization processes of both deterministic and non-deterministic finite automata, applying critical thinking skills.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Analyze the resemblance between non-deterministic push-down automata and context-free grammars, demonstrating advanced understanding and synthesis of concepts.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
5	Examine the fundamental characteristics of Turing machines and their computational applications, showcasing analytical skills in computing theory. .	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2204R	Formal Language and Automata Theory	CO1	2	2	1							1		1
		CO2	3	3	2	2	1					1		2
		CO3	3	3	3	1	1					1		2
		CO4	3	3	2	2	1					1		2
		CO5	3	3	2	2	1					1		2

SEMESTER – IV									
Course Title	**PE-I : Artificial Intelligence**								
Course code	**24BTCS2205R**	**Total credits:4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Programming Skills, Data structures	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. Describe the field of AI and its subfields machine learning, NLP and computer vision 2. Learn and implement key NLP algorithms and models, such as virtual agents, chatbots, sentiment classifiers, and language translators, to solve a variety of natural language understanding tasks. 3. Explore Practical Applications of Computer Vision**,** Investigate real-world applications of computer vision in various fields, including autonomous vehicles, surveillance systems, medical imaging, and augmented reality.								
CO1	Remember and understand relevance of AI in today's world.								
CO2	Gain an idea about Machine Learning application in real world problem.								
CO3	Develop an understanding deep learning concept in AI.								
CO4	Apply and Analyze data by applying various machine learning algorithms.								
CO5	Enhance problem-solving abilities with IBM Cloud service.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Artificial Intelligence:** Overview of Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL), Natural Language Processing(NLP), Computer Vision. Describe the types of AI, Explain what the main focus of AI is, List of practical applications of AI, Describe the eras of computing, Explain the difference between deterministic and probabilistic systems, Road Map of AI.	**8**	Gain a foundational understanding of Artificial Intelligence, its types, key focus areas, practical applications, historical development, and the differences between deterministic and probabilistic systems, along with an overview of the AI roadmap.	**1,2**
II	**Introduction to Machine Learning:** Introduction to Machine Learning, Supervised Learning & Unsupervised Learning. Supervised Learning Algorithms: Linear Regression, Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, KNN. Unsupervised Learning Algorithms: K-Means Clustering, Hierarchical Clustering.	**8**	Develop a fundamental understanding of Machine Learning, including supervised and unsupervised learning techniques, and gain practical experience with algorithms such as Linear Regression, Logistic Regression, SVM, Decision Trees, Random Forest, KNN, K-Means Clustering, and Hierarchical Clustering.	**1,2,4**

III	**Introduction to Deep Learning:** Introduction to Deep Learning: Neural Networks, Layers, and Activation Functions, Back Propagation, Optimizers, Deep Learning Algorithm: ANN, CNN, RNN, LSTM with Tensorflow and Keras, Introduction to Autoencoders, Generative Adversarial Networks (GANs), Restricted Boltzmann Machines (RBMs) and Applications.	**8**	Students will gain a comprehensive introduction to Deep Learning, covering neural networks, activation functions, backpropagation, optimizers, and algorithms such as ANN, CNN, RNN, LSTM using TensorFlow and Keras, along with an understanding of Autoencoders, GANs, RBMs, and their applications in various domains.	**2,3,4,5**
IV	**Introduction to Natural Language Processing and Computer Vision:** Introduction to Natural Language Processing, NLP application in real world, components of NLP, Virtual Agents overview, Chatbot and its parts, Tokenization, Stemming & Lemmatization in NLP, BagofWords,TF-IDF,Word2Vec, Word Embedding in NLP. Introduction to Computer Vision, Image Classification and Object Recognition, Feature Detection and Description algorithms with OpenCV.	**8**	Acquire foundational knowledge in Natural Language Processing (NLP) including its applications, components, and techniques like tokenization, stemming, TF-IDF, and Word Embedding, as well as in Computer Vision covering image classification, object recognition, and feature detection using OpenCV.	**2,3**
V	**Evolution from DeepQA to Watson services on IBM Cloud:** Explain the DeepQA architecture was; Explain why IBM decided to commercialize Watson and Describe the evolution of Watson services from the original DeepQA architecture to the present. Introduction to IBM Watson: explore Watson Studio, IBM's integrated environment for data scientists, developers, and domain experts, to build and deploy ML models. Creating Chatbots using Watson Assistant: learn the basics of Watson Assistant, including intents, entities, dialog flows, and integration capabilities. Natural Language Understanding (NLU), Language Translator, Text to Speech, Speech to Text on IBM Cloud.	**8**	Understand the evolution of IBM Watson from DeepQA to its current services on IBM Cloud, including Watson Studio for ML model development, Watson Assistant for creating chatbots, and capabilities like Natural Language Understanding, Language Translator, Text to Speech, and Speech to Text on the IBM Cloud platform.	**1,2,3**

Practical Component				
Practical 1	**Exploring AI Types and Applications :** Create a presentation or simulation to classify types of AI (Reactive, Limited Memory, Theory of Mind, Self-Aware) and demonstrate real-world applications in NLP, Computer Vision, and robotics.	**3**	Understand the classification and scope of AI. Explore real-world use cases across industries.	1,2 ,3
Practical 2	**Deterministic vs. Probabilistic Systems** Simulate a deterministic system (e.g., rule-based chatbot) and a probabilistic system (e.g., predictive text) using Python. Compare their behaviour.	**3**	Differentiate between deterministic and probabilistic systems. Analyze the strengths and limitations of each system.	4,5
Practical 3	**Implementing Supervised Learning Algorithms** Implement and compare Linear Regression, Logistic Regression, and Support Vector Machines on a dataset (e.g., Iris or Titanic dataset).	**3**	Understand supervised learning algorithms and apply them to real-world datasets. Analyze model performance using evaluation metrics.	2,3,4
Practical 4	**Clustering Algorithms for Unsupervised Learning:** Implement K-Means and Hierarchical Clustering algorithms on a dataset to group data points and visualize clusters.	**3**	Understand and apply unsupervised learning algorithms. Evaluate the quality of clustering using metrics like Silhouette Score.	3,5
Practical 5	**Building a Neural Network with TensorFlow** Build an Artificial Neural Network (ANN) for binary classification using TensorFlow and evaluate its performance using accuracy, precision, and recall.	**3**	Understand the structure and functionality of ANNs. Implement and optimize neural networks for specific tasks.	3,5
Practical 6	**Introduction to Natural Language Processing** Implement tokenization, stemming, lemmatization, and a Bag-of-Words model on a text dataset to preprocess and analyze textual data.	**3**	Understand fundamental NLP preprocessing techniques. Apply these techniques to clean and analyze textual data.	1,2,3
Practical 7	**Image Classification Using CNN** Build a Convolutional Neural Network (CNN) in TensorFlow to classify images from the CIFAR-10 dataset.	**3**	Understand the structure and functionality of CNNs. Train and evaluate CNNs for image classification tasks.	3,6

Practical 8	**Creating a Chatbot with Watson Assistant** Use IBM Watson Assistant to build a chatbot with intents, entities, and dialog flows. Test it with real-world queries.	**3**	Understand the components and functionality of chatbots. Design and deploy a simple chatbot using IBM Watson.	3,6
Practical 9	**Generative Adversarial Networks (GANs)** Objective: Build a simple GAN in TensorFlow to generate synthetic images from noise, using a dataset like MNIST.	**3**	Understand the architecture of GANs and their training process. Generate and evaluate synthetic data using GANs.	3,5
Practical 10	**Exploring Watson Services** Objective: Use IBM Watson Studio to deploy a machine learning model and integrate services like Speech-to-Text, Text-to-Speech, and Natural Language Understanding.	**3**	Explore IBM Watson's capabilities for AI solutions. Apply Watson services to build integrated AI applications.	3,6

TEXT BOOKS:

T1: Introduction to Data Science: B.Uma Maheswari& R. Sujata.

T2: Deep Learning with TensorFlow and Keras by Amita Kapoor, Antonio Gulli & Sujit pal

REFERENCE BOOKS:

R1: Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence" by Jon Krohn, Grant Beyleveld, and Aglaé Bassens.

R2: Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth.

OTHER LEARNING RESOURCES:

O1: Python & Machine Learning -https://www.w3schools.com/python/python_ml_getting_started.asp

O2: https://developer.ibm.com/technologies/artificial-intelligence

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO-PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Remember and understand relevance of AI in today's world.	PO1, PO2, PO3, PO4, PO10, PO12

2	Gain an idea about Machine Learning application in real world problem.	PO1, PO2, PO3, PO4, PO10, PO12
3	Develop an understanding deep learning concept in AI.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Apply and Analyze data by applying various machine learning algorithms.	PO1, PO2, PO3, PO4, PO10, PO12
5	Enhance problem-solving abilities with IBM Cloud service.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS2205R	**Artificial Intelligence**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – IV									
Course Title	**PE-I: Introduction to Internet of Things**								
Course code	**24BTCS2205R**	**Total credits: 4** **Total hours 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To introduce the different terminology, technology of IoT and its applications 2. To introduce the concept of M2M (machine to machine) with necessary protocols 3. To introduce the Python Scripting Language which is used in many IoT devices 4. To introduce the Raspberry PI platform, that is widely used in IoT applications 5. To introduce the implementation of web-based services on IoT devices								
CO1	Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules.								
CO2	Understand IoT value chain structure (device, data cloud), application areas and technologies involved.								
CO3	Market forecast for IoT devices with a focus on sensors								
CO4	Explore and learn about Internet of Things with the help of preparing projects designed for Raspberry Pi								
CO5	Apply database knowledge to a real-world data analytics project.								
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**						**BL**
I	**Introduction to Internet of Things-** Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Embedded Systems, Domain Specific IoTs – Home, City, Environment, Energy, Agriculture and Industry.	**8**	Understand IoT concepts, architecture, communication protocols, sensor integration, and design basic IoT applications for various domains.						**1,2**

II	**IoT Physical Devices and Endpoints-** Introduction to Arduino and Raspberry Pi- Installation, Interfaces (serial, SPI, I2C), Controlling Hardware-Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors.	**8**	comprehend IoT device hardware, sensors, actuators, communication interfaces, and configure/manage physical endpoints for IoT applications effectively.	**2,4**
III	**IOT Data Communication:** Transfer data by Wireless / Wired connectivity, Ipv4/Ipv6, 2g ,3g ,4g & 5g IEEE 802.15.4, IEEE 802.15.4e, 802.11ah, Relay Access Point (AP) Grouping of station Target Wake Time (TWT), Software defined networks, network function virtualization, difference between SDN and NFV for IoT.	**10**	Understand IoT communication protocols, data formats, transmission techniques, and implement efficient data communication solutions for IoT networks.	**2,3**
IV	**IOT Data Storage & Retrieval:** Overview and Role of Storage in Cloud / Server /Inhouse Storage, Databases Connectivity with IOT devices and users, Case Study over Mysql / NoSql / NewSql, Cloud Services and Administration, Case Study Of Big Data & Hadoop Platforms.	**10**	Comprehend IoT data storage architectures, database systems, retrieval methods, and implement scalable and efficient data storage solutions for IoT applications.	**2,3,4**
V	**IoT Data Analysis:** An Introduction to Data Science and Analytics Data Analysis Using NumPy , Pandas Data Cleaning, Missing Data, Feature extraction, Data Visualization – Static, Dynamic & Geographical Visualization. Linear Regression, Supervised Learning (Classification) and Unsupervised Learning (Clustering) algorithms, Time Series Analysis.	**9**	Analyze IoT data using statistical techniques, machine learning algorithms, and extract actionable insights to optimize IoT system performance.	**2,3,4**

Practical Component				
Practical 1	**Setting up IoT Devices** Set up and configure an Arduino board or Raspberry Pi. Write a basic program to blink an LED and control it via GPIO pins.	**3**	Understand the basics of IoT hardware setup. Apply programming to control hardware devices.	1,2,3
Practical 2	**IoT Communication Models** Implement an IoT communication model (e.g., Publish-Subscribe using	**3**	Explore IoT communication protocols like MQTT.	2,3

	MQTT) to send sensor data from one device to another.		Implement data exchange between IoT devices.	
Practical 3	**IoT-enabled Home Automation** Create a home automation system using IoT (e.g., controlling a light bulb or fan with an IoT-enabled app). Use Raspberry Pi or Arduino with relays.	**3**	Understand IoT applications in home automation. Design a system to control devices remotely.	3,6
Practical 4	**Wireless Sensor Network Simulation** Simulate a wireless sensor network (WSN) using IoT-enabled devices to monitor temperature and humidity data.	**3**	Understand WSN and its role in IoT. Apply IoT technologies to environmental monitoring.	2,3
Practical 5	**Data Transfer and Communication in IoT** Transfer sensor data over different wireless protocols and observe the differences in speed and power consumption.	**3**	Analyze various communication protocols for IoT. Evaluate the suitability of protocols for specific use cases.	4,5
Practical 6	**IoT Data Storage and Retrieval** Store IoT sensor data in a cloud database (e.g., Firebase or AWS) and retrieve it for display on a web dashboard.	**3**	Understand cloud storage integration with IoT devices. Apply database connectivity for IoT data retrieval.	3,6
Practical 7	**Data Analysis and Visualization** Objective: Use Python libraries (NumPy, Pandas, and Matplotlib) to analyze IoT sensor data and create static and dynamic visualizations.	**3**	Understand IoT data analysis and visualization. Apply data science techniques to extract insights from IoT data.	2,3
Practical 8	**Implementing Linear Regression for IoT Data** Objective: Use IoT sensor data to predict future values using linear regression models in Python.	**3**	Understand machine learning techniques for IoT data analysis. Apply linear regression for prediction tasks.	3,4
Practical 9	**IoT Security Simulation** Simulate an IoT system with encrypted data transfer using secure protocols.	**3**	Understand the importance of security in IoT communication. Implement secure communication between IoT devices.	2,3
Practical 10	**IoT with Big Data and Hadoop** Perform a case study to store and process IoT-generated data using Hadoop or similar Big Data platforms. Analyze data using Hive or Spark.	**3**	Understand the integration of IoT and Big Data platforms. Analyze large-scale IoT data using Big Data tools.	3,5

Text Books:

T1: Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547

T2: Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759

Reference Books:

R1: Peter Waher, 'Learning Internet of Things', Packet Publishing, 2015 3. Editors Ovidiu Vermesan

R2: Peter Friess, 'Internet of Things – From Research and Innovation to Market Deployment', River Publishers, 2014

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules.	CO1, CO2, CO3, CO4, CO5, CO10, CO12.
2	Understand IoT value chain structure (device, data cloud), application areas and technologies involved.	CO1, CO2, CO3, CO4, CO5, CO10, CO12.
3	Market forecast for IoT devices with a focus on sensors	CO1, CO2, CO3, CO4, CO5, CO10, CO12.
4	Explore and learn about Internet of Things with the help of preparing projects designed for Raspberry Pi	CO1, CO2, CO3, CO4, CO5, CO10, CO12.
5	Apply database knowledge to a real-world data analytics project.	CO1, CO2, CO3, CO4, CO5, CO10, CO12.

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS2205R	**Introduction to Internet of Things**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER IV									
Course Title	**Quantitative Aptitude For Computer Science**								
Course code	**24BTCS2209R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:30T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives	1. To develop the mathematical foundation required for solving quantitative problems in computer science. 2. To enhance logical reasoning and analytical skills for computational thinking. 3. To introduce the concepts of data interpretation and statistical analysis for decision-making. 4. To familiarize students with combinatorial and probability techniques relevant to algorithm design. 5. To understand optimization methods and computational models for real-world applications in computer science.								
CO1	Apply mathematical principles and logic to solve quantitative problems in computer science.								
CO2	Analyze and interpret data effectively for informed decision-making in technical scenarios.								
CO3	Demonstrate problem-solving skills using combinatorial techniques and probability models.								
CO4	Develop optimization solutions for computational problems using mathematical tools.								
CO5	Evaluate and analyze the complexity of algorithms and systems using appropriate models.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Mathematical Foundations** - Number Theory: Divisibility, Modular Arithmetic, GCD, LCM - Linear Algebra: Matrices, Determinants, Eigenvalues, Eigenvectors - Permutations and Combinations: Principles of Counting, Probability Basics	**6**	Understand basic mathematical concepts and their applications in computational problems.	**1,2,3**
II	**Logical Reasoning and Problem Solving** - Logical Deductions, Syllogisms - Puzzles and Series: Number, Letter, and Symbol Series - Coding-Decoding, Blood Relations	**6**	Develop logical thinking and problem-solving skills essential for algorithm design.	**3,4**
III	**Data Interpretation and Analysis** - Tabular Data: Representation and Analysis - Graphs and Charts: Bar, Line, Pie - Basic Statistical Measures: Mean, Median, Mode, Standard Deviation	**6**	Analyse and interpret data efficiently for decision-making and pattern recognition.	**4,5**
IV	Combinatorics and Probability - Advanced Probability: Bayes' Theorem, Conditional Probability - Binomial Theorem, Principle of Inclusion-Exclusion	**6**	Apply combinatorial techniques and probability concepts in computational problems.	**4,5**

	- Applications in Algorithm Analysis			
V	Optimization and Computational Models - Optimization: Linear Programming, Graph Theory Basics - Time and Space Complexity: Big O, Big Theta, Big Omega - Applications in Computer Science (e.g., Scheduling, Search, Sort Algorithms)	**6**	Understand optimization techniques and computational complexity for problem-solving.	**4,5**

TEXT BOOKS:

T1: Greene, D. E. - "Mathematical Structures for Computer Science: A Modern Treatment of Discrete Mathematics", W. H. Freeman, 7th Edition, 2014.

T2: Rosen, K. H. - "Discrete Mathematics and Its Applications", McGraw-Hill, 7th Edition, 2011.

T3: Babu, R. S. - "Quantitative Aptitude for Competitive Examinations", S. Chand Publishing, 2018.

REFERENCE BOOKS:

R1: Mazumdar, S. - "Probability, Statistics, and Random Processes for Engineers", Pearson, 4th Edition, 2015.

R2 : Papadimitriou, C. H., & Steiglitz, K. - "Combinatorial Optimization: Algorithms and Complexity", Dover Publications, 1998.

OTHER LEARNING RESOURCES:

O1: NPTEL Lectures:

Discrete Mathematics for Computer Science by Prof. Kamala Krithivasan, IIT Madras.

Probability and Statistics by Prof. Prathap Haridoss, IIT Madras.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply mathematical principles and logic to solve quantitative problems in computer science.	PO1, PO5, PO12
2	Analyze and interpret data effectively for informed decision-making in technical scenarios.	PO1, PO5, PO12
3	Demonstrate problem-solving skills using combinatorial techniques and probability models.	PO1, PO5, PO12
4	Develop optimization solutions for computational problems using mathematical tools.	PO1, PO5, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS2209R	Quantitative Aptitude For Computer Science	CO1	3				3							2
		CO2	2				2							1
		CO3	2				2							1
		CO4	2				2							1
		CO5												

SEMESTER – IV									
Course Title	**Digital Literacy**								
Course code	**24UCDL1002R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**All Degree / Diploma programme of Engineering, Computer Technology, Management, Humanities, Science, and Agricultural Sciences**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. Students will be able to understand the fundamentals of computer systems and Internet search along with advanced features of MS-Office. 2. Students will be able to understand about the introduction to Social Media and E-Commerce and utility software. 3. Students will develop a solid foundation in computational thinking, which includes problem-solving, algorithmic design, and logical reasoning.								
CO1	Fundamentals of Computer Systems, Office Automation and Internet Search.								
CO2	Know more about the Internet & Cyber World								
CO3	Know Social Media, E-Commerce, and apply the same for digital branding								
CO4	Know to use the digital payments and digital transactions, and other utility software								
CO5	Able to build basic level websites								

Practical Component

Unit-No.	Content	Contact Hour	Learning Outcome	BL
Practical 1	Disassemble and reassemble a desktop computer, identifying and explaining the function of each component. Discuss the importance of hardware compatibility.	**3**	By disassembling and reassembling a desktop computer, learners understand component functions, emphasize hardware compatibility for system stability, and gain practical troubleshooting skills.	3
Practical 2	Install an operating system (e.g., Windows or Linux) on a virtual machine. Configure essential settings, such as user accounts, network connections, and system updates.	**3**	Creating and configuring a virtual machine with an operating system develops skills in installation, setup of user accounts, network configurations, and system maintenance, enhancing proficiency in virtualized environments.	3
Practical 3	Use office software (e.g., Microsoft Office or Google Workspace) to create documents, spreadsheets, and presentations. Teach formatting, inserting images, and collaboration features.	**3**	Using office software for document creation, formatting, image insertion, and collaboration enhances proficiency in productivity tools, fostering effective communication and	3

			collaborative skills in professional settings.	
Practical 4	Instruct students on effective internet searching, including the use of search engines, keywords, and advanced search operators.	**3**	Teaching effective internet searching covers using search engines proficiently, employing appropriate keywords, and leveraging advanced search operators to refine results, enhancing research skills and information retrieval efficiency.	3
Practical 5	Set up email accounts, compose and send emails, attach files, and organize emails into folders. Discuss email etiquette and best practices.	**3**	Setting up email accounts, composing, sending emails, attaching files, organizing with folders, and understanding email etiquette enhances communication efficiency and professionalism, emphasizing clarity, conciseness, and proper etiquette in digital correspondence.	3
Practical 6	Use cloud storage services (e.g., Google Drive or Dropbox) to store and synchronize files. Create a backup of important data and discuss data recovery options.	**3**	Using cloud storage services for file storage and synchronization ensures accessibility and collaboration, while creating backups of important data safeguards against loss, with data recovery options providing security and continuity in digital operations.	3
Practical 7	Instruct students to create accounts on popular social media platforms (e.g., Facebook, Twitter, Instagram). Guide them through profile setup, privacy settings, and content posting.	**3**	Guiding students to create social media accounts, set up profiles, configure privacy settings, and post content fosters digital presence management, emphasizing responsible sharing, online safety, and personal branding awareness.	3
Practical 8	Have students create a content calendar for a fictional business or brand. Plan posts, including text, images, and hashtags, and use social media management tools to schedule posts.	**3**	Creating a content calendar for a fictional business includes planning posts with text, images, and hashtags, using social media management tools to schedule and optimize engagement, fostering	3

			skills in strategic content planning and digital marketing execution.	
Practical 9	Guide students in building a basic e-commerce website using platforms like Shopify or WooCommerce. They should add products, set up payment gateways, and configure the online store.	**3**	Guiding students to build a basic e-commerce website involves adding products, setting up payment gateways, and configuring the online store using platforms like Shopify or WooCommerce, fostering skills in online retail setup and management for digital entrepreneurship.	3
Practical 10	Set up a payment gateway for the e-commerce website created in Experiment 3. Test payment transactions and discuss security protocols such as SSL encryption.	**3**	Setting up a payment gateway for the e-commerce website involves integrating services like PayPal, Stripe, or others, testing transactions to ensure functionality, and emphasizing security protocols such as SSL encryption to protect customer data during online transactions, ensuring safe and secure e-commerce operations.	3

TEXT BOOKS:

T1: Sinha, Pradeep K. and Priti Sinha. *Computer Fundamentals: Concepts Systems Applications*. 3rd ed. New Delhi: BPB Publications.

T2: Goel, A. 2010. *Computer Fundamentals*. Pearson India.

REFERENCE BOOKS:

R1: Balaguruswamy, E. 2009. *Fundamentals of Computers*. Tata McGraw-Hill Education.

R2: Balaguruswamy, E. 2014. *Fundamentals of Computer and Programming (Updated Ed Sem. I, Au)*. Tata McGraw-Hill Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will have basic knowledge of cooking methods.	PO1, PO5, PO12
2	Students will gain the knowledge of organizing & Cleaning of Rooms.	PO1, PO5, PO12

3	Students will be able to gain the travel management concept.	PO1, PO5, PO12
4	Students will be able to acquire the knowledge of basic household's amenities for day- to-day use.	PO1, PO5, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24UCDL1002R	Digital Literacy	CO1	3				3							2
		CO2	2				2							1
		CO3	2				2							1
		CO4	2				2							1
		CO5												

SEMESTER – IV									
Course Title	**Basic Acclimatizing Skills**								
Course code	**24UULS2201R**	**Total credits:1** **Total hours: 52**	**L** **0**	**T** **0**	**P** **2**	**S** **0**	**R** **0**	**O/F** **0**	**C** **1**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**All the Under Graduate Programmes**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To impart knowledge of the fundamentals of Hospitality industry and its applications. 2. Students will be able to familiarize with the cooking equipment's & Utensils. 3. Students will be able to handle different modes of reservations.								
CO1	Students will have basic knowledge of cooking methods.								
CO2	Students will gain the knowledge of organizing & Cleaning of Rooms.								
CO3	Students will be able to gain the travel management concept.								
CO4	Students will be able to acquire the knowledge of basic household's amenities for day-to-day use.								
CO5	Students will be able to gain the hospitality management concept.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Accommodation Management** Telephone handling technique Organizing of Rooms. Cleaning equipment's and uses. Bed making Process.	**12**	Introduction to Accommodation Management covers telephone handling techniques, room organization, cleaning equipment usage, and bed making processes, ensuring comprehensive skills in guest service and room maintenance.	**1,2**
II	**Fundamental of Cooking** Uses of basic cooking equipment's Uses of fire & Fuel Different cuts of vegetables Uses of herbs & spices Regional Food Habits	**10**	Fundamentals of Cooking include mastering basic cooking equipment, understanding fire and fuel usage, learning various vegetable cuts, utilizing herbs and spices effectively, and exploring regional food habits for diverse culinary knowledge.	**2,3,4**
III	**Food and Beverage skills** Introduction to catering industry Types menus and beverages Identifications of Cutlery, crockery & glassware Table etiquettes or manners Customer handling skills or Situation Handling	**12**	Food and Beverage skills encompass an introduction to the catering industry, knowledge of menu and beverage types, identification and use of cutlery, crockery, and glassware, understanding table etiquette, and developing customer	**2,3,5**

			handling and situational management abilities.	
IV	**Travel management** Travel Documentation (Types) Application of passport & Visa Tourism products (UNESCO sites) Types of logistics in travel and tourism management	**10**	Travel Management involves understanding travel documentation types, applying passport and visa procedures, exploring tourism products such as UNESCO sites, and managing logistics in travel and tourism operations effectively.	**2,3**
V	**Basic Hospitality Skills** Various Egg Preparations Canapés preparations Mock tail & Shakes Preparations Butter Rice / Lemon Rice Various Lentils Preparations 1 non-veg preparation/ 1 veg preparation	**8**	Basic Hospitality Skills include mastering various egg preparations, creating canapés, preparing mocktails and shakes, making butter rice or lemon rice, mastering various lentil dishes, and preparing one non-vegetarian and one vegetarian dish to ensure a diverse culinary repertoire.	**1,2,3**

TEXT BOOKS:

T1: Arora K (2011). Theory of cookery, Frank brothers & company (pub) pvt.ltd.-New Delhi.
T2: Bruce H. Axler, Carol A. Litrides (2010) Food and Beverage Service Volume 1 of Wiley Professional Restauranteur, Guides.

REFERENCE BOOKS:

R1: Mohammed Zulfikar (2010) - Introductions to Tourism and Hotel Industry Introduction to Tourism and Hotel Industry. Vikas Publishing.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will have basic knowledge of cooking methods.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
2	Students will gain the knowledge of organizing & Cleaning of Rooms.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
3	Students will be able to gain the travel management concept.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
4	Students will be able to acquire the knowledge of basic household's amenities for day- to-day use.	PO1, PO2, PO3, PO4, PO7, PO10,PO12

5	Students will be able to gain the hospitality management concept.	PO1, PO2, PO3, PO4, PO7, PO10,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24UULS2201R	Basic Acclimatizing Skills	CO1	1	1	2	2			3			1		2
		CO2	1	1	2	2			3			1		2
		CO3	1	2	3	2			3			1		2
		CO4	2	3	2	3			3			1		2
		CO5	3	3	3	3			3			1		2

<table>
<tr><th colspan="9">SEMESTER – IV</th></tr>
<tr><td>Course Title</td><td colspan="8">Mini Project II</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">23BTCS228R</td><td rowspan="2">Total credits:1
Total hours: 48</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>4</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Pre-requisite</td><td colspan="2">NIL</td><td>Co-requisite</td><td colspan="6">NIL</td></tr>
<tr><td>Programme</td><td colspan="8">Bachelor of Technology in Computer Science and Engineering</td></tr>
<tr><td>Semester</td><td colspan="8">Winter/ II semester of the second year of the Programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="8">1. To develop the knowledge, skills and attitudes of a professional Computer engineering professional.
2. To become confident in designing engineering solutions to complex software problems utilising a systems approach.
3. Apply theoretical knowledge gained in earlier semesters to solve real-world problems through the development and implementation of a software project.</td></tr>
<tr><td>CO1</td><td colspan="8">Apply IT principles for real-world problem-solving using programming languages and network solutions.</td></tr>
<tr><td>CO2</td><td colspan="8">Contribute to complex projects as a team member, demonstrating effective personal and team management.</td></tr>
<tr><td>CO3</td><td colspan="8">Develop programming skills to create and assess software, hardware, and network solutions.</td></tr>
<tr><td>CO4</td><td colspan="8">Apply personal and team management skills as a professional software developer.</td></tr>
<tr><td>CO5</td><td colspan="8">Cultivate employability skills and uphold professionalism in software development.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="3">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:

a) theoretical studies
b) computer simulations
c) hardware construction

3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the general area of work being</td><td>48</td><td colspan="3">This course involves individual or small group projects supervised by academic staff, focusing on topics aligned with student and supervisor interests. Students will conduct extensive literature reviews to understand current developments in their chosen technical area. They will then engage in detailed technical work, employing theoretical studies, computer simulations, or hardware construction.
Throughout the project, students will document their progress through regular reports or a professional journal, ensuring accountability and effective time management. They will present their findings and contributions in a seminar, preparing a formal report that outlines the project's scope, methodologies, and results. The</td><td>1,2,3,4,5</td></tr>
</table>

	undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.		culmination will involve showcasing their work in a public forum, utilizing poster presentations and operational demonstrations of hardware and software. This comprehensive approach aims to enhance students' research, technical, communication, and presentation skills in a real-world project environment.	

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
23BTCS228R	Mini Project II	CO 1	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

SEMESTER – V									
Course Title	**Data Communication and Computer Network**								
Course code	**24BTCS3101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Understand the fundamentals of data communications principles of media access, switching, routing and flow control. 2. Understand the basics of network protocol design and analysis. 3. Be familiar with the TCP/IP protocol suite and with application layer protocols. 4. Be able to write network-capable programs using the socket libraries. 5. Understand the structure of client-server systems and be able to build client-server programs.								
CO1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.								
CO2	Analyze data link layer techniques, flow control, and error protocols.								
CO3	Analyse network layer protocols along with routing issues.								
CO4	Summarize transport and application layer operations and protocols along with QoS services.								
CO5	Design and execute computer network programming projects, showcasing advanced skills in network application development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to networks, internet, protocols and standards, the OSI model, layers in OSI model, TCP/IP suite, Addressing, Analog and digital signals. Physical Layer: digital transmission, multiplexing, transmission media, circuit switched networks, Datagram networks, virtual circuit networks, switch and Telephone network.	**8**	Understand the OSI model, TCP/IP suite, and basic concepts of addressing, signals, and transmission media.	**2**
II	Data link layer: Introduction, Block coding, cyclic codes, checksum, framing, flow and error control, Noiseless channels, noisy channels, HDLC, point to point protocols	**8**	Learn error detection and correction techniques, framing, and protocols like HDLC and point-to-point protocols.	**2,3**
III	Network Layer: Logical addressing, internetworking, tunnelling, address mapping, ICMP, IGMP, forwarding, uni-cast routing protocols, multicast routing protocols.	**10**	Gain knowledge on logical addressing, routing protocols, and address mapping techniques.	**2,4**
IV	Transport Layer: Process to process delivery, UDP and TCP protocols, SCTP, data traffic, congestion, congestion control, QoS, integrated services, differentiated services, QoS in switched networks.	**10**	Comprehend the fundamentals of process-to-process delivery, TCP/UDP protocols, and congestion control mechanisms.	**2,3**

V	Application Layer – Domain name space, DNS in internet, electronic mail, FTP, WWW, HTTP, SNMP, multi-media, network security	**9**	Understand domain name space, DNS operations, email protocols, FTP, HTTP, and basics of network security.	**2**
Practical Component				
Practical 1	Configure a simple network consisting of multiple devices such as routers, switches, and computers. Test connectivity and troubleshoot any issues encountered.	2	Demonstrate the ability to configure basic network settings and troubleshoot connectivity problems.	1,2,3
Practical 2	Use network analysis tools like Wireshark to capture and analyze network traffic. Identify different protocols, their headers, and their roles in the OSI model.	2	Gain proficiency in protocol analysis and understanding of the OSI model layers.	3,4
Practical 3	Use oscilloscopes or signal analysers to examine analog and digital signals transmitted over the network. Interpret signal characteristics and identify any anomalies.	2	Understand the fundamentals of analog and digital signals in networking and their analysis	3,4,5
Practical 4	Implement a block coding technique such as Hamming codes to detect and correct errors in transmitted data frames. Analyse its effectiveness in error detection and correction.	2	Gain hands-on experience in implementing error detection and correction mechanisms in the data link layer.	3,4,5
Practical 5	Implement CRC error detection mechanism in a data link layer protocol such as Ethernet. Calculate CRC values for given data frames and verify error detection capability.	2	Understand the theory and practical implementation of CRC for error detection.	3,4,5
Practical 6	Implement and analyze flow control mechanisms such as Stop-and-Wait and Sliding Window protocols. Measure throughput and efficiency under different network conditions.	2	Understand the concepts of flow control and error control in data link layer protocols.	3,4,5
Practical 7	Configure IP addresses using both IPv4 and IPv6 addressing schemes for a network topology. Verify connectivity and troubleshoot addressing conflicts.	2	Demonstrate proficiency in configuring logical addresses andunderstanding IP addressing schemes.	3,4,5
Practical 8	Implement and compare different routing protocols such as RIP, OSPF, and BGP in a simulated network environment. Analyze routing tables and convergence times.	2	Understand the operation of routing protocols and their impact on network performance.	3,4,5
Practical 9	Configure Address Resolution Protocol (ARP) and Neighbor Discovery Protocol (NDP) for	2	Gain practical experience in address resolution and mapping techniques.	3,4,5

	mapping network layer addresses to data link layer addresses. Verify mappings and troubleshoot resolution issues.			
Practical 10	Capture and analyse UDP and TCP traffic using packet analysis tools. Compare their characteristics, such as connection establishment, reliability, and overhead.	2	Understand the differences between UDP and TCP protocols and their suitability for different applications.	3,4,5
Practical 11	Configure QoS mechanisms such as traffic prioritization, traffic shaping, and congestion management in a network. Measure and analyze QoS metrics under different traffic loads.	2	Understand the importance of QoS in ensuring reliable and efficient network performance.	3,4,5
Practical 12	Implement and evaluate congestion control algorithms such as TCP's congestion avoidance and congestion control mechanisms. Measure throughput and packet loss under congested conditions.	2	Gain practical insights into congestion control mechanisms and their impact on network performance.	3,4,5
Practical 13	Configure and troubleshoot DNS servers and clients. Analyze DNS queries and responses using network diagnostic tools.	2	Understand the operation of DNS and its role in translating domain names to IP addresses.	4,5,6
Practical 14	Set up email servers (SMTP, IMAP) and FTP servers. Demonstrate the exchange of emails and file transfers between clients and servers. Implement security measures such as encryption and authentication.	2	Gain practical experience in configuring and securing email and FTP services.	4,5,6
Practical 15	Deploy web servers (HTTP, HTTPS) and web applications. Implement security measures such as SSL/TLS encryption, HTTPS, and web application firewalls to protect against common attacks.	2	Understand the principles of web service security and implement security measures to protect against threats.	4,5,6

TEXT BOOKS:

T1: Data Communications and Networking–Behrouz A. Forouzan, Fourth EditionTMH,2006.

T2: Computer Networks—Andrew S Tanenbaum, 4th Edition, Pearson Education.

REFERENCE BOOKS:

R1: An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education

R2: UnderstandingcommunicationsandNetworks,3rd Edition, W.A. Shay, Cengage Learning.

R3: Computer and Communication Networks, Nader F. Mir, Pearson Education

R4: Computer Networking: A Top-Down Approach Featuring the Internet, James F. Kurose, K. W. Ross,3rd Edition, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.	PO1, PO2, PO3, PO4, PO5, PO12
2	Analyze data link layer techniques, flow control, and error protocols.	PO1, PO2, PO3, PO4, PO5, PO12
3	Analyse network layer protocols along with routing issues.	PO1, PO2, PO3, PO4, PO5, PO12
4	Summarize transport and application layer operations and protocols along with QoS services.	PO1, PO2, PO3, PO4, PO5, PO12
5	Design and execute computer network programming projects, showcasing advanced skills in network application development.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS3101R	**Data Communication and Computer Network**	**CO1**	3	1	2	1	1							3
		CO2	3	2	3	1	1							3
		CO3	3	2	3	1	1							3
		CO4	3	2	3	1	2							3
		CO5	3	3	3	3	2					1		3

SEMESTER – V									
Course Title	**Signals and Systems**								
Course code	**24BTCS3102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Introduces students for an intermediate level of fluency with signals and systems in both continuous time and discrete time. 2. Prepares students for more advanced subjects in digital signal processing (including audio, image and video processing). 3. Prepares students in communication theory, and system theory, control and robotics.								
CO1	Understand different types of signals-continuous and discrete, odd and even, periodic and aperiodic etc.								
CO2	Understand Classifications of the standard forms of Signals with respect to systems based on their properties.								
CO3	Analyse the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.								
CO4	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.								
CO5	Analyzation of Z-transform in continuous time signals and systems by using appropriate mathematical tools.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Signal and System:** signal and system as seen in everyday life, signal properties, periodicity, determinism and stochastic characteristics, special signal, system properties.	**7**	To Explain different types of signals-continuous and discrete, odd and even, periodic and aperiodic etc.	1,2
II	**Behaviour of continuous and discrete time LTI system:** Impulse response, step response, convolution, input-output behaviour, Characterization of causality and stability of LTI system.	**8**	Classifications of the standard forms of Signals with respect to systems based on their properties	1,2
III	**Laplace Transform:** Laplace Transform and its inverse: Definition, existence conditions, Region of Convergence and properties, Application of Laplace transform for the analysis of continuous time LTI system (stability etc.) Significance of poles & zeros.	**10**	Analysing the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.	1,2
IV	**Fourier Analysis of Continuous and discrete Time Signals and Systems:** Fourier Series, Fourier Transform and properties, Parseval‟s theorem, Frequency response of LTI systems. Sampling Theorem.	**12**	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.	1,2
V	**Z-Transform:** Z-Transform and its inverse: Definition, Existence Region	**8**	Analyzation of Z-transform in	1,2

	of convergence and properties. Application of Z-Transform for the analysis of Discrete time LTI systems, Significance of poles and zeros.		continuous time signals and systems by using appropriate mathematical tools.	
Practical Component				
Practical 1	**Introduction to Signals and Their Properties:**Generate and analyze basic signals (e.g., sinusoidal, exponential, and unit step signals) using MATLAB/Python. Study their properties such as periodicity, determinism, and energy/power.	**3**	Understand and classify different types of signals and their properties.	**1,2,3**
Practical 2	**Impulse and Step Response of LTI Systems:** Determine the impulse and step response of a continuous-time LTI system using MATLAB/Python.	**3**	Analyze the behavior of LTI systems using impulse and step responses.	**3,4**
Practical 3	**Convolution of Signals:** Perform convolution of two continuous-time or discrete-time signals using MATLAB/Python to compute the system's output.	**3**	Understand and apply the convolution operation to analyze input-output relationships in LTI systems.	**3,4**
Practical 4	**Stability and Causality of LTI Systems:** Verify the stability and causality of an LTI system using its impulse response and step response.	**3**	Determine the stability and causality of LTI systems based on input-output characteristics.	**3,4**
Practical 5	**Laplace Transform and System Analysis:** Compute the Laplace transform of continuous-time signals and analyze the stability of an LTI system using poles and zeros.	**3**	Apply Laplace transform techniques to analyze the stability of continuous-time systems.	**3,5**
Practical 6	**Fourier Series Representation of Periodic Signals:** Compute the Fourier series coefficients of a periodic signal and reconstruct the signal using MATLAB/Python.	**3**	Analyze and represent periodic signals using Fourier series.	3,4
Practical 7	**Fourier Transform of Continuous-Time Signals:** Compute the Fourier transform of given continuous-time signals and analyze their frequency components.	**3**	Understand the frequency domain representation of signals using Fourier transform.	**3,4**
Practical 8	**Sampling Theorem and Reconstruction of Signals:** Demonstrate the sampling theorem by sampling a continuous-time signal at different rates and reconstructing it using MATLAB/Python.	**3**	Verify the sampling theorem and understand its significance in signal processing.	**3,5**
Practical 9	**Z-Transform and System Analysis:** Compute the Z-transform of discrete-time signals and analyze the stability	**3**	Apply Z-transform techniques for analyzing discrete-time LTI systems.	**3,4**

	of a discrete-time LTI system using poles and zeros			
Practical 10	**Frequency Response of LTI Systems:** Analyze the frequency response of continuous and discrete-time LTI systems using MATLAB/Python.	**3**	Evaluate the frequency response of LTI systems and interpret system behavior in the frequency domain.	**4,5**

TEXTBOOKS:

T1: Oppenheim Alan, V., Willsky Alan. S., and Nawab, H., "Signals and Systems", Prentice Hall, 1997.

REFERENCE BOOKS:

R1: Haykin Simon, "Communication Systems", 3rd Edition, John Wiley, 1995.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.	PO1, PO2, PO3, PO4, PO5, PO12
2	Understand Classifications of the standard forms of Signals with respect to systems based on their properties.	PO1, PO2, PO3, PO4, PO5, PO12
3	Analyse the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.	PO1, PO2, PO3, PO4, PO5, PO12
4	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.	PO1, PO2, PO3, PO4, PO5, PO12
5	Analyzation of Z-transform in continuous time signals and systems by using appropriate mathematical tools.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BCSM3102R	Signals and systems	CO1	3	1	2	1	1							3
		CO2	3	2	3	1	1							3
		CO3	3	2	3	1	1							3
		CO4	3	2	3	1	2							3
		CO5	3	3	3	3	2					1		3

SEMESTER – V									
Course Title	**Cloud Computing**								
Course code	**24BTCS3103R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**4**	**0**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Programming Skills	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. To provide an overview of an exciting field of Cloud Computing 2. To introduce tools requires building, deploying, running and managing applications on a cloud platform. 3. To develop the cloud application development skills, such as Python, REST architecture, JSON, Cloud Foundry and DevOps services 4. To enable students to have skills that will help them to solve complex real-world problems in decision support.								
CO1	Understand fundamental cloud computing concepts and architectures.								
CO2	Analyze and implement RESTful APIs and data services on cloud platforms.								
CO3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.								
CO4	Develop and deploy applications using Python and related frameworks.								
CO5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Cloud Computing and IBM Cloud:** Defining cloud computing, Factors leading to cloud adoption, Types of cloud services (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid) Overview of IBM Cloud Creating an IBM Cloud account, Navigating the IBM Cloud dashboard.	**8**	Understand cloud computing concepts, service models, deployment types, and IBM Cloud basics, including account creation and dashboard navigation.	**1, 2**
II	**IBM Cloud Core Services or data services on IBM Cloud:** Watson AI services on IBM Cloud, Introduction to DevOps, Implementing DevOps practices on IBM Cloud, Using IBM Cloud toolchains for DevOps, REST API, Types and capabilities of databases, Apis interaction with Cloudant database.	**8**	Develop proficiency in IBM Cloud core services, Watson AI, DevOps practices, and interacting with databases and APIs on IBM Cloud.	**1, 2, 3**
III	**Application Development on IBM Cloud:** Understanding business problems and goals, Functional and non-functional requirements, IBM Cloud App ID, Container orchestration (Kubernetes), Pods,	**9**	Develop skills in application development on IBM Cloud, including requirements analysis, App ID, Kubernetes,	**3, 4**

	Deployment, and Service, Cloud Foundry.		and Cloud Foundry deployment.	
IV	**Developing Python Applications for the Cloud:** Building Python applications for cloud deployment Integrating popular Python frameworks (Django, Flask) with cloud services Leveraging cloud storage, databases, and AI services with Python Implementing scalability and high availability for Python applications on the cloud	**10**	Develop proficiency in building scalable and high-availability Python applications using frameworks and cloud services.	**2, 3**
V	**Security and Compliance on IBM Cloud:** Overview of IBM Cloud security features, Implementing security best practices on IBM Cloud, Ensuring compliance on IBM Cloud, OAuth protocol.	**10**	Implement IBM Cloud security features, apply best practices, ensure compliance, and use OAuth protocol for secure cloud deployments.	**3, 4, 5**

TEXT BOOKS:

T1: Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood.

T2: Cloud Computing: A Hands-On Approach by Arshdeep Bahga and Vijay Madisetti.

T3: Cloud Native Architectures: Design High-availability and Cost-effective Applications for the Cloud by Tom Laszewski, Kamal Arora, and Erik Farr.

REFERENCE BOOKS:

R1: Cloud Computing Principles and Paradigms by Rajkumar Buyya, James Broberg, and Andrzej Goscinski.

R2: Cloud Computing: Principles, Systems and Applications by Nick Antonopoulos and Lee Gillam..

R3: Data Mining: Concepts and Techniques", Third Edition, 2 Jiawei Han, Micheline Kamber and Jian Pei, ISBN 0123814790,(2011).

R4: Cloud Computing: From Beginning to End by Ray J. Rafaels.

OTHER LEARNING RESOURCES:

O1: Coursera provides several courses like "Cloud Computing Specialization", "AWS Fundamentals Specialization" and "Google Cloud Platform Fundamentals: Core Infrastructure"

O2: NPTEL provides several courses like "Cloud Computing" and "Cloud Computing and Distributed Systems"

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand fundamental cloud computing concepts and architectures.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Analyze and implement RESTful APIs and data services on cloud platforms.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Develop and deploy applications using Python and related frameworks.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS3103R	**Cloud Computing**	**CO1**	3	3	3	2	3					1		3
		CO2	2	2	2	2	2					1		2
		CO3	3	2	3	2	3					1		3
		CO4	3	3	3	2	3					1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – V									
Course Title	**Software Engineering**								
Course code	**24BTCS3104R**	**Total credits: 4** **Total hours: 36T + 30P**	**L** **3**	**T** **0**	**P** **2**	**S** **0**	**R** **0**	**O/F** **0**	**C** **4**
Pre-requisite	Basic Computer Science	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives (Minimum 3)	1. Master software development methodologies. 2. Design scalable and maintainable software architectures. 3. Implement effective software testing and quality assurance practices.								
CO1	Demonstrate ethical software development practices.								
CO2	Apply systems development lifecycle phases effectively.								
CO3	Elicit, analyze, and specify software requirements collaboratively.								
CO4	Create and evaluate standard procedures and documentation.								
CO5	Collaborate productively in interdisciplinary software project teams.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Introduction: Lifecycle models Function-oriented software design: Structured analysis and structured design.	**7**	Students will gain proficiency in life cycle models and function-oriented software design through structured analysis and design methodologies, preparing for effective software development practices.	**1,2**
II	Software requirements, analysis, and specification: Informal and formal specification.	**8**	Develop skills in software requirements, analysis, and specification, encompassing both informal and formal specification methods to ensure comprehensive software development understanding and practice.	**1,2,3**
III	Object-Oriented Design: User interface design, GUI design primitives, Window management system, and the X Windows system. Coding and Testing: Coding standards and unit testing.	**7**	Master Object-Oriented Design principles including user interface and GUI design primitives, along with Window management systems like X Windows. Understand coding standards and practice effective unit testing methods to ensure robust software development.	**1,2,3**
IV	Project management: Estimation, scheduling, risk management, and configuration management.	**8**	Acquire proficiency in project management essentials such as estimation, scheduling,	**1,2,3**

			risk management, and configuration management, crucial for effective planning and execution of software projects.	
V	Software reliability and quality assurance: Reliability metrics and growth modeling, ISO-9000, SEI, and CMM.	**6**	Explore software reliability and quality assurance, focusing on reliability metrics, growth modelling, and standards like ISO-9000, SEI, and CMM to ensure high-quality software development practices and processes.	**1,2**
Practical Component				
Practical 1	Compare and analyse various SDLC models (e.g., WaterSummer, Agile, Spiral) for different project scenarios.	**3**	Understand the features, advantages, and limitations of different lifecycle models.	1,2,4
Practical 2	Develop a Data Flow Diagram (DFD) for a given system to represent its functional components and data flow.	**3**	Apply structured analysis techniques to create functional models for software systems.	3,4
Practical 3	Prepare a formal SRS document for a given problem statement using IEEE standards.	**3**	Write clear, concise, and comprehensive software requirement specifications.	3,5
Practical 4	Create UML diagrams (class, sequence, activity, and use case diagrams) for a software system.	**3**	Understand and apply object-oriented design principles using UML.	3,4
Practical 5	Design and develop a user-friendly graphical user interface (GUI) for a given application using GUI primitives.	**3**	Design intuitive and accessible user interfaces for software systems.	3,4
Practical 6	Write code following industry-standard coding guidelines and perform unit testing using tools like JUnit or PyTest.	**3**	Develop high-quality code adhering to coding standards and validate it using unit tests.	3,5
Practical 7	Estimate project effort and duration using techniques like Function Point Analysis (FPA) and create a Gantt chart for scheduling.	**3**	Apply estimation and scheduling techniques to plan software projects effectively.	4,5
Practical 8	Identify potential risks in a software project, assess their impact, and develop a risk mitigation plan.	**3**	Understand risk management processes and apply them to reduce project uncertainties.	3,4
Practical 9	Implement quality assurance techniques and explore ISO-9000 standards and SEI CMM levels for software quality management.	**3**	Demonstrate knowledge of software quality standards and assurance techniques.	2,4
Practical 10	Calculate reliability metrics (MTBF, MTTR) and use a reliability growth model to assess software reliability.	**3**	Apply reliability metrics to evaluate and improve software robustness.	4,5

TEXT BOOKS:

T1: An Integrated Approach to Software Engineering by Jalote. Narosa Publishing House.
T2: Software Engineering by R. Mall, PHI.

REFERENCE BOOKS:

R1: Software Engineering by R.S.Pressman, McGraw Hill

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate ethical software development practices.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
2	Apply systems development lifecycle phases effectively.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
3	Elicit, analyze, and specify software requirements collaboratively.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
4	Create and evaluate standard procedures and documentation.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
5	Collaborate productively in interdisciplinary software project teams.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS3104R	Software Engineering	CO1	2	2	3	2	2				3		3	3
		CO2	3	2	3	2	2				2		3	3
		CO3	3	2	3	3	3				3		3	2
		CO4	3	2	3	1	3				2		3	2
		CO5	3	2	3	3	3				2		2	2

SEMESTER – V									
Course Title	**PE-II: Data Mining and Analytics**								
Course code	**24BTCS3105R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/I semester of the third year of the programme**								
Course Objectives (Minimum 3)	This course is an introduction to data mining techniques, algorithms, and applications. Covers data pre-processing, pattern discovery, clustering, classification, and association rule mining. Emphasizes practical implementation and real-world data analysis.								
CO1	Understand diverse real-life data, mastering statistical foundations, and identifying challenges in data mining.								
CO2	Utilize Association Rule Mining to address practical problems, demonstrating understanding and proficiency in application.								
CO3	Apply Cluster analysis in unsupervised learning to real-world datasets, showcasing competence in data clustering techniques.								
CO4	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.								
CO5	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Data Mining:** What is data mining, applications of data mining, issues and challenges faced in data mining, types of data attributes – nominal, binary, ordinal, numeric, discrete and continuous, basic statistical descriptions of data – measures of central tendency and dispersion, major tasks in data pre-processing	**8**	Learning Outcome: Understand data mining concepts, applications, challenges, data attributes, statistical descriptions, and data pre-processing tasks for effective data analysis and decision-making.	**1, 2**
II	**Association Rule Mining:** Definitions of frequent itemset, maximal frequent itemset, closed frequent itemset, association rules, Frequent itemset generation – Apriori algorithm, DIC algorithm and FP-growth algorithm, Generating association rules from frequent item sets	**9**	Learning Outcome: Master association rule mining concepts including frequent itemsets, algorithms (Apriori, DIC, FP-growth), and generating association rules for data pattern discovery.	**3,4**
III	**DataClustering:** what is cluster analysis, need for cluster analysis, data matrix, dissimilarity matrix, proximity and dissimilarity measures for different data attribute types, partitioning based clustering methods, density-based clustering methods and hierarchical clustering methods, measuring cluster quality	**10**	Understand cluster analysis principles, dissimilarity measures, and various clustering methods for effective data segmentation and cluster quality evaluation.	**1, 2**
IV	**Prediction**: What is prediction, application	**8**	Master prediction concepts, including	**2, 3**

	areas of prediction, Simple linear regression, Multiple linear regression, Predictor Error measures, Common techniques for assessing accuracy – holdout, random sampling, cross-validation, bootstrap		regression techniques, predictor error measures, and accuracy assessment methods for effective predictive modeling and decision-making.	
V	**Classification**: what is classification, differences between classification and prediction, applications of classification, Some classification algorithms - Decision Trees, Naïve Bayes Classification, Logistic Regression and K-nearest Neighbour Classification, Confusion matrix and metrics for evaluating classifier performance, Overview of ensemble methods – Bagging and Boosting, Ensemble-based classifier - Random Forest	**10**	Understand classification principles, algorithms (Decision Trees, Naïve Bayes, Logistic Regression, K-nearest Neighbor), evaluation metrics, and ensemble methods for effective data classification.	**2, 3, 4**
Practical Component				
Practical 1	Explore what data mining is and its applications.	**3**	Understand data mining and its applications, foundational for data analysis.	**2**
Practical 2	Identify and classify types of data attributes (nominal, binary, ordinal, numeric, discrete, continuous).	**3**	Recognize various data attributes, crucial for data preprocessing.	**2**
Practical 3	Calculate measures of central tendency and dispersion for a dataset.	**3**	Apply statistical measures to summarize data, essential for data analysis.	**3**
Practical 4	Perform major tasks in data preprocessing, including handling missing data and normalization.	**3**	Execute data preprocessing techniques, key for data quality improvement.	**3**
Practical 5	Implement the Apriori algorithm to generate frequent itemsets and association rules.	**3**	Apply Apriori algorithm for association rule mining, critical for pattern discovery.	**3**
Practical 6	Use FP-Growth algorithm to generate frequent itemsets.	**3**	Develop frequent itemsets using FP-Growth, important for efficient pattern mining.	**3**
Practical 7	Perform K-means clustering on a dataset and analyze clusters.	**3**	Apply K-means clustering, essential for grouping similar data points.	**3**
Practical 8	Implement hierarchical clustering and analyze the resulting dendrogram.	**3**	Utilize hierarchical clustering to understand data structure, crucial for unsupervised learning.	**3**

Practical 9	Implement simple and multiple linear regression models for prediction.	**3**	Develop predictive models using linear regression, fundamental for forecasting.	**3**
Practical 10	Build a decision tree classifier and evaluate its performance.	**3**	Create classification models with decision trees, key for categorical data analysis.	**3**

TEXT BOOKS:

T1: Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, India

T2: AK Pujari, Data Mining Techniques, University Press, India

REFERENCE BOOKS:

R1: Han, Manilla and Smyth, Principles of Data Mining, PHI, India

R2: Tan, Steinbach,Vipin Kumar, Introduction to Data Mining, Pearson Education Inc.

OTHER LEARNING RESOURCES:

O1: Coursera course like "Data Mining Specialization" covers fundamental data mining concepts, techniques, and applications.

O2: edX course like "Data Mining for Business Analytics" covers data mining techniques for business analytics, including data preprocessing, classification, prediction, clustering, association rules, and text mining.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand diverse real-life data, mastering statistical foundations, and identifying challenges in data mining.	PO1, PO2, PO3, PO4, PO10, PO12
2	Utilize Association Rule Mining to address practical problems, demonstrating understanding and proficiency in application.	PO1, PO2, PO3, PO4, PO10, PO12
3	Apply Cluster analysis in unsupervised learning to real-world datasets, showcasing competence in data clustering techniques.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.	PO1, PO2, PO3, PO4, PO10, PO12
5	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS3105R	Data Mining and Analytics	CO1	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – V									
Course Title	**PE-II: Information Retrieval Systems**								
Course code	**24BTCS3105R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives (Minimum 3)	This course provides an in-depth introduction to the principles and practices of Information Retrieval (IR). Students will learn about the theory behind IR systems, the algorithms and techniques used to implement them, and the evaluation methods to measure their effectiveness. Topics covered include text processing, indexing, querying, ranking, and various IR models.								
CO1	Understand the fundamental concepts of Information Retrieval.								
CO2	Learn about various indexing and search techniques.								
CO3	Explore different IR models and their applications.								
CO4	Study the methods of evaluating IR systems.								
CO5	Gain hands-on experience with IR system implementation.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Information Retrieval** Course overview and objectives History of IR systems Applications of IR	8	Understand the fundamentals of information retrieval systems, including their history, applications, and objectives for effective information management and retrieval.	1, 2, 3
II	**Text Processing and Indexing** Text preprocessing: tokenization, stemming, and lemmatization Stopword removal and handling special characters Inverted index and its construction Index compression techniques	8	Master text processing techniques including tokenization, stemming, lemmatization, stopwords removal, inverted index construction, and index compression for efficient information retrieval systems.	2, 3, 4
III	**Retrieval Models** Boolean retrieval model Vector space model Probabilistic retrieval models	10	Understand retrieval models including Boolean, vector space, and probabilistic models for efficient information retrieval and relevance ranking in various contexts.	2, 3
IV	**Relevance Feedback and Query Expansion** Relevance feedback mechanisms Pseudo-relevance feedback Query expansion techniques	10	Master relevance feedback mechanisms, pseudo-relevance feedback, and query expansion techniques to enhance information retrieval precision and relevance.	3, 4, 5

V	**Evaluation of IR Systems** Evaluation metrics: precision, recall, F-measure, MAP, NDCG Test collections: TREC, Cranfield paradigm User studies and A/B testing	9	Evaluate information retrieval systems using metrics like precision, recall, F-measure, MAP, NDCG, test collections, and user studies for system improvement.	4, 5
Practical Component				
Practical 1	Review course objectives and explore the history of Information Retrieval (IR) systems.	3	Understand the fundamentals and evolution of IR systems.	2
Practical 2	Investigate various applications of IR in different fields.	3	Recognize the diverse applications of IR systems.	2
Practical 3	Implement tokenization, stemming, and lemmatization on a text dataset.	3	Apply text preprocessing techniques to prepare data for indexing.	3
Practical 4	Remove stopwords and handle special characters in a text dataset.	3	Clean text data by removing irrelevant components.	6
Practical 5	Construct an inverted index for a given text corpus.	3	Develop an inverted index to facilitate efficient search and retrieval.	3
Practical 6	Implement basic index compression techniques on an inverted index.	3	Compress index data to optimize storage and retrieval efficiency.	3
Practical 7	Implement Boolean, Vector Space, and Probabilistic retrieval models.	3	Apply various retrieval models to query and retrieve relevant documents.	3
Practical 8	Implement relevance feedback mechanisms and query expansion techniques.	3	Enhance search results using relevance feedback and query expansion.	3
Practical 9	Calculate precision, recall, F-measure, MAP, and NDCG for an IR system.	3	Evaluate the performance of IR systems using standard metrics.	5
Practical 10	Conduct user studies and perform A/B testing for an IR system.	3	Assess IR system effectiveness through user studies and A/B testing.	5

TEXT BOOKS:

T1: "Introduction to Information Retrieval" by Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze.

REFERENCE BOOKS:

R1: "Modern Information Retrieval: The Concepts and Technology behind Search" by Ricardo Baeza-Yates and Berthier Ribeiro-Neto.

OTHER LEARNING RESOURCES:

O1: TREC (Text REtrieval Conference) datasets and proceedings

O2: Online tutorials and courses (e.g., Coursera, edX)

O3: Software tools and libraries (e.g., Apache Lucene, Solr, Elasticsearch)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental concepts of Information Retrieval.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Learn about various indexing and search techniques.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Explore different IR models and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Study the methods of evaluating IR systems.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Gain hands-on experience with IR system implementation.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS3105R	**Information Retrieval Systems**	**CO1**	2	2	1	1	1					1		2
		CO2	3	3	3	3	2					2		3
		CO3	3	2	3	3	3					2		3
		CO4	2	2	2	2	2					2		2
		CO5	2	2	3	2	2					2		3

SEMESTER – V									
Course Title	**Design Thinking and Entrepreneurship**								
Course code	**24BTCS3106R**	**Total credits: 2** **Total hours: 36T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer / I semester of the third year of the programme**								
Course Objectives (Minimum 3)	1. To introduce students to the principles and processes of design thinking for solving real-world problems empathetically and innovatively. 2. To enable students to conduct contextualized research and define user-centric needs. 3. To develop creative problem-solving skills through brainstorming, idea evaluation, and iterative prototyping. 4. To foster entrepreneurial mindsets by integrating innovation strategies into the design and development processes. 5. To equip students with the ability to test, refine, and launch user-centric solutions effectively.								
CO1	Compare and select problems suitable for DT projects and use techniques for empathetic research.								
CO2	Identify and document insights, user habits and identify user needs.								
CO3	Visualize solutions, evaluate solution concepts and able to create rough prototypes, gather feedback.								
CO4	Able to create high-fidelity prototypes. Able to test user experience								
CO5	Able to identify a business model for a solution concept. Able to estimate financial results2. Apply the various techniques of parsing to construct a syntax analyzer for a specific programming language.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Design Thinking for Contextualized Problem-Solving and Empathetic Research.	8	Master design thinking for empathetic research and contextualized problem-solving, enhancing innovation and user-centered solutions.	1,2
II	Analysis of Research, Defining Needs	8	Gain proficiency in analyzing research data to accurately define needs, facilitating effective problem-solving and decision-making processes.	2,3,4
III	Brainstorming and Evaluation,	8	Develop skills in brainstorming and evaluating ideas to foster creativity and critical thinking for effective decision-making and problem-solving.	2,5
IV	Prototyping, Testing and testing to Launch.	8	Students will acquire proficiency in prototyping, testing, and launching processes to validate ideas and products efficiently, ensuring successful implementation and user satisfaction.	3,4,5
V	Entrepreneurial Innovation.	8	Develop entrepreneurial skills and strategies for fostering innovation, enabling effective identification, development, and implementation of business opportunities.	4,5

TEXT BOOKS:

T1: Ries, E. - "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses", Crown Business, 2011.

REFERENCE BOOKS:

R1: Van Der Pijl, P., Lokitz, J., & Solomon, L. K. (2016). Design a better business: New tools, skills, and mindset for strategy and innovation. John Wiley & Sons.

R2: Kelley, T., & Kelley, D. - "Creative Confidence: Unleashing the Creative Potential Within Us All", Crown Business, 2013.

OTHER LEARNING RESOURCES:

O1: "Design Thinking for Innovation" by the University of Virginia (Coursera).

O2: "Entrepreneurship 10"1by MIT OpenCourseWare.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Compare and select problems suitable for DT projects and use techniques for empathetic research.	PO1, PO2, PO3, PO4, PO10, PO12
2	Identify and document insights, user habits and identify user needs.	PO1, PO2, PO3, PO4, PO10, PO12
3	Visualise solutions, evaluate solution concepts and able to create rough prototypes, gather feedback.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Able to create high-fidelity prototypes. Able to test user experience	PO1, PO2, PO3, PO4, PO10, PO12
5	Able to identify a business model for a solution concept. Able to estimate financial results2. Apply the various techniques of parsing to construct a syntax analyzer for a specific programming language.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**

24BTCS3106R	**Design Thinking**	**CO1**		2	2	3		1		3	1	1		2
		CO2		3	2	3		1		2	1	1		2
		CO3		3	2	2		1		2	1	1		2
		CO4		2	3	2		1		2	1	1		2
		CO5		2	3	3		1		3	1	1		2

SEMESTER - V									
Course Title	**Essence of Indian Traditional Knowledge**								
Course code	**24BTCS3107R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives	1. To introduce the concept and relevance of Indian traditional knowledge in various domains of life. 2. To explore the foundational Indian philosophical traditions and their influence on ethics and values. 3. To study the contributions of traditional Indian knowledge to science, technology, and sustainability. 4. To appreciate the cultural heritage and traditional art forms of India as a source of knowledge and creativity. 5. To promote the application of traditional knowledge systems for holistic living and addressing modern challenges.								
CO1	Understand the scope and importance of Indian traditional knowledge in contemporary contexts.								
CO2	Analyze Indian philosophical traditions and their relevance to ethics and value systems.								
CO3	Examine the contributions of traditional knowledge systems to science, technology, and environmental conservation.								
CO4	Appreciate the role of cultural heritage and art forms in shaping India's identity and global recognition.								
CO5	Apply principles of Indian traditional knowledge to promote sustainable and holistic living.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	- Definition and importance of traditional knowledge. - Ancient systems of knowledge in India: Vedic, Upanishadic, and other ancient texts. - Scope and relevance of traditional knowledge in contemporary society.	**7**	Understand the concept and relevance of Indian traditional knowledge in various domains of life.	**1,2,3**
II	- Introduction to six schools of Indian philosophy (Shad-Darshanas). - Key ideas of Advaita Vedanta, Sankhya, and Yoga. - Influence of Indian philosophy on ethics and values in modern life.	**7**	Understand the fundamental Indian philosophical schools and their contributions to knowledge systems.	**2,3,4**
III	- Ancient Indian contributions to mathematics, astronomy, and medicine (Ayurveda). - Traditional agricultural practices and environmental conservation techniques.	**7**	Explore the contributions of traditional Indian knowledge to science, technology, and environmental sustainability.	**3,4**

	- Integration of traditional and modern science for sustainable development.			
IV	- Indian classical music and dance (Bharatanatyam, Kathak, etc.). - Traditional crafts, architecture, and their socio-economic significance. - Role of cultural heritage in fostering national identity and global recognition.	**7**	Appreciate Indian cultural heritage and art forms as expressions of traditional knowledge and creativity.	**3,4**
V	- Yoga and Ayurveda for physical and mental well-being. - Ethical principles and sustainability in traditional Indian practices. - Relevance of traditional knowledge in solving modern-day problems.	**7**	Understand the role of traditional knowledge in promoting a holistic and balanced way of life.	**3,5**

Textbooks

T1: Kapil Kapoor & Michel Danino - "Knowledge Traditions and Practices of India", CBSE, 2012.

T2: Dayalbagh Educational Institute - "Essence of Indian Knowledge Tradition (EIKT)", New Age International Publishers, 2019.

Reference Books

R1: Satya Prakash et al. - "Foundations of Indian Culture", Bharatiya Vidya Bhavan, 1994.

R2: M. Hiriyanna - "Essentials of Indian Philosophy", Motilal Banarsidass Publishers, 2000.

Other Learning Resources

O1: NPTEL Course: "Introduction to Indian Knowledge Systems" by Prof. B. Mahadevan, IIM Bangalore.

O2: SWAYAM Course: "Understanding Indian Culture" by IGNOU.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the scope and importance of Indian traditional knowledge in contemporary contexts.	PO6, PO12
2	Analyze Indian philosophical traditions and their relevance to ethics and value systems.	PO6, PO12
3	Examine the contributions of traditional knowledge systems to science, technology, and environmental conservation.	PO6, PO12
4	Appreciate the role of cultural heritage and art forms in shaping India's identity and global recognition.	PO6, PO12
5	Apply principles of Indian traditional knowledge to promote sustainable and holistic living.	PO6, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS3107R	**Essence of Indian Traditional Knowledge**	**CO1**						3						2
		CO2						3						2
		CO3						2						2
		CO4						2						1
		CO5						2						1

<table>
<tr><th colspan="9">SEMESTER – V</th></tr>
<tr><td>Course Title</td><td colspan="8">Mini Project III</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BTCS3108R</td><td>Total credits: 1</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>Total hours: 48</td><td>0</td><td>0</td><td>0</td><td>4</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="6">Nil</td></tr>
<tr><td>PROGRAMME</td><td colspan="8">Bachelor of Technology in Computer Science and Engineering</td></tr>
<tr><td>Semester</td><td colspan="8">Summer/I semester of the third year of the Programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="8">1. To develop the knowledge, skills and attitudes of a professional Computer engineering professional.
2. To become confident in designing engineering solutions to complex software problems utilising a systems approach.</td></tr>
<tr><td>CO1</td><td colspan="8">Apply IT principles for real-world problem-solving using programming languages and network solutions.</td></tr>
<tr><td>CO2</td><td colspan="8">Contribute to complex projects as a team member, demonstrating effective personal and team management.</td></tr>
<tr><td>CO3</td><td colspan="8">Develop programming skills to create and assess software, hardware, and network solutions.</td></tr>
<tr><td>CO4</td><td colspan="8">Apply personal and team management skills as a professional software developer.</td></tr>
<tr><td>CO5</td><td colspan="8">Cultivate employability skills and uphold professionalism in software development.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="3">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
• theoretical studies
• computer simulations
• hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;
5. Prepare a formal report describing the work undertaken and results obtained so far.
6. Present the work in a forum involving poster presentations and</td><td>48</td><td colspan="3">Conduct a research project involving literature review, technical work, progress reporting, seminar delivery, and formal presentation under academic supervision.</td><td>3, 4, 5, 6</td></tr>
</table>

	demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BTCS3108R	**Mini Project III**	**CO1**	2	2	2	1	2	2				2		2
		CO2	2	2	2	1	2	3				2		2
		CO3	1	1	2	1	2	2				3		3
		CO4	2	2	2	1	2	2				2		2
		CO5	1	1	1	1	2	1				1		3

SEMESTER – VI									
Course Title	**Web Technologies**								
Course code	**24BTCS3201R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. To make a student familiar with client server architecture. 2. To teach developing a web application using various technologies. 3. To make the students learn the skills and project-based experience needed for entry into web application and development careers..								
CO1	Learn about Internet and Web Page Design.								
CO2	Understand detailed concepts on Web Browsers, Markup Language Basics and XML.								
CO3	Learn basic concepts of client side programming								
CO4	Learn detailed concepts on Web Server Side technologies.								
CO5	Understand few advanced Web Technologies and the Web Security.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**An Introduction to Web Technology** History of web Development, Time line, Motivation, Categories of Web Applications, Characteristics of Web Applications. Evolution and Need for Web Engineering, Web Engineering Models, Software Engineering v/s Web Engineering . World Wide Web: Introduction to TCP/IP and WAP, DNS, Email, TelNet, HTTP and FTP. Introduction to Browser and search engines, Search fundamentals, Search strategies, Directories search engines and Meta search engines, Working of the search engines , Miscellaneous Web Browser details, Introduction to Web Servers: Features of web servers, caching, case study-IIS, Apache, Configuring web servers	**8**	To understand web development basics, HTML, CSS, JavaScript, client-server architecture, and create simple web applications.	**1,2,3**
II	**Technologies for Web Applications** HTML and DHTML, HTML Basic Concepts, Static and dynamic HTML, Structure of HTML documents, HTML Elements, Linking in HTML, Anchor Attributes, Image Maps, Meta Information, Image Preliminaries, Layouts, Backgrounds, Colors and Text, Fonts, Tables, Frames and layers, Audio and Video Support with HTML Database integration, CSS, Positioning with Style sheets, Forms Control, Form. Elements. Introduction to CGI PERL, JAVA SCRIPT, ASP, Cookies Creating and Reading Cookies. Introduction of XML, Validation of XML documents, DTD, Ways to use XML, XML for data files, HTML Vs XML, Embedding XML into HTML documents, Converting XML to HTML for Display,	**10**	To master the advanced web technologies, frameworks, APIs, databases, and develop complex, scalable web applications with robust functionality.	**3,4**

	Displaying XML using CSS and XSL, Rewriting HTML as XML, Relationship between HTML, SGML and XML			
III	**Client Side Programming** Document Object Model (DOM),Overview of DOM, Jquery: ,Element Selector Document ready function, Events ,Ajax call ,plugins: Using JQuery UI ,Unobtrusive client validation ,JQuery templates ,Feature detection: Browser detection ,Feature detection ,Modernizer , polyfills. Introduction to AJAX, Overview of Traditional Web Communication Processes and Technologies, Interacting with the Web Server Using the XMLHttpRequest , Create an XMLHttpRequest Object Interact with the Web Server Working with PHP and AJAX: Introduction Process Client Requests Accessing Files Using PHP Applying Basic AJAX Techniques: Introduction Download Images Using AJAX Auto, Populate Select Boxes, Implementing Security and Accessibility in AJAX Applications: production, Secure AJAX Applications Accessible Rich Internet Applications	**10**	To learn JavaScript, DOM manipulation, event handling, form validation, AJAX, and create interactive, responsive web interfaces.	**3,4**
IV	**Server Side Programming** J2SE:Data Types, Arrays, Type Casting, Classes and Objects, Inheritance, Interfaces, Exception Handling, Multithreading,J2EE as a framework, Client Server Traditional model, Comparison amongst 2-tier, 3-tier and N-tier Architectures, Thin and Thick Clients. J2EE Servlet 2.x Specification, Writing small Servlet Programs, Deployment Descriptor, Inter Servlet Collaboration, Session: Definition, State on web, Different ways to track sessions, JSP Technology Introduction-JSP and Servlets-Running JSP Applications Basic JSP- JavaBeans Classes - Support for the Model- View-Controller Paradigm- Case Study- Related Technologies.	**9**	To understand server-side scripting, databases, authentication, RESTful APIs, server deployment, and build dynamic, data-driven web applications.	**4**
V	**Advanced Web Technologies and Web Security** Exposure to Advanced Web Technologies: Distributed Object based models- DCOM, CORBA, EJB; Web services and Related Technologies- ISAPI, SOAP, UDDI, WSDL; Other Advanced Web Technologies- AJAX, ISAPI, .NET. Web Security: Firewalls-definition and uses, network layer firewalls and application layer firewalls; Proxy servers, HTPS for secure web communication.	**8**	To explore advanced web frameworks, progressive web apps, web security principles, encryption, and secure web application development practices.	**4**
Practical Component				

Practical Number	Topic / Experiment	Contact Hours	Learning Outcome	BL
Practical 1	Identify common HTML tags and their functions.	2	Identify and label common HTML tags and understand their basic functions in web development.	1,2,3
Practical 2	Match CSS properties to their effects.	2	To match CSS properties to their corresponding effects, enhancing their understanding of CSS styling.	3,4
Practical 3	Create a concept map of basic JavaScript concepts.	2	To create a concept map, demonstrating their understanding of basic JavaScript concepts and their interrelationships.	3,4
Practical 4	Explain the structure of a basic web page.	2	analyze and explain the structure and purpose of different sections in a basic web page.	4,5
Practical 5	Use HTML and CSS to create a simple web page.	2	Use HTML and CSS to create a functional web page with various elements and styles.	3,4
Practical 6	Implement basic form validation using JavaScript.	2	implement JavaScript to perform basic form validation, ensuring proper data entry before submission.	4,5
Practical 7	Evaluate the responsiveness of a web page.	2	evaluate a web page's responsiveness, identifying areas for improvement using developer tools.	4,5
Practical 8	Analyze the performance of a web page.	2	To analyze web page performance, using tools to identify and suggest solutions for performance bottlenecks..	3,4
Practical 9	Conduct a usability test on a web application..	2	Conduct and analyze usability tests, making recommendations to improve web application user experience.	5
Practical 10	Review and evaluate the quality of web code.	2	Review and evaluate web code quality for readability, efficiency, and best practices adherence.	5,6
Practical 11	Create a web application using HTML, CSS, and JavaScript.	2	Design and build a dynamic web application using	4,5

			HTML, CSS, and JavaScript.	
Practical 12	Develop a custom CSS framework.	2	To develop a reusable custom CSS framework for use in various web development projects.	4,5
Practical 13	Develop a single page application using a JavaScript framework (e.g., React or Vue.js).	2	To create a SPA using a JavaScript framework, integrating dynamic content through API interactions.	3,4
Practical 14	Design and implement a RESTful API.	2	To design and implement a RESTful API using Node.js and Express, connecting it to a front-end..	5
Practical 15	Develop an interactive data dashboard.	2	develop interactive dashboards to visualize and explore data insights dynamically..	5,6

Text Books:

T1: Web Technologies: HTML, Javascript, PHP, Java, JSP, XML and AJAX, Black Book, Kogent Learning Solutions Inc., 1st Edition, 2009, Dreamtech Press.

T2: Web Technologies, Uttam K. Roy, 1st Edition, 2010, Oxford Publications.

Reference Books:

R1: N. P. Gopalan and J. Akilandeswari, Web Technology: A Developer's Perspective,1st Edition, 2014,Prentice-Hall of India Pvt. Ltd.

R2: Jeffrey C. Jackson, Web Technologies: A Computer Science Perspective, 1st Edition, 2008, Pearson Education India.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Learn about Internet and Web Page Design.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
2	Understand detailed concepts on Web Browsers, Markup Language Basics and XML.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Learn basic concepts of client side programming	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Learn detailed concepts on Web Server Side technologies.	CO1, CO2, CO3, CO4, CO5, CO10, CO12

5	Understand few advanced Web Technologies and the Web Security.	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS3201R	Web Technologies	CO1	2	2	2	2	3					2		2
		CO2	2	3	3	2	3					2		2
		CO3	2	2	2	2	3					2		2
		CO4	2	2	2	3	3					2		2
		CO5	3	3	3	3	3					2		2

SEMESTER – VI									
Course Title	**Compiler Design**								
Course code	**24BTCS3202R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**1**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	Theory of Computation	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Discuss the process of translating a high-level language to machine code required for compiler construction. 2. Analyze the software tools and techniques used in compiler construction such as lexical analyser and parser generators 3. Discuss the data structures used in compiler construction such as abstract syntax trees, symbol tables, three address code and stack machines.								
CO1	Understand different phases of a compiler and explain lexical analysis principles.								
CO2	Apply parsing techniques to build a language-specific syntax analyser.								
CO3	Analyse syntax-directed translations, evaluation order, and type checking.								
CO4	Understand the association of runtime storage allocation with control flow and procedure calls.								
CO5	Apply code optimization on intermediate code for target code generation.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Compiler structure:** Analysis-synthesis model of compilation, Various phases of a compiler, Tool based approach to compiler construction. Lexical analysis: Interface with input, Parser and symbol table, Token, Lexeme and patterns, Regular definition, Transition diagrams, LEX.	**8**	Understand the analysis-synthesis model of compilation and the role of lexical analysis in converting source code into tokens using tools like LEX.	**1,2,3**
II	**Syntax analysis:** CFG's, Ambiguity, Associativity, Precedence, Top-down parsing, Recursive descend parsing, Transformation on the grammars, Predictive parsing, Bottom-up parsing, Operator precedence grammars, LR parsers (SLR, Canonical, LALR), YACC	**8**	Grasp the concepts of context-free grammars (CFGs) and develop skills in various parsing techniques including top-down and bottom-up parsing methods, utilizing tools like YACC.	**3,4**
III	**Syntax directed translation:** Inherited and synthesized attributes, Dependency graph, Evaluation order, Bottom up evaluation of S- attributed definitions L- attributed definitions and top down translation of attributes Type checking: Type system, Type expressions, Structural and name equivalence of types, Type conversion.	**10**	Learn to design and implement syntax-directed translation schemes, focusing on attribute grammars and type checking mechanisms, and understand the differences between structural and name equivalence of types.	**3,4**
IV	**Run time environments:** Storage organization, Storage-allocation strategies, Access to nonlocal names, Activation tree,	**10**	Comprehend the organization and management of run-time storage, including	**4**

	Activation record, Parameter passing, Symbol table and dynamic storage allocation Intermediate code generation: Intermediate representations, Translation of declarations, Assignments, Control flow, Boolean expressions and procedure calls.		activation records and parameter passing techniques, and learn the principles of intermediate code generation.	
V	**Code generation:** Issues in the design of a code generator, Basic blocks and flow graphs, Next use information, Register allocation, Code generation algorithm, Dag representation of programs, Code generation from dags, Peephole optimization and code generator generators.	**9**	Acquire knowledge on the design and implementation of a code generator, emphasizing optimization techniques, basic blocks, flow graphs, and register allocation strategies.	**4**

TEXT BOOKS:

T1: A.V. Aho, R. Sethi, J.D. Ullman, "Compilers: Principles, Techniques and Tools", Addison – Wesley.

T2: Steven S. Muchnick, "Advanced Compiler Design and Implementation", Elsevier

REFERENCE BOOKS:

R1: W. Appel, "Modern Compiler Implementation in C: Basic design", Cambridge Press

R2: Fraser and Hanson, "A Retargetable C Compiler: Design and Implementation", Addison-Wesley

R3: Dhamdhere, "Compiler Construction", McMillan

OTHER LEARNING RESOURCES:

1. www.vssut.ac.in/lecture_notes/lecture1422914957.pdf
2. 2.http://csenote.weebly.com/principles-of-compiler-design.html
3. http://www.faadooengineers.com/threads/32857,Compiler-Design-Notes-full-book-pdf-download
4. https://www.vidyarthiplus.com/vp/thread,37033.html#.WF0PhlMrLDc

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand different phases of a compiler and explain lexical analysis principles.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
2	Apply parsing techniques to build a language-specific syntax analyser.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Analyse syntax-directed translations, evaluation order, and type checking.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Understand the association of runtime storage allocation with control flow and procedure calls.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
5	Apply code optimization on intermediate code for target code generation.	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS3202R	Compiler Design	CO1	3	2	1	2	2					2		3
		CO2	3	3	3	3	2					2		3
		CO3	3	3	3	3	2					2		3
		CO4	3	1	2	2	2					2		3
		CO5	3	3	3	3	2					2		3

SEMESTER – VI									
Course Title	**Design and Analysis of Algorithms**								
Course code	**24BTCS3203R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Data Structures	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Analyse the asymptotic performance of algorithms. 2. Demonstrate a familiarity with major algorithms and data structures. 3. Apply important algorithmic design paradigms and methods of analysis.								
CO1	Understand analysis of algorithms using asymptotic notations for advanced problem-solving.								
CO2	Implement sorting, searching, matrix multiplication algorithms with varied strategies.								
CO3	Apply greedy approaches and dynamic programming to optimize problem-solving, for advanced algorithm design.								
CO4	Apply problem-solving strategies to complex scenarios, solving N-Queen, knapsack, and TSP, etc.								
CO5	Evaluate decision tree bounds, interpret complexity classes P, NP, NP-Complete proficiently.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction: Characteristics of algorithm.** Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.	**8**	Understand and analyze the best, average, and worst-case complexities of algorithms.	**2,4**
II	**Fundamental Algorithmic Strategies:** Brute-Force, Greedy, Dynamic Programming, Branch and Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics –characteristics and their application domains.	**8**	Apply different algorithmic strategies such as Brute-Force, Greedy, and Dynamic Programming to solve problems.	**3,4,5**
III	**Graph and Tree Algorithms:** Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	**10**	Implement traversal algorithms and determine the shortest paths in graphs.	**3,4**
IV	**Tractable and Intractable Problems:**	**10**	Differentiate between P, NP, NP-complete, and NP-hard classes and	**2,4,5**

	Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques		understand the significance of Cook's theorem.	
V	**Advanced Topics:** Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE	**9**	Develop and analyze approximation and randomized algorithms for complex problems.	**3,4,6**

Practical Component

Practical Number	**Topic / Experiment**	**Contact Hours**	**Learning Outcome**	**BL**
Practical 1	Implement and analyze the time complexity of Bubble Sort, Selection Sort, and Insertion Sort.	**2**	Compare the time complexities of different sorting algorithms in best, average, and worst cases.	1,2,3
Practical 2	Use the substitution method to solve simple recurrence relations.	**2**	Apply the substitution method to determine the time complexity of recursive algorithms.	3,4
Practical 3	Implement a recursive algorithm and analyze its performance using the recursion tree method.	**2**	Visualize and analyze the time complexity of recursive algorithms using recursion trees.	3,4
Practical 4	Solve the 0/1 Knapsack problem using dynamic programming.	**2**	Develop and apply dynamic programming to solve optimization problems.	4,5
Practical 5	Implement a solution to the Traveling Salesman Problem using the branch and bound technique.	**2**	Apply branch and bound to solve combinatorial optimization problems.	3,4
Practical 6	Design and implement a greedy algorithm for the Fractional Knapsack problem.	**2**	Understand and apply the greedy strategy to solve optimization problems.	4,5
Practical 7	Implement backtracking to solve the N-Queens problem.	**2**	Apply backtracking techniques to solve constraint satisfaction problems.	4,5
Practical 8	Implement a brute-force solution for the Bin Packing problem and evaluate its efficiency.	**2**	Evaluate the efficiency of brute-force algorithms for combinatorial problems.	3,4
Practical 9	Implement Depth First Search (DFS) and Breadth First Search (BFS) for graph traversal.	**2**	Implement and compare DFS and BFS traversal techniques.	5
Practical 10	Implement Dijkstra's algorithm to find the shortest path in a weighted graph.	**2**	Apply shortest path algorithms to find the minimum distance between nodes in a graph.	5,6
Practical 11	Implement Kruskal's and Prim's algorithms to find the Minimum Spanning Tree (MST).	**2**	Compare and apply different algorithms to find MST in a graph.	4,5

Practical 12	Perform topological sorting on a directed acyclic graph (DAG).	**2**	Understand and implement topological sorting on DAGs.	4,5
Practical 13	Implement the reduction technique to show that a problem is NP-complete.	**2**	Demonstrate the reduction technique to classify problems within NP-complete.	3,4
Practical 14	Solve an NP-hard problem using approximation algorithms.	**2**	Apply approximation techniques to find near-optimal solutions to NP-hard problems.	5
Practical 15	Implement a randomized algorithm for the Minimum Cut problem.	**2**	Understand and apply randomized algorithms to solve complex problems.	5,6

TEXT BOOKS:

T3: Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.

T4: Fundamentals of Algorithms – E. Horowitz et al.

REFERENCE BOOKS:

R1: Algorithm Design, 1ST Edition, Jon Kleinberg and ÉvaTardos, Pearson.

R2: Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition,

R3: Michael T Goodrich and Roberto Tamassia, Wiley.

R4: Algorithms -- A Creative Approach, 3RD Edition, UdiManber, Addison-Wesley, Reading, MA.

OTHER LEARNING RESOURCES:

1. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/
2. https://www.vssut.ac.in/lecture_notes/lecture1428551222.pdf

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand analysis of algorithms using asymptotic notations for advanced problem-solving.	CO1, CO2, CO3, CO4, CO11, CO12
2	Implement sorting, searching, matrix multiplication algorithms with varied strategies.	CO1, CO2, CO3, CO4, CO11, CO12
3	Apply greedy approaches and dynamic programming to optimize problem-solving, for advanced algorithm design.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12
4	Apply problem-solving strategies to complex scenarios, solving N-Queen, knapsack, and TSP, etc.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12
5	Evaluate decision tree bounds, interpret complexity classes P, NP, NP-Complete proficiently.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BTCS3203R	Design and Analysis of Algorithms	CO1	2	2	2	2							3	2
		CO2	2	2	2	2							2	2
		CO3	2	2	3	3					1	1	2	2
		CO4	2	3	2	2					1	1	2	2
		CO5	1	2	1	2	2				1	1	1	2

SEMESTER – VI									
Course Title	**PE-III: Machine Learning**								
Course code	**24BTCS3203R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T + 30P	**4**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the program**								
Course Objectives (Minimum 3)	1. To provide fundamental theoretical foundations knowledge of various learning algorithms 2. Apply suitable machine learning techniques for data handling and knowledge extraction 3. To gain the fundamental knowledge and understand the context of supervised and unsupervised learning through real-life examples 4. Evaluate the performance of algorithms based on corresponding metrics identified and to provide solutions for various real world applications								
CO1	Recognize the characteristics of machine learning strategies								
CO2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations								
CO3	Analyze and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns								
CO4	Understand the concept of ensemble learning and its applications								
CO5	Evaluate the performance of various algorithms								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Machine Learning** What is Machine Learning?, Types of Machine Learning: Supervised, Unsupervised, and Reinforcement Learning, Applications of Machine Learning in Various Fields	**7**	Gain knowledge of Machine Learning fundamentals, including types (supervised, unsupervised, reinforcement), and applications across diverse fields for practical implementation.	**1, 2**
II	**Data Preprocessing and Exploratory Data Analysis** Data Collection and Cleaning, Handling Missing Values, Feature Scaling and Normalization, Data Visualization Techniques, Exploratory Data Analysis (EDA), statistical measures, Basic tools (plots, graphs and summary statistics) of EDA	**8**	Master data pre-processing techniques including collection, cleaning, handling missing values, feature scaling, normalization, visualization, and exploratory data analysis tools.	**3, 4**
III	**Supervised Learning and Unsupervised Learning** Linear and Non-Linear examples – Multi– Class & Multi-Label classification – Linear Regression,	**12**	Demonstrate proficiency in supervised learning (regression, classification) and unsupervised learning	**3, 4**

	Multiple Linear Regression – Naïve Bayes Classifier – Decision Trees – ID3 – CART, K-NN classifier Logistic regression, Perceptron – Single layer & Multi-layer, Support, Clustering basics (Partitioned, Hierarchical and Density based) - K- Means clustering – K-Mode clustering – Self organizing maps. Expectation maximization – Principal Component Analysis		(clustering, dimensionality reduction) algorithms for data analysis and modeling tasks.	
IV	**Ensemble Learning** Bagging-Committee Machines and Stacking-Boosting-Ranking based aggregation based aggregation	**8**	Apply ensemble learning techniques such as bagging, boosting, stacking, and ranking for improving predictive model accuracy and robustness.	**3, 4, 5**
V	**Model Evaluation and Optimization** Model Validation Techniques: Train/Test Split, Cross-Validation, Performance Metrics, Confusion Matrix, Accuracy, Precision, Recall, F1-Score	**10**	Effectively evaluate machine learning models using validation techniques like train/test split, cross-validation, and performance metrics (accuracy, precision, recall, F1-score).	**3, 5**

Practical Component				
Practical Number	**Topic / Experiment**	**Contact Hours**	**Learning Outcome**	**BL**
Practical 1	**Data Preprocessing and Cleaning:** Collect a real-world dataset, handle missing values, apply feature scaling (standardization or normalization), and clean the data for analysis.	**3**	Understand the importance of data preprocessing and apply techniques to prepare datasets for machine learning models.	1,2,3
Practical 2	**Exploratory Data Analysis (EDA):** Perform EDA using statistical measures, summary statistics, and visualizations (e.g., histograms, box plots, scatter plots, correlation matrices).	**3**	Analyze and visualize data to uncover patterns, trends, and relationships.	3,4
Practical 3	**Linear and Multiple Linear Regression:** Implement linear and multiple linear regression models on a dataset to predict continuous	**3**	Develop regression models for predictive analysis and assess their accuracy.	3,4

	variables and evaluate model performance.			
Practical 4	**Classification Using Naïve Bayes and Logistic Regression:** Build and evaluate classification models using Naïve Bayes and logistic regression techniques for binary classification tasks.	**3**	Apply probabilistic and regression-based methods for classification problems.	4,5
Practical 5	**Decision Trees and K-Nearest Neighbors (K-NN):** Implement Decision Trees (ID3, CART) and K-NN classifiers to classify datasets and compare their performances.	**3**	Understand decision-based and distance-based classification techniques.	3,4
Practical 6	**Clustering Techniques:** Perform clustering using K-Means, K-Mode, and hierarchical clustering methods. Visualize clusters and evaluate using metrics like Silhouette Score.	**3**	Apply unsupervised learning techniques to group and analyze datasets.	4,5
Practical 7	**Principal Component Analysis (PCA):** Apply PCA to reduce the dimensionality of a high-dimensional dataset, visualize the results, and analyze the variance retained.	**3**	Understand dimensionality reduction and apply PCA to simplify data representation.	4,5
Practical 8	**Ensemble Learning Techniques:** Implement ensemble models like Bagging (Random Forest), Boosting (AdaBoost, Gradient Boosting), and Stacking on classification datasets.	**3**	Combine multiple models to improve predictions and analyze ensemble learning advantages.	3,4
Practical 9	**Model Evaluation and Optimization:** Evaluate classification models using metrics like confusion matrix, accuracy, precision, recall, F1-score, and perform hyperparameter tuning (e.g., Grid Search).	**3**	Assess and optimize machine learning models for better performance.	5
Practical 10	**Self-Organizing Maps and Expectation Maximization:** Implement self-organizing maps (SOM) and expectation-maximization clustering for unsupervised learning tasks.	**3**	Apply advanced unsupervised learning algorithms to analyze and cluster data.	5,6

TEXT BOOKS:

T1: S. Shalev-Shwartz, S.Ben-David, “Understanding Machine Learning: From Theory to Algorithms”, Cambridge University Press, 2014.

T2: Ethem Alpaydin,"Introduction to Machine Learning”, MIT Press, Prentice Hall of India,

REFERENCE BOOKS:

R1: Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning", MIT Press, 2nd Edition, 2018.
R2: Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, 1997.

OTHER LEARNING RESOURCES:

O1: Coursera provides course like "Machine Learning" that covers a broad introduction to machine learning, data mining, and statistical pattern recognition.

O2: edX provides course like "Machine Learning" that covers the essentials of machine learning, focusing on supervised and unsupervised learning, mathematical and heuristic aspects, and hands-on implementation of algorithms.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Recognize the characteristics of machine learning strategies	PO1, PO2, PO3, PO4, PO10, PO12
2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations	PO1, PO2, PO3, PO4, PO10, PO12
3	Analyze and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Understand the concept of ensemble learning and its applications	PO1, PO2, PO3, PO4, PO10, PO12
5	Evaluate the performance of various algorithms	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS3203R	Machine Learning	CO1	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VI										
Course Title	**PE-III: Cloud Computing**									
Course code	**24BTCS3204R**	**Total credits: 3** **Total hours: 45T + 30P**	**L** **3**	**T** **0**	**P** **2**	**S** **0**	**R** **0**	**O/F** **0**	**C** **4**
Pre-requisite	Basic Programming Skills	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter / II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. To provide an overview of an exciting field of Cloud Computing 2. To introduce tools requires building, deploying, running and managing applications on a cloud platform. 3. To develop the cloud application development skills, such as Python, REST architecture, JSON, Cloud Foundry and DevOps services 4. To enable students to have skills that will help them to solve complex real-world problems in decision support.									
CO1	Understand fundamental cloud computing concepts and architectures.									
CO2	Analyse and implement RESTful APIs and data services on cloud platforms.									
CO3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.									
CO4	Develop and deploy applications using Python and related frameworks.									
CO5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Cloud Computing and IBM Cloud:** Defining cloud computing, Factors leading to cloud adoption, Types of cloud services (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid), Overview of IBM Cloud Creating an IBM Cloud account, Navigating the IBM Cloud dashboard.	**8**	Understand cloud computing concepts, service models, deployment types, and IBM Cloud basics, including account creation and dashboard navigation.	**1, 2**
II	**IBM Cloud Core Services or data services on IBM Cloud:** Watson AI services on IBM Cloud, Introduction to DevOps, Implementing DevOps practices on IBM Cloud, Using IBM Cloud toolchains for DevOps, REST API, Types and capabilities of databases, Apis interaction with Cloudant database.	**8**	Develop proficiency in IBM Cloud core services, Watson AI, DevOps practices, and interacting with databases and APIs on IBM Cloud.	**1, 2, 3**
III	**Application Development on IBM Cloud:** Understanding business problems and goals, Functional and non-functional requirements, IBM Cloud App ID, Container orchestration (Kubernetes), Pods, Deployment, and Service, Cloud Foundry.	**9**	Develop skills in application development on IBM Cloud, including requirements analysis, App ID, Kubernetes, and Cloud Foundry deployment.	**3, 4**

IV	**Developing Python Applications for the Cloud:** Building Python applications for cloud deployment Integrating popular Python frameworks (Django, Flask) with cloud services Leveraging cloud storage, databases, and AI services with Python Implementing scalability and high availability for Python applications on the cloud	**10**	Develop proficiency in building scalable and high-availability Python applications using frameworks and cloud services.	**2, 3**
V	**Security and Compliance on IBM Cloud:** Overview of IBM Cloud security features, Implementing security best practices on IBM Cloud, Ensuring compliance on IBM Cloud, OAuth protocol.	**10**	Implement IBM Cloud security features, apply best practices, ensure compliance, and use OAuth protocol for secure cloud deployments.	**3, 4, 5**

Practical Component				
Practical Number	**Topic / Experiment**	**Contact Hours**	**Learning Outcome**	**BL**
Practical 1	**Introduction to Cloud and Cloud Deployment Models:** Create an IBM Cloud account, navigate the dashboard, and explore various cloud deployment models (Public, Private, Hybrid)	**2**	Understand cloud computing concepts, deployment models, and the IBM Cloud interface.	1,2,3
Practical 2	**Exploring Cloud Core Services:** Use Cloud services like Watson AI or Cloudant to set up a simple data-driven application using cloud resources.	**2**	Familiarize with IBM Cloud core services and apply them to a basic application.	3,4
Practical 3	**Implementing DevOps Practices on Cloud:** Set up a DevOps pipeline using IBM Cloud toolchains for continuous integration and deployment of an application.	**2**	Understand and implement DevOps practices on the IBM Cloud platform.	3,4
Practical 4	**Using REST APIs with IBM Cloudant Database:** Interact with a Cloudant database using REST APIs to store and retrieve data	**2**	Understand API interactions and database capabilities on IBM Cloud.	4,5
Practical 5	**Kubernetes and Container Orchestration on IBM Cloud:** Deploy a containerized Python application on Kubernetes, manage pods, deployments, and services on IBM Cloud.	**2**	Apply Kubernetes concepts for container orchestration in the cloud environment.	3,4

Practical 6	**Building Python Applications for IBM Cloud:** Develop a Python application using Flask/Django, integrate it with cloud storage and Watson AI services, and deploy it on IBM Cloud.	**2**	Build and deploy scalable Python applications integrated with cloud services.	4,5
Practical 7	**Implementing Scalability and High Availability:** Scale a Python-based cloud application horizontally and vertically, ensuring high availability using IBM Cloud.	**2**	Implement scalability and availability concepts for cloud applications.	4,5
Practical 8	**Security Best Practices on IBM Cloud:** Configure IBM Cloud security features like encryption, IAM roles, and OAuth protocols for a cloud application.	**2**	Implement security best practices and compliance requirements on IBM Cloud.	3,4
Practical 9	**Cloud Foundry Application Deployment:** Deploy a web application using IBM Cloud Foundry and configure functional and non-functional requirements.	**2**	Understand and utilize Cloud Foundry for rapid application deployment.	5
Practical 10	**Case Study on IBM Cloud Security Compliance:** Conduct a case study to ensure compliance with security standards using IBM Cloud tools. Implement OAuth for user authentication in an application.	**2**	Evaluate and implement compliance measures for applications on the IBM Cloud platform.	5,6

TEXT BOOKS:

T1: Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood.

T2: Cloud Computing: A Hands-On Approach by Arshdeep Bahga and Vijay Madisetti.

T3: Cloud Native Architectures: Design High-availability and Cost-effective Applications for the Cloud by Tom Laszewski, Kamal Arora, and Erik Farr.

REFERENCE BOOKS:

R1: Cloud Computing Principles and Paradigms by Rajkumar Buyya, James Broberg, and Andrzej Goscinski.

R2: Cloud Computing: Principles, Systems and Applications by Nick Antonopoulos and Lee Gillam..

R3: Data Mining: Concepts and Techniques", Third Edition, 2 Jiawei Han, Micheline Kamber and Jian Pei, ISBN 0123814790,(2011).

R4: Cloud Computing: From Beginning to End by Ray J. Rafaels.

OTHER LEARNING RESOURCES:

O1: Coursera provides several courses like "Cloud Computing Specialization", "AWS Fundamentals Specialization" and "Google Cloud Platform Fundamentals: Core Infrastructure"

O2: NPTEL provides several courses like "Cloud Computing" and "Cloud Computing and Distributed Systems"

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand fundamental cloud computing concepts and architectures.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Analyze and implement RESTful APIs and data services on cloud platforms.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Develop and deploy applications using Python and related frameworks.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS3204R	**Cloud Computing**	**CO1**	3	3	3	2	3					1		3
		CO2	2	2	2	2	2					1		2
		CO3	3	2	3	2	3					1		3
		CO4	3	3	3	2	3					1		3
		CO5	3	3	3	3	3					1		3

SEMESTER VI									
Course Title	**Statistical Methods and Modelling**								
Course code	**24BTCS3205R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Mathematics	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives	1. To provide a comprehensive understanding of statistical methods and their application in data analysis and decision-making. 2. To equip students with the ability to perform hypothesis testing and statistical inference to draw meaningful conclusions from data. 3. To develop skills in regression, correlation, and advanced statistical modeling for real-world problem-solving. 4. To introduce time series analysis and predictive modeling techniques for forecasting and trend analysis. 5. To enable students to apply statistical tools and software to analyze and visualize data effectively.								
CO1	Understand fundamental statistical concepts and probability distributions to analyze data.								
CO2	Apply statistical inference techniques for hypothesis testing and decision-making.								
CO3	Develop regression and correlation models to establish relationships between variables.								
CO4	Implement advanced statistical modeling methods, including ANOVA and time series analysis, for complex datasets.								
CO5	Utilize statistical tools and programming languages to analyze and visualize data effectively.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Statistics and Probability:** - Descriptive Statistics: Measures of Central Tendency (Mean, Median, Mode), Measures of Dispersion (Variance, Standard Deviation, Range). - Probability Basics: Definitions, Rules of Probability, Conditional Probability, Bayes' Theorem. - Probability Distributions: Binomial, Poisson, and Normal Distributions, Introduction to Sampling Distributions	**8**	Understand basic statistical concepts and probability theory for analyzing and modeling data.	**1,2, 3**
II	**Statistical Inference and Hypothesis Testing :** - Sampling Techniques: Simple Random Sampling, Stratified Sampling, and Systematic Sampling. - Hypothesis Testing: Null and Alternative Hypotheses, Type I and Type II Errors, p-values, and Confidence Intervals.	**12**	Apply statistical inference methods to draw conclusions and test hypotheses based on sample data.	**3,4**

	- Parametric Tests: Z-Test, t-Test (One-sample, Two-sample), F-Test. - Non-Parametric Tests: Chi-Square Test, Mann-Whitney U Test, Kruskal-Wallis Test.			
III	**Regression and Correlation:** - Simple Linear Regression: Fitting a Line, Interpretation of Coefficients, and Assumptions. - Multiple Linear Regression: Interpretation of Results, Multicollinearity, and Model Evaluation Metrics. - Correlation: Pearson, Spearman, and Kendall Tau Correlation Coefficients. - Logistic Regression: Binary Classification Problems, Odds Ratio, and Model Evaluation Metrics (ROC, AUC).	**9**	Model relationships between variables using regression and correlation techniques.	**3,4**
IV	**Advanced Statistical Modelling** : - Analysis of Variance (ANOVA): One-Way and Two-Way ANOVA, Interpretation of Results. - Time Series Analysis: Components (Trend, Seasonality, Randomness), Moving Averages, Exponential Smoothing. - Introduction to Generalized Linear Models (GLM): Poisson Regression, Multinomial Logistic Regression. - Model Diagnostics and Residual Analysis.	**9**	Develop advanced models for handling complex datasets and patterns.	**4**
V	**Applications of Statistical Methods in Real-World Problems :** - Case Study 1: Statistical Analysis for Market Research and Consumer Behavior. - Case Study 2: Predictive Modelling in Healthcare (e.g., Survival Analysis). - Case Study 3: Time Series Analysis for Stock Market Predictions. - Data Visualization Techniques: Static and Interactive Plots using tools like Python (Matplotlib, Seaborn) or R (ggplot2).	**9**	Apply statistical methods to solve real-world problems across domains like business, healthcare, and social sciences.	**4**
Practical Component				
Practical 1	Calculate measures of central tendency (mean, median, mode) and dispersion (variance, standard deviation). Create visualizations such as histograms, box plots, and scatter plots.	**2**	Summarize and visualize data to understand its structure and distribution.	1,2

Practical 2	Generate and analyze binomial, Poisson, and normal distributions. Fit data to these distributions and interpret the results.	**2**	Understand and apply probability distributions to model real-world scenarios.	1,3
Practical 3	Perform a one-sample t-test, two-sample t-test, and paired t-test on given datasets. Interpret the p-values and draw conclusions.	**2**	Conduct hypothesis tests and interpret results for decision-making.	2,3
Practical 4	Perform Chi-Square tests for independence and goodness-of-fit, and use Mann-Whitney U test for comparing two independent groups.	**2**	Apply non-parametric tests to datasets that do not meet parametric assumptions.	3
Practical 5	Fit a simple linear regression model to a dataset, interpret coefficients, and evaluate model performance using R^2 and residual analysis.	**2**	Model and interpret relationships between two variables using linear regression.	3,4
Practical 6	Build a multiple linear regression model to analyze relationships between a dependent variable and multiple independent variables. Check for multicollinearity and assess model performance.	**2**	Develop and evaluate regression models with multiple predictors.	1,2
Practical 7	Compute Pearson, Spearman, and Kendall Tau correlation coefficients for a dataset. Visualize correlations using a heatmap.	**2**	Analyze and interpret the strength and direction of relationships between variables.	1,3
Practical 8	Decompose a time series into trend, seasonality, and residual components. Perform moving average smoothing and exponential smoothing.	**2**	Analyze time series data and apply smoothing techniques for trend detection.	2,3
Practical 9	Perform one-way and two-way ANOVA to test differences between group means. Analyze the results and visualize group comparisons.	**2**	Compare means across multiple groups and interpret statistical significance.	3
Practical 10	Apply PCA on a high-dimensional dataset, reduce dimensionality, and visualize the results in a lower-dimensional space. Analyze the variance explained by each principal component.	**2**	Reduce dimensionality and extract meaningful patterns from high-dimensional datasets	3,4

TEXT BOOKS:

T1: Gupta, S. C., & Kapoor, V. K. - *"Fundamentals of Mathematical Statistics"*, Sultan Chand & Sons, 11th Edition, 2014.

T2: Montgomery, D. C., & Runger, G. C. - *"Applied Statistics and Probability for Engineers"*, Wiley, 7th Edition, 2020.

REFERENCE BOOKS:

R1: Freund, J. E., & Perles, B. M. - "Statistics: A First Course", Pearson, 9th Edition, 2017.

R2: Casella, G., & Berger, R. L. - "Statistical Inference", Cengage Learning, 2nd Edition, 2001.

OTHER LEARNING RESOURCES:

O1: Online Courses:

"Statistics for Data Science and Business Analysis" by Udemy.

"Statistical Thinking for Data Science and Analytics" by EdX (Columbia University).

O2: Web Resources:

Khan Academy: Modules on probability, statistics, and regression analysis.

DataCamp: Interactive tutorials on statistical modeling and data visualization using Python.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enabling solving skills of ordinary differential equations of various order.	PO1, PO2, PO5, PO12
2	Enabling solving skills of higher order ODE with constants coefficients.	PO1, PO2, PO5, PO12
3	Enable to formulate and solve problems involving random variables.	PO1, PO2, PO4, PO6, PO12
4	Enable to solve problems of various probability distribution.	PO1, PO4, PO5, PO10, PO12
5	Analysing experimental data by applying statistical methods.	PO2, PO4, PO5, PO11, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS3205R	Statistical Methods and Modelling	CO1	3	3			2							1
		CO2	3	3			2							2
		CO3	2	2		2		1						1
		CO4	2			2	2					1		1
		CO5		3		3	2						1	2

SEMESTER – VI									
Course Title	**Mini Project IV**								
Course code	**24BTCS3206R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 48	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the third year of the program**								
Course Objectives (Minimum 3)	1. Develop advanced technical skills through individual or group project work in engineering domains. 2. Enhance project management abilities, including workload management and meeting project deadlines effectively. 3. Improve presentation skills through seminar presentations and public demonstrations of project outcomes.								
CO1	Apply IT principles to solve real-world problems.								
CO2	Contribute to complex projects as a team member.								
CO3	Apply programming languages and platforms to develop software and hardware solutions.								
CO4	Demonstrate personal and team management skills in software development.								
CO5	Cultivate employability skills and professionalism commitment.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a. theoretical studies b. computer simulations c. hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;	**48**	Demonstrate proficiency in conducting literature reviews, performing technical work, maintaining progress reports, delivering seminars, preparing formal reports, and presenting project outcomes effectively.	**2,3,4,5,6**

	5. Prepare a formal report describing the work undertaken and results obtained so far; and 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

REFERENCE BOOKS:

R1. "The Craft of Research" by Wayne C. Booth, Gregory G. Colomb, and Joseph M. Williams

R2. "Doing Your Research Project: A Guide for First-Time Researchers" by Judith Bell and Stephen Waters

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles to solve real-world problems.	CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9, CO10, CO11, CO12.
2	Contribute to complex projects as a team member.	CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9, CO10,CO11, CO12.
3	Apply programming languages and platforms to develop software and hardware solutions.	CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9, CO10, CO11, CO12.
4	Demonstrate personal and team management skills in software development.	CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9, CO10, CO11, CO12.
5	Cultivate employability skills and professionalism commitment.	CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9, CO10,CO11, CO12.

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BTCS3206R	Mini Project IV	CO 1	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

Semester VII										
Course Title	**PE-IV: Fundamentals of Cyber Security**									
Course code	**24BTCS4101R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**	
			3	**0**	**2**	**0**	**0**	**0**	**4**	
Pre-requisite	NIL	**Co-requisite**	NIL							
Programme	**Bachelor of Technology in Computer Science & Engineering**									
Semester	**Summer / I semester of the fourth year of the Programme**									
Course Objectives (Minimum 3)	1.To provide students with a thorough understanding of the key principles and practices of cyber security. 2. To equip students with the technical skills necessary to identify, analyze, and mitigate cyber threats. 3. To foster an understanding of ethical considerations and legal implications in the field of cyber security.									
CO1	Demonstrate an understanding of various types of cyber threats and vulnerabilities.									
CO2	Apply risk management strategies to protect information systems.									
CO3	Develop technical skills in using tools and techniques for cyber defence.									
CO4	Formulate and implement effective incident response and recovery strategies.									
CO5	Recognize and apply ethical principles and legal regulations relevant to cyber security.									

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Cyber Security** Definition and importance of cyber security, Overview of cyber threats and attacks, Cyber security principles and practices, The role of cyber security professionals	**8**	Understand cybersecurity fundamentals, threats, vulnerabilities, defences, and apply security best practices to protect digital assets effectively.	**1,2,3**
II	**Network Security** Fundamentals of network security Firewalls, VPNs, and intrusion detection/prevention systems Secure network design and architecture Network monitoring and analysis	**8**	Comprehend network security principles, protocols, encryption techniques, intrusion detection/prevention, and implement measures to safeguard network infrastructure effectively.	**2,4**
III	**Cryptography** Introduction to cryptography and its importance Symmetric and asymmetric encryption Digital signatures and certificates Cryptographic protocols and applications	**10**	Understand cryptographic algorithms, encryption/decryption methods, digital signatures, cryptographic protocols, and apply them to secure communication and data.	**2,4**
IV	**Risk Management and Incident Response** Risk assessment and management strategies ,Identifying and mitigating vulnerabilities, Incident response planning and execution Business continuity and disaster recovery	**10**	Analyze cybersecurity risks, develop risk management strategies, implement incident response plans, and mitigate security incidents effectively.	**2,3,4**

V	**Ethical and Legal Aspects of Cyber Security** Ethical issues in cyber security Legal and regulatory frameworks Compliance requirements (e.g., GDPR, HIPAA) Case studies of cyber security breaches and legal implications Objectives:.	**9**	Comprehend ethical dilemmas, legal frameworks, regulations, and apply ethical principles to cybersecurity practices and decision-making processes.	**2,4**

Practical Component				
Practical 1	Match key cybersecurity terms with their definitions.	2	Identify and understand key cyber security terms and definitions.	1,2,3
Practical 2	Identify different types of cyber threats.	2	Recognize and categorize different types of cyber threats.	3,4
Practical 3	Explain basic concepts of network security.	2	understand and explain the components and principles of network security.	3,4
Practical 4	Explain various encryption techniques.	2	Understand and explain different encryption techniques and their applications.	4,5
Practical 5	Configure a basic firewall.	2	Apply firewall configuration techniques to control network traffic.	3,4
Practical 6	Implement basic encryption for data protection.	2	Apply encryption techniques to protect data.	4,5
Practical 7	Conduct a vulnerability assessment on a system.	2	Analyze and identify vulnerabilities in systems using appropriate tools.	4,5
Practical 8	Analyze network traffic for suspicious activity.	2	Analyze network traffic to detect suspicious activities.	3,4
Practical 9	Evaluate the effectiveness of security policies.	2	Evaluate and suggest improvements for security policies.	5
Practical 10	Assess the robustness of an incident response plan.	2	Assess and critique the robustness of incident response plans. runtime environment.	5,6
Practical 11	Develop a security awareness program for an organization.	2	Create an effective security awareness program for	4,5

			organizational training.	
Practical 12	Design a secure network architecture.	2	Design a secure network architecture to protect against cyber threats.	4,5
Practical 13	Implement an intrusion detection system (IDS).	2	Implement and configure an IDS to monitor and detect intrusions.	3,4
Practical 14	Develop a penetration testing plan.	2	Develop and document a comprehensive penetration testing plan.	5
Practical 15	Create a detailed cyber security incident report.	2	Create a comprehensive incident report detailing their response and findings from a simulated cybersecurity incident.	5,6

Textbooks

1. "Cybersecurity Essentials" by Charles J. Brooks, Christopher Grow, Philip Craig, and Donald Short

Reference Books

1. "Network Security Essentials: Applications and Standards" by William Stallings.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate an understanding of various types of cyber threats and vulnerabilities.	CO1, CO2, CO3, CO4, CO5, CO12.
2	Apply risk management strategies to protect information systems.	CO1, CO2, CO3, CO4, CO5, CO12.
3	Develop technical skills in using tools and techniques for cyber defence.	CO1, CO2, CO3, CO4, CO5, CO12.
4	Formulate and implement effective incident response and recovery strategies.	CO1, CO2, CO3, CO4, CO5, CO12.
5	Recognize and apply ethical principles and legal regulations relevant to cyber security.	CO1, CO2, CO3, CO4, CO5, CO10, CO12.

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS4101R	Fundamentals of Cyber Security	CO1	3	1	2	1	1							3
		CO2	3	2	3	1	1							3
		CO3	3	2	3	1	1							3
		CO4	3	2	3	1	2							3
		CO5	3	3	3	3	2					1		3

SEMESTER – VII									
Course Title	**PE-IV: Foundations of Data Science**								
Course code	**24BTCS4101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+ 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Mathematics and Statistics, Programming	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science & Engineering**								
Semester	**Summer / I semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. Develop Proficiency in Key Data Science Tools and Programming Languages. 2. Understand and Apply Fundamental Statistical and Machine Learning Methods. 3. Cultivate Critical Thinking and Problem-Solving Skills in a Data-Driven Context.								
CO1	Acquire foundational knowledge in data science.								
CO2	Understand and apply various data analytics techniques.								
CO3	Apply advanced tools to work on dimensionality reduction and mathematical operations.								
CO4	Manage diverse data types and employ programming for knowledge representation through visualization.								
CO5	Demonstrate numerous open source data science tools to solve real-world problems through industrial case studies.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Foundations of Data Science:** Overview of data science, its significance, and application across various sectors. Introduction to the data science workflow: data collection, cleaning, analysis, and visualization.	**8**	Grasp foundational concepts in data science, acquire programming skills, analyze data, and communicate findings effectively using statistical methods.	**1, 2**
II	**Statistical Methods and Data Manipulation:** Essential statistics for data science, probability theory, data summarization, and visualization techniques. Introduction to programming with Python and R for data manipulation using Pandas and NumPy.	**8**	Understand statistical techniques, manipulate data using programming languages, analyze datasets, interpret results, and communicate findings accurately.	**2,3**
III	**Machine Learning Techniques:** Supervised learning models (regression, decision trees, SVMs), unsupervised learning models (clustering, PCA), and model evaluation metrics. Introduction to machine learning algorithms implementation using scikit-learn.	**10**	Understand machine learning algorithms, apply techniques to analyze data, develop models, evaluate performance, and make predictions effectively.	**2,4**
IV	**Advanced Data Science Concepts:** Deep learning with TensorFlow, natural language	**10**	Explore advanced topics in data science such as deep learning, natural	**2,3**

	processing basics, and introduction to big data technologies like Hadoop and Spark. Real-world applications and case studies.		language processing, and big data analytics, applying them effectively.	
V	**Capstone Project in Data Science:** Application of data science processes to a real-world problem. Project phases include problem identification, data acquisition, preprocessing, modelling, analysis, and presentation of results.	**9**	Demonstrate proficiency in applying data science skills to solve real-world problems, presenting comprehensive solutions and insights effectively.	**2,5**
Practical Component				
Practical 1	Match basic data science terminology with their definitions.	2	Identify and understand key data science terms and definitions.	1,2,3
Practical 2	Identify different types of data (categorical, numerical, etc.).	2	Recognize and categorize different types of data in various data sets.	3,4
Practical 3	Interpret basic data visualizations (bar charts, histograms, scatter plots).	2	Interpret and explain information presented in basic data visualizations.	3,4,5
Practical 4	Explain the concept and importance of descriptive statistics.	2	Understand and explain the importance and application of descriptive statistics.	3,4,5
Practical 5	Perform basic data cleaning tasks.	2	Apply data cleaning techniques to prepare data sets for analysis.	3,4
Practical 6	Create data visualizations using software (e.g., Excel, Tableau, Python libraries).	2	Create accurate and effective data visualizations to represent data insights.	3,4
Practical 7	Analyze the correlation between variables.	2	Analyze and interpret the correlation between variables in data sets..	3,4
Practical 8	Perform and interpret hypothesis tests.	2	Perform hypothesis tests and interpret their significance in data analysis..	3,4
Practical 9	Evaluate the performance of machine learning models..	2	evaluate and interpret the performance of machine learning models using appropriate metrics.	3,4,5
Practical 10	Critically analyze the quality and reliability of data sources.	2	critically assess the quality and reliability of various data sources.	3,4,5
Practical 11	Build and test predictive models.	2	Build and test predictive models, demonstrating their understanding of model development.	3,4,5

Practical 12	Design a survey for data collection.	2	design effective surveys to collect relevant and unbiased data.	2,3,5
Practical 13	Implement a data processing pipeline.	2	Implement an ETL pipeline to process data efficiently.	2,3,5
Practical 14	Create a data-driven story.	2	Create and present a data-driven story, effectively communicating insights.	2,3,5
Practical 15	Develop an interactive data dashboard.	2	develop interactive dashboards to visualize and explore data insights dynamically..	2,3,5

Text Books:

1. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking, Foster Provost and Tom Fawcett, 1st Edition, 2013.
2. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython", Wes McKinney, 2nd Edition, 2017.
3. Pattern Recognition and Machine Learning", Christopher M. Bishop, 1st Edition, 2006.
4. Applied Predictive Modeling", Max Kuhn and Kjell Johnson, 1st Edition, 2013.

Reference Books:

1. An Introduction to Statistical Learning: with Applications in R, Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, 2nd Edition, 2021.
2. Machine Learning: A Probabilistic Perspective, Kevin P. Murphy, 1st Edition, 2012.
3. Data Mining: Practical Machine Learning Tools and Techniques, Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal, 4th Edition, 2016.
4. Big Data: Principles and Best Practices of Scalable Realtime Data Systems, Nathan Marz and James Warren, 1st Edition, 2015.

Additional Resources:

- SKaggle - https://www.kaggle.com/
- DataCamp - https://www.datacamp.com/
- Google Colab - https://colab.research.google.com/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Acquire foundational knowledge in data science.	CO1, CO2, CO3, CO4, CO10, CO12.
2	Understand and apply various data analytics techniques.	CO1, CO2, CO3, CO4, CO10, CO12.

3	Apply advanced tools to work on dimensionality reduction and mathematical operations.	CO1, CO2, CO3, CO4, CO10, CO12.
4	Manage diverse data types and employ programming for knowledge representation through visualization.	CO1, CO2, CO3, CO4, CO10, CO12.
5	Demonstrate numerous open source data science tools to solve real-world problems through industrial case studies.	CO1, CO2, CO3, CO4, CO10, CO12.

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS4101R	**Foundations of Data Science**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

<table>
<tr><th colspan="10">SEMESTER – VII</th></tr>
<tr><td>Course Title</td><td colspan="9">Project-I</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BTCS4102R</td><td rowspan="2">Total credits: 6
Total hours: 144</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>16</td><td>12</td><td>0</td><td>6</td></tr>
<tr><td>Pre-requisite</td><td>NIL</td><td>Co-requisite</td><td colspan="7">NIL</td></tr>
<tr><td>Programme</td><td colspan="9">Bachelor of Technology in Computer Science & Engineering</td></tr>
<tr><td>Semester</td><td colspan="9">Summer / I semester of the fourth year of the Programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="9">1. To enable students to perform comprehensive literature searches, critically review current knowledge, and synthesize information on advanced technical topics.
2. To provide students with hands-on experience in conducting detailed technical work, including theoretical studies, computer simulations, or hardware construction.
3. To improve students' ability to communicate their research findings and technical work effectively through progress reports, seminars, formal reports, and presentations.</td></tr>
<tr><td>CO1</td><td colspan="9">Apply IT principles for real-world problem-solving using programming languages and network solutions.</td></tr>
<tr><td>CO2</td><td colspan="9">Contribute to complex projects as a team member, demonstrating effective personal and team management.</td></tr>
<tr><td>CO3</td><td colspan="9">Develop programming skills to create and assess software, hardware, and network solutions.</td></tr>
<tr><td>CO4</td><td colspan="9">Apply personal and team management skills as a professional software developer.</td></tr>
<tr><td>CO5</td><td colspan="9">Cultivate employability skills and uphold professionalism in software development.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="3">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
a) theoretical studies
b) computer simulations
c) hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;</td><td>144</td><td colspan="3">Develop and Demonstrate comprehensive research, technical, and communication skills through literature review, technical implementation, progress documentation, and presentation.</td><td>3,4,5,6</td></tr>
</table>

	4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Distinguish between different sampling methods and identify sampling and non-sampling errors in data collection.	CO1,CO2,CO3, CO4,CO5,CO6,CO7,CO8,CO9,CO10,CO11, CO12
2	Apply probability theorems to solve problems involving binomial, Poisson, exponential, beta, and normal distributions.	CO1,CO2,CO3, CO4,CO5,CO6,CO7,CO8,CO9,CO10,CO11, CO12
3	Conduct hypothesis tests and determine the significance for attributes and variables in large and small samples.	CO1,CO2,CO3, CO4,CO5,CO6,CO7,CO8,CO9,CO10,CO11, CO12
4	Analyse relationships using multiple regression and correlation techniques; apply control charts in quality control.	CO1,CO2,CO3, CO4,CO5,CO6,CO7,CO8,CO9,CO10,CO11, CO12
5	Perform non-parametric tests and analysis of variance (ANOVA) to evaluate data distributions.	CO1,CO2,CO3, CO4,CO5,CO6,CO7,CO8,CO9,CO10,CO11, CO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS4102R	**PROJECT-I**	**CO1**	3	3	2	3	2	1	1	2	3	2	2	3
		CO2	3	3	3	3	2	1	1	2	3	2	2	3
		CO3	3	2	3	2	3	1	1	2	3	3	2	3
		CO4	3	2	2	2	2	1	1	2	3	3	3	3

		CO5	3	3	3	3	3	1	3	2	3	3	2	3

<table>
<tr><th colspan="10">SEMESTER – VII</th></tr>
<tr><td>Course Title</td><td colspan="9">Summer Internship</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BTCS4104R</td><td>Total credits: 4</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>Total hours: 120</td><td>0</td><td>0</td><td>0</td><td>16</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="7">Nil</td></tr>
<tr><td>Programme</td><td colspan="9">Bachelor of Technology in Computer Science & Engineering</td></tr>
<tr><td>Semester</td><td colspan="9">Summer / I semester of the fourth year of the Programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="9">1. Equip students with practical experience in their field of study, ensuring they are prepared to transition smoothly into industry roles by mastering modern tools, technologies, and industry-standard practices.
2. Enhance students' soft skills, including presentation, interpersonal communication, and documentation, as well as office etiquette, to ensure they can effectively navigate and contribute to professional environments.
3. Foster a deep understanding of professional and ethical responsibilities in engineering, encouraging students to adhere to industry best practices, processes, and regulations.
4. Promote the ability to work effectively in teams, especially within multidisciplinary contexts, emphasizing collaboration, communication, and the integration of diverse perspectives and expertise.
5. Cultivate a professional work ethic characterized by productivity, consistency, and punctuality, preparing students to meet the demands and expectations of the industry.</td></tr>
<tr><td>CO1</td><td colspan="9">Apply modern tools and technologies in real-world scenarios to solve complex engineering problems.</td></tr>
<tr><td>CO2</td><td colspan="9">Analyse industry-standard reporting methods to produce technical documents adhering to professional guidelines.</td></tr>
<tr><td>CO3</td><td colspan="9">Develop solutions using design, development, and testing practices aligned with current industry standards.</td></tr>
<tr><td>CO4</td><td colspan="9">Demonstrate effective communication, teamwork, and interpersonal skills in a multi-disciplinary team environment.</td></tr>
<tr><td>CO5</td><td colspan="9">Evaluate professional and ethical responsibilities to make informed decisions in engineering practices.</td></tr>
<tr><td>Unit-No.</td><td colspan="5">Content</td><td>Contact Hour</td><td colspan="2">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="5">Method and Pedagogy:
● Summer internship shall be at least 90 hours during the summer vacation only.
● Department/Institute will help students to find an appropriate company/industry/organization for the summer internship.
● The student must fill up and get approved a Summer Internship Acceptance form by the company and provide it to the Coordinator of the department within the specified deadline.
● Students shall commence the internship after the approval of the department Coordinator. Summer internships in research centers is also allowed.
● During the entire period of internship, the student shall obey the rules and regulations of the company/industry/organization and also those of the University.
● Due to inevitable reasons, if the student will not able to attend the internship for few days with</td><td>120</td><td colspan="2">Complete a 90-hour summer internship, submit an internship report, and demonstrate practical industry experience through documentation and evaluation.</td><td>3, 4, 5</td></tr>
</table>

	the permission of the supervisor, the department Coordinator should be informed via e-mail and these days should be compensated later. ● The student shall submit two documents to the Coordinator for the evaluation of the summer internship: a) Summer Internship Report b) Summer Internship Assessment Form ● Upon the completion of summer internship, a hard copy of "Summer Internship Report" must be submitted to the Coordinator by the first day of the new term. ● The report must outline the experience and observations gained through practical internship, in accordance with the required content and the format described in this guideline. Each report will be evaluated by a faculty member of the department on a satisfactory/unsatisfactory basis at the beginning of the semester. ● If the evaluation of the report is unsatisfactory, it shall be returned to the student for revision and/or rewriting. If the revised report is still unsatisfactory the student shall be requested to repeat the summer internship.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply modern tools and technologies in real-world scenarios to solve complex engineering problems.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
2	Analyse industry-standard reporting methods to produce technical documents adhering to professional guidelines.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
3	Develop solutions using design, development, and testing practices aligned with current industry standards.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
4	Demonstrate effective communication, teamwork, and interpersonal skills in a multi-disciplinary team environment.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
5	Evaluate professional and ethical responsibilities to make informed decisions in engineering practices.	PO1, PO2, PO3, PO4, PO5, PO6, PO8, PO9, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS4104R	Summer Internship	CO1	3	3	3	3	3	3			2	1		2
		CO2	2	2	3	3	2	2			2	1		2
		CO3	2	2	3	2	2	2			2	1		2
		CO4	3	3	3	3	3	3			2	3		2
		CO5	2	2	2	2	2	2		3	2	1		2

SEMESTER – VIII									
Course Title	**Cryptography and Network Security**								
Course code	**24BTCS4201R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. To provide deeper understanding into cryptography, its application to network security, threats/vulnerabilities to networks and countermeasures. 2. To explain various approaches to Encryption techniques, strengths of Traffic Confidentiality, Message Authentication Codes. 3. To familiarize Digital Signature Standard and provide solutions for their issues.								
CO1	Identify basic security attacks and services.								
CO2	Use symmetric and asymmetric key algorithms for cryptography.								
CO3	Design a security solution for a given application.								
CO4	Analyze Key Management techniques and the importance of number Theory.								
CO5	Understanding of Authentication functions how Message Authentication Codes and Hash Functions work.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**INTRODUCTION**: Security trends, The OSI Security Architecture, Security Attacks, Security Services, and Security Mechanisms, A model for Network security. CLASSICAL ENCRYPTION TECHNIQUES: Symmetric Cipher Modes, Substitute Techniques, Transposition Techniques, Rotor Machines, Stenography.	7	To understand different security services and security mechanisms and different classical encryption techniques	1,2,3
II	**BLOCK CIPHER AND DATA ENCRYPTION STANDARDS:** Block Cipher Principles, Data Encryption Standards, the Strength of DES, Differential and Linear Crypt Analysis, Block Cipher Design Principles. ADVANCED ENCRYPTION STANDARDS: Evaluation Criteria for AES, the AES Cipher. MORE ON SYMMETRIC CIPHERS: Multiple Encryption, Triple DES, Block Cipher Modes of Operation, Stream Cipher and RC4.	7	Understanding block cipher and data encryption standards	3,4,5
III	**PUBLIC KEY CRYPTOGRAPHY AND RSA:** Principles Public key crypto Systems, Diffie Hellman Key Exchange, the RSA	7	Analyzing the concepts of Public key	3,4,5

	algorithm, Key Management, Elliptic Curve Arithmetic, Elliptic Curve Cryptography. MESSAGE AUTHENTICATION AND HASH FUNCTIONS: Authentication Requirement, Authentication Function, Message Authentication Code, Hash Function, Security of Hash Function and MACs. HASH AND MAC ALGORITHM: Secure Hash Algorithm, Whirlpool, HMAC, CMAC. DIGITAL SIGNATURE: Digital Signature, Authentication Protocol, Digital Signature Standard.		cryptography and RSA.	
IV	**AUTHENTICATION APPLICATION:** Kerberos, X.509Authentication Service, Public Key Infrastructure. EMAILSECURITY: Pretty Good Privacy (PGP) and S/MIME. IPSECURITY: Overview, IP Security Architecture, Authentication Header, Encapsulating Security Pay load, Combining Security Associations and Key Management.	7	Analyzation of the concepts of Authentication Application, Email Security and IP Security.	3,5
V	**WEB SECURITY:** Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET), Intruders, Viruses and related threats. FIREWALL: Firewall Design principles, Trusted Systems	7	Analyzation of Web Security and firewall.	3,4,5

TEXT BOOKS:

T1: William Stallings (2006), Cryptography and Network Security: Principles and Practice,4th edition, Pearson Education, India.

T2: William Stallings (2000), Network Security Essentials (Applications and Standards), Pearson Education, India.

REFERENCEBOOKS:

R1: Charlie Kaufman (2002), Network Security: Private Communication in a Public World, 2nd edition, Prentice Hall of India, New Delhi.

R2: Atul Kahate (2008), Cryptography andNetworkSecurity,2nd edition, Tata McGraw hill, India.

R3: Robert Bragg, Mark Rhodes (2004), Network Security: The complete reference, Tata McGraw-Hill, India.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Identify basic security attacks and services.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
2	Use symmetric and asymmetric key algorithms for cryptography.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
3	Design a security solution for a given application.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
4	Analyze Key Management techniques and the importance of number Theory.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
5	Understanding of Authentication functions the manner in which Message Authentication Codes and Hash Functions work.	CO1,CO2,CO3,CO4,CO5,CO10,CO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BTCS4201R	**Cryptography & Network Security**	**CO1**	2	2	1	1	1					1		2
		CO2	3	3	3	3	2					2		3
		CO3	3	2	3	3	3					2		3
		CO4	2	2	2	2	2					2		2
		CO5	2	2	3	2	2					2		3

SEMESTER – VIII									
Course Title	**Machine Learning**								
Course code	**24BTCS4105R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**

Pre-requisite	Nil	**Co-requisite**	Nil	
Programme	**Bachelor of Technology in Computer Science and Engineering**			
Semester	**Winter/ II semester of the fourth year of the Programme**			
Course Objectives (Minimum 3)	1. To provide fundamental theoretical foundations knowledge of various learning algorithms 2. Apply suitable machine learning techniques for data handling and knowledge extraction 3. To gain the fundamental knowledge and understand the context of supervised and unsupervised learning through real-life examples 4. Evaluate the performance of algorithms based on corresponding metrics identified and to provide solutions for various real world applications			
CO1	Recognize the characteristics of machine learning strategies			
CO2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations			
CO3	Analyse and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns			
CO4	Understand the concept of ensemble learning and its applications			
CO5	Evaluate the performance of various algorithms			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Machine Learning:** Machine Learning introduction, Types of Machine Learning: Supervised, Unsupervised, and Reinforcement Learning, Applications of Machine Learning in Various Fields	7	Gain knowledge of Machine Learning fundamentals, including types (supervised, unsupervised, reinforcement), and applications across diverse fields for practical implementation.	**1, 2**
II	**Data Preprocessing and Exploratory Data Analysis** Data Collection and Cleaning, Handling Missing Values, Feature Scaling and Normalization, Data Visualization Techniques, Exploratory Data Analysis (EDA), statistical measures, Basic tools (plots, graphs and summary statistics) of EDA	8	Master data pre-processing techniques including collection, cleaning, handling missing values, feature scaling, normalization, visualization, and exploratory data analysis tools.	**3, 4**
III	**Supervised Learning and Unsupervised Learning** Linear and Non-Linear examples – Multi– Class & Multi-Label classification, Linear Regression, Multiple Linear Regression – Naïve Bayes Classifier – Decision Trees – ID3 – CART, K-NN classifier – Logistic regression – Perceptron –	12	Demonstrate proficiency in supervised learning (regression, classification) and unsupervised learning (clustering, dimensionality reduction) algorithms for data analysis and modelling tasks.	**3, 4**

	Single layer & Multi-layer – Support, Clustering basics (Partitioned, Hierarchical and Density based) - K- Means clustering – K-Mode clustering – Self organizing maps, Expectation maximization – Principal Component Analysis			
IV	**Ensemble Learning** Bagging-Committee Machines and Stacking-Boosting-Ranking based aggregation-based aggregation	8	Apply ensemble learning techniques such as bagging, boosting, stacking, and ranking for improving predictive model accuracy and robustness.	**3, 4, 5**
V	**Model Evaluation and Optimization** Model Validation Techniques: Train/Test Split, Cross-Validation, Performance Metrics, Confusion Matrix, Accuracy, Precision, Recall, F1-Score	10	Effectively evaluate machine learning models using validation techniques like train/test split, cross-validation, and performance metrics (accuracy, precision, recall, F1-score).	**3, 5**
Practical Component				
Practical 1	Collect and clean a real-world dataset, handle missing values, and apply feature scaling and normalization techniques.	3	Understand the importance of data preprocessing and apply cleaning techniques to prepare data for machine learning models.	1,2,3
Practical 2	Perform EDA on a dataset using statistical measures and visualization techniques such as histograms, scatter plots, box plots, and correlation matrices.	3	Analyze and visualize data to uncover patterns, trends, and relationships.	3,4
Practical 3	Implement linear regression for a continuous target variable and logistic regression for binary classification tasks.	3	Understand and implement regression models for predictive analysis.	3,4
Practical 4	Build and evaluate classification models using Decision Trees (e.g., ID3, CART) and K-Nearest Neighbors (K-NN).	3	Develop classification models and compare their performance on given datasets.	4,5
Practical 5	Perform clustering using K-Means, K-Mode, and hierarchical clustering methods. Evaluate the clusters using metrics such as Silhouette Score.	3	Group data points into clusters and analyze their structure and patterns.	3,4
Practical 6	Apply PCA to reduce the dimensionality of a high-dimensional dataset and visualize the transformed data.	3	Understand and implement dimensionality reduction techniques to simplify data while retaining essential features.	4,5
Practical 7	Implement ensemble models such as Bagging, Boosting (e.g., AdaBoost,	3	Combine multiple models to improve performance	4,5

	Gradient Boosting), and Stacking on a classification dataset.		and analyze the advantages of ensemble learning.	
Practical 8	Implement a single-layer perceptron and extend it to a multi-layer perceptron (MLP) using a library like TensorFlow or PyTorch.	3	Understand the structure and working of perceptrons and neural networks for classification tasks.	3,4
Practical 9	Evaluate classification models using metrics such as confusion matrix, accuracy, precision, recall, and F1-score. Perform cross-validation.	3	Assess the performance of machine learning models using evaluation metrics.	5
Practical 10	Perform hyperparameter tuning using techniques such as Grid Search and Random Search to optimize machine learning models.	3	Optimize machine learning models to achieve better performance.	5,6

TEXT BOOKS:

T1: S. Shalev-Shwartz, S.Ben-David, "Understanding Machine Learning: From Theory to Algorithms", Cambridge University Press, 2014.

T2: Ethem Alpaydin,"Introduction to Machine Learning", MIT Press, Prentice Hall of India,

REFERENCE BOOKS:

R1: Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning", MIT Press, 2nd Edition, 2018.

R2: Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, 1997.

R3: Data Mining: Concepts and Techniques", Third Edition, 2 Jiawei Han, Micheline Kamber and Jian Pei, ISBN 0123814790,(2011).

R4: Duda, Richard, Peter Hart, and David Stork, "Pattern Classification," 2nd Edition, John Wiley & Sons, Hoboken, 2000.

OTHER LEARNING RESOURCES:

O1: Coursera provides course like "Machine Learning" that covers a broad introduction to machine learning, data mining, and statistical pattern recognition.

O2: edX provides course like "Machine Learning" that covers the essentials of machine learning, focusing on supervised and unsupervised learning, mathematical and heuristic aspects, and hands-on implementation of algorithms.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**

1	Recognize the characteristics of machine learning strategies	PO1, PO2, PO3, PO4, PO10, PO12
2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations	PO1, PO2, PO3, PO4, PO10, PO12
3	Analyze and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Understand the concept of ensemble learning and its applications	PO1, PO2, PO3, PO4, PO10, PO12
5	Evaluate the performance of various algorithms	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BTCS4105R	**Machine Learning**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VIII									
Course Title	**Image Processing and Pattern Recognition**								
Course code	**24BTCS4202R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives	1. To provide an in-depth understanding of the fundamentals of digital image processing, including image representation, enhancement, and transformation								

(Minimum 3)	techniques. 2. To equip students with the knowledge of image segmentation and compression methods for efficient image analysis and storage. 3. To introduce the principles of feature extraction and pattern recognition, focusing on classifiers and dimensionality reduction techniques. 4. To explore the applications of machine learning and deep learning models in image classification and object detection tasks. 5. To enable students to design and implement real-world projects involving image processing and pattern recognition, such as face recognition or medical image analysis.			
CO1	Understand spatial filtering techniques to enhance digital images in the spatial domain effectively.			
CO2	Evaluate the impact of Fourier Transform properties on image enhancement in the frequency domain.			
CO3	Compare and contrast error-free and lossy image compression models for efficient data storage.			
CO4	Implement adaptive thresholding techniques for image segmentation and edge detection in practice.			
CO5	Understand the core concepts and principles of pattern recognition.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction** Digital image representation, Fundamentals steps in Image Processing, Elements of DIP systems. **Digital Image Fundamentals** Elements of Visual Perception, Sampling and Quantization, Relationships between pixels, Linear and Nonlinear operations.	8	To understand digital image fundamentals.	1,2
II	**Image Enhancement in Spatial domain** Enhancement by Point Processing, Histogram Processing, Spatial Filtering.. **Image Enhancement in Frequency Domain** Introduction to the Fourier Transform, The discrete Fourier Transform, Properties of the two-dimensional Fourier Transform, Smoothing Frequency-domain filters, Sharpening Frequency domain filters.	8	Understanding Image Enhancement in Spatial domain and Frequency Domain.	3,4
III	Image Compression Fundamentals, Image Compression Models, Error Free	8	Analyzing the concepts of Image Compression Fundamentals.	3,4,5

	Compression, Lossy Compression.			
IV	Image Segmentation Threshold Techniques: Global, Adaptive and Optimum thresholding, Edge detection, Region Growing.	8	Analyzation of the concepts of Image Segmentation Threshold Techniques.	3,5
V	**Pattern Recognition** Definition and scope of pattern recognition History and applications of pattern recognition Basic concepts: patterns, features, and classifiers Overview of different approaches: statistical, syntactic, and neural Pattern recognition system design cycle	8	Analyzation of Pattern Recognition.	4,5,6

Practical Component				
Practical 1	**Implementing Basic Image Enhancement Techniques:** Perform contrast stretching and histogram equalization on grayscale and color images using Python/OpenCV.	**3**	Understand and apply basic image enhancement techniques to improve image quality.	1,2,3
Practical 2	**Spatial and Frequency Domain Filtering:** Apply spatial filters (e.g., smoothing, sharpening) and frequency domain filters (e.g., low-pass, high-pass) to enhance image features.	**3**	Analyze and enhance images using spatial and frequency domain filters.	3,4
Practical 3	**Edge Detection and Morphological Operations:** Implement edge detection techniques (Sobel, Canny) and morphological operations (dilation, erosion, opening, and closing) on binary and grayscale images.	**3**	Identify edges and refine image structure using morphological techniques.	3,4
Practical 4	**Image Segmentation Techniques:** Perform image segmentation using thresholding (global, adaptive, and Otsu's method) and region-based methods (region growing).	**3**	Segment images into meaningful regions for further analysis.	4,5
Practical 5	**Feature Extraction (Shape, Texture, and Color):** Extract and visualize shape, texture (e.g., GLCM), and color features from images using Python/OpenCV.	**3**	Understand and implement feature extraction techniques for image representation.	3,4

Practical 6	**Dimensionality Reduction with PCA and LDA:** Implement Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) to reduce feature dimensions of image datasets.	**3**	Reduce the dimensionality of datasets for computational efficiency in pattern recognition tasks.	4,5
Practical 7	**Classifier Development for Pattern Recognition:** Develop and evaluate classifiers (e.g., SVM, k-NN) for pattern recognition tasks such as digit or character recognition.	**3**	Train and evaluate classifiers for recognizing patterns in image datasets.	4,5
Practical 8	**Convolutional Neural Networks (CNNs) for Image Classification:** Implement CNNs for image classification tasks using TensorFlow/Keras, and visualize feature maps.	**3**	Develop CNN models to classify images and understand their underlying mechanics.	3,5
Practical 9	**Transfer Learning Using Pre-trained Models:** Apply transfer learning using pre-trained models like VGG16 or ResNet to classify images in custom datasets.	**3**	Utilize pre-trained models for efficient and accurate image classification.	3,5
Practical 10	**Object Detection and Real-Time Processing:** Develop an object detection system using YOLO or Faster R-CNN, and test it on real-time video feeds.	**3**	Design and implement an object detection system for practical applications.	5,6

TEXT BOOKS:

T1: Rafael C Gonzalez, Richard E Woods,―Digital Image Processing, Pearson Education Publications.

T2: Rajjan Shinghal - Pattern Recognition, Oxford Publications.

T3: Chanda and Majumder,―Digital Image Processing and Analysis, Prentice Hall Publications.

REFERENCEBOOKS:

R1: Rafael C Gonzalez, Richard E Woods,―Digital Image Processing with Matlab, Pearson Education Publications.

R2: S. Sridhar,―Digital Image Processing, Oxford University Press.

R3: Jayaraman,―Digital Image Processing, McGraw Hill.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping

SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand spatial filtering techniques to enhance digital images in the spatial domain effectively.	CO1,CO2,CO3, CO5,CO10,CO12
2	Evaluate the impact of Fourier Transform properties on image enhancement in the frequency domain.	CO1,CO2,CO3, CO5,CO10,CO12
3	Compare and contrast error-free and lossy image compression models for efficient data storage.	CO1,CO2,CO3, CO5,CO10,CO12
4	Implement adaptive thresholding techniques for image segmentation and edge detection in practice.	CO1,CO2,CO3, CO4, CO5,CO10,CO12
5	Understand the core concepts and principles of pattern recognition.	CO1,CO2,CO3, CO4, CO5,CO10,CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS4202R	Image Processing and Pattern Recognition	CO1	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VIII									
Course Title	**Project II**								
Course code	**24BTCS4203R**	**Total credits: 6**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 144	**0**	**0**	**0**	**12**	**16**	**0**	**6**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	4. To equip students with the IT principles and programming skills. 5. To equip students with team management abilities necessary to develop and implement software, hardware, and network solutions effectively. 6. To cultivate employability skills and professionalism in software development.								
CO1	Apply IT principles for real-world problem-solving using programming languages and network solutions.								
CO2	Manage complex projects as a team member, demonstrating effective personal and team management skills.								
CO3	Develop programming skills to create and assess software, hardware, and network solutions.								
CO4	Apply personal and team management skills as a professional software developer.								
CO5	Cultivate employability skills and uphold professionalism in software development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a) theoretical studies b) computer simulations c) hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;	144	To understand, analyze, manage, develop and cultivate programming skills to create and assess software, hardware, and network solutions.	1,2,3,4,5

	4. Deliver a presentation on the general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	CO1,CO2,CO3, CO4, CO5,CO6, CO 7, CO8, CO9, CO10, CO12
2	Manage complex projects as a team member, demonstrating effective personal and team management skills.	CO1,CO2,CO3, CO4, CO5,CO6, CO 7, CO8, CO9, CO10, CO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	CO1,CO2,CO3, CO4, CO5,CO6, CO 7, CO8, CO9, CO10, CO12
4	Apply personal and team management skills as a professional software developer.	CO1,CO2,CO3, CO4, CO5,CO6, CO 7, CO8, CO9, CO10, CO12
5	Cultivate employability skills and uphold professionalism in software development.	CO1,CO2,CO3, CO4, CO5,CO6, CO 7, CO8, CO9, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BTCS4203R	PROJECT-II	CO1	3	3	2	3	2	1	1	2	3	2	2	3
		CO2	3	3	3	3	2	1	1	2	3	2	2	3
		CO3	3	2	3	2	3	1	1	2	3	3	2	3
		CO4	3	2	2	2	2	1	1	2	3	3	3	3
		CO5	3	3	3	3	3	1	3	2	3	3	2	3

Assam down town University

Curriculum and Syllabus

Bachelor of Technology
in
Computer Science and Engineering
On
Data Science & Artificial Intelligence

OUTCOME BASED EDUCATION FRAMEWORK
CHOICE BASED CREDIT SYSTEM
Version: 2.2

FACULTY OF COMPUTER TECHNOLOGY

July, 2024

PREAMBLE

Assam down town University is a premier higher educational institution which offers Bachelor, Master, and Ph.D. degree programmes across various faculties. These programmes, collectively embodies the vision and mission of the university. In keeping with the vision of evolutionary changes taking place in the educational landscape of the country, the university has restructured the course curriculum as per the guidelines of National Education Policy 2020. This document contains outline of teaching and learning framework and complete detailing of the courses. This document is a guidebook for the students to choose desired courses for completing the programme and to be eligible for the degree. This volume also includes the prescribed literature, study materials, texts, and reference books under different courses as guidance for the students to follow.

Recommended by the Board of Studies (BOS) meeting of the Faculty of Computer Technology held on dated 15/07/2024 and approved by the 51st Academic Council (AC) meeting held on dated 26/07/2024

Chairperson, Board of Studies

Member Secretary, Academic Council

Vision

To become a Globally Recognized University from North Eastern Region of India, dedicated to the Holistic Development of Students and Making Society Better

Missions

1. Creation of curricula that address the local, regional, national, and international needs of graduates, providing them with diverse and well-rounded education.

2. Build a diverse student body from various socio-economic backgrounds, provide exceptional value-based education, and foster holistic personal development, strong academic careers, and confidence.

3. Achieve high placement success by offering students skill-based, innovative education and strong industry connections.

4. Become the premier destination of young people, desirous of becoming future professional leaders through multidisciplinary learning and serving society better.

5. Create a highly inspiring intellectual environment for exceptional learners, empowering them to aspire to join internationally acclaimed institutions and contribute to global efforts in addressing critical issues, such as sustainable development, Climate mitigation and fostering a conflict-free global society.

6. To be renowned for creating new knowledge through high quality interdisciplinary research for betterment of society.

7. Become a key hub for the growth and excellence of AdtU's stakeholders including educators, researchers and innovators

8. Adapt to the evolving needs and changing realities of our students and community by incorporating national and global perspectives, while ensuring our actions are in harmony with our foundational values and objectives of serving the community.

Programme Details

Programme Overview (not more than 100 words)

Bachelor of Technology in Computer Science and Engineering is an Under Graduate Degree awarded for the programme of Computer Science and Engineering. As one of the best computer engineering colleges, we intend to create a cohesive learning experience with the latest technological developments to that of industry demand.

I. Specific Features of the Curriculum

The B. Tech CSE curriculum, guided by the outlined Program Specific Outcomes (PSOs), emphasizes a comprehensive educational framework. It prepares students for competitive exams through updated and focused syllabi. It integrates advanced research principles with sustainable technology applications, ensuring graduates possess the necessary engineering and management skills for societal impact. The curriculum fosters effective communication, teamwork, and project management in multidisciplinary settings. Additionally, it incorporates contemporary industrial and research trends, encouraging innovation and entrepreneurial proficiency. This holistic approach equips students to develop novel software engineering solutions, addressing real-world problems with cutting-edge techniques and strategies.

II. Eligibility Criteria:

Minimum 45% with English, Physics & Mathematics in HS.

III. Program Educational Objectives (PEOs):

PEO1	To produce graduates who have strong foundation of knowledge and skills in the field of Computer Science and Engineering.
PEO2	To produce graduates who can provide solutions to challenging problems in their profession by applying Computer Engineering theory and practices.
PEO3	To produce graduates who are employable in industries/public sector/research organizations or work as an entrepreneur, as well as can provide leadership and are effective in multidisciplinary environment.

IV. Program Specific Outcomes (PSOs):

PSO 1: Educational Advancement: Graduates will be prepared for state, national and international competitive examinations with focused and updated syllabi.

PSO 2: Advanced Research: Graduates will have a holistic comprehension of engineering and management principles required for application of sustainable technologies for societal

development, and also will be able to communicate effectively in oral, written, visual and graphic modes as a member and leader in a team, to manage projects in multidisciplinary environments.

PSO 3: Entrepreneurial Proficiency: Graduates will be acquainted with the contemporary trends in industrial/research environments and also will capable of innovating novel solutions to prevailing problems by applying software engineering techniques and strategies.

V. Program Outcome: (8-12)

PO1 Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2 Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4 Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

VI. Total Credits to be Earned:

The total credit to be earned is 173

VII. Career Prospects:

A B. Tech in Computer Science and Engineering (CSE) offers diverse and promising career prospects. Graduates can pursue roles such as software developers, data scientists, cybersecurity analysts, and AI/ML engineers in top tech firms. They are well-suited for positions in research and development, working on cutting-edge technologies. The curriculum also prepares them for competitive exams, enabling careers in public sector and government organizations. Additionally, the strong emphasis on entrepreneurial skills allows graduates to innovate and launch their own start-ups. With rapid advancements in technology, CSE graduates remain in high demand across various industries, ensuring robust career opportunities.

EVALUATION METHODS

The student performance shall be evaluated through In-semester (Sessional) and semester-end examinations. A weightage of 40% or as prescribed by the programme shall be added to the score of the end-semester examination.

A. INTERNAL ASSESSMENT:

The teacher who offers the course shall be responsible for internal assessment by conducting in-semester (sessional) examination and evaluating the performance of the students pursuing that course. The components for internal assessment are illustrated in the table given below.

Table 1: Question paper pattern for End semester examination

SN	Components/ Examinations	Marks Allotted
1.	In-Sem Exam – I (ISE-I) (Written Examination)*	30
2.	In-Sem Exam – II (ISE-II) (Written Examination)*	30
3.	Assignment	10
4.	Presentation (SP)	10
5.	Quiz	5
6.	Class Performance based score*	5

are compulsory

Note: *Total Internal assessment should be out of 40*

INSTRUCTION

1. If a student fails to appear in the any of the component without any valid reason he/she shall be marked zero in that component. However, the course teacher at his discretion may arrange for the missed test on an alternate date for the absentee students after determining ground with genuine/valid reasons for the absent.

2. The report of evaluation of an activity towards the in-semester (sessional) component of a course shall be duly notified by the concerned course teacher within a week of completion.

3. The program coordinators should upload the in-semester marks to the ERP and forward acknowledgement of all the courses of the program to the Controller of Examinations before the start of the End-semester examination.

B. SEMESTER END EXAMINATION:

Time table for end semester examination is published at least 25 days prior to the start of

Examination.

I. Pre-Examination:

Eligibility Criteria for a student to appear in University Examinations:

The student shall only be allowed to appear in a University Examination, if:

i) He/ She is a registered student of the University;

ii) He/ She is of good conduct and character;

iii) He/ She has completed the prescribed Programme of study with minimum percentage of attendance as laid down in the Regulations of the Programme concerned.

Under special cases, a student may be allowed to appear for an examination without being registered in the University but the result of the said student will be kept on hold till the registration of the concerned student is completed.

II. Admit Card:

Admit card for the examination may be downloaded through ERP where the system will generate a Unique ID Cards through online.

The University shall have the right to cancel admission for examination of any candidate on valid grounds.

III. Pattern of Question Papers:

The question paper shall follow the principles of Bloom's Taxonomy.

Table 2: Question paper pattern for End semester examination

S. N.	Level	Questions /verbs for test
1	Remember	List, Define, tell, describe, recite, recall, identify, show who, when, where, etc.
2	Understand	Describe, explain, contrast, summarize, differentiate, discuss, etc.
3	Apply	Predict, apply, solve, illustrate, determine, examine, modify
4	Analyze	Classify, outline, categorize, analyze, diagrams, illustrate, infer, etc.
5	Evaluate	Assess, summarize, choose, evaluate, recommend, justify, compare etc.
6	Create	Design, Formulate, Modify, Develop, integrate, etc.

Note: No course is to be evaluated on basis of **all 6 knowledge levels.**

The format of the question paper across all the program follow a unique pattern and the total marks is 60

Table 3: Question paper pattern for End semester examination

Sl no	Question pattern	Total marks
1	MCQs (10 Questions)	10

2	2 Marks questions (10 Questions)	20
3	4 Marks questions (5 Questions)	20
4	10 Marks questions (1 Question)	10

IV. Examination Duration:

Each paper of 60 marks shall ordinarily be of two hours duration.

V. Practical Examinations, Viva-Voce etc.:

i) Practical examination shall be conducted in the presence of one external expert and one or more internal examiners.

ii) Viva-Voce, Oral examinations of the Project report, Dissertation etc. shall be undertaken by a Board of Examiners constituted by the respective Dean of Program with the advice of Supervisor(s).

VI. Procedure of Expulsion:

If any candidate is found to be using any unfair-means during the examination, the invigilator may cease his/her answer sheet and report it directly to the Officer-in-Charge. The Office-in-Charge of the center may take appropriate decisions as per the rules and procedure of the examination. The Officer-in-Charge may allow the students to write the exam with new answer sheet or may expel the student from appearing the paper depending on the nature of unfair-means. In case of Computer based test, the students may be directed to write an apology letter and sign in the prescribe expulsion form. The student may not be allowed to write that examination.

VII. Instruction to the Students:

(i) The students shall not bring to the Examination Hall, any electronic gadget used as a means of communication or record except electronic calculator, if required.

(ii) The students shall not receive any book or printed or hand written or photo copy (Xerox) or blank-paper from any other person while he/she is in the examination-room or in laboratory or in any other place to which he/she is allowed to have access during course of examination.

(iii) The students shall not communicate with any other candidate in the examination room or with any other person in and outside the examination-room.

(iv) The students shall not see, read or copy anything written by any other candidate, nor shall he/she knowingly or negligently permit any other candidate to see, read or copy anything written by him/her or conveyed by him/her.

(v) The students shall not write anything on the Question Paper or in other paper or materials during the examination, or pass any kind of paper to any other candidate in the examination-room, or to any person outside the room.

(vi) The students shall not disclose his/her identity to the examiner by writing his/her name or putting any sign / symbol in any part of his answer-script.

(vii) The students shall not use any abusive language or write any objectionable remark or make any appeal to examiner by writing in any part of his answer-script.

(viii) The students shall not detach any page from the answer-script or insert any authorized or unauthorized loose sheet into it. He /she shall also not insert any other answer-script / loose sheet by removing the pins of the origin answer-scripts and re-fixing it.

(ix) The students shall not resort to any disorderly conduct inside the examination-room or misbehave with the invigilator or any other examination official.

VIII. Provision for an Amanuensis (writer):

(i) A candidate may be provided with an Amanuensis (writer) to write down on dictation on his / her behalf on ground of his / her physical disability to write down by himself / herself due to accident or any other reason. The amanuensis may be provided till he / she recovers from the physical disability. The physical disability to write down by himself / herself must be supported by Medical Certificate from a competent Medical Officer.

(ii) The qualifications of the amanuensis so provided must not be equal or higher than that of the candidate. This is also to be supported by Certificate from the Faculty of Study where the Amanuensis is provided.

(iii) Such candidates are to be accommodated in a separate room under the supervision of an invigilator so that the fellow candidates are not disturbed in the process.

C. Credit Point:

It is the product of grade point and number of credits for a course, thus, CP = GP x CR

i. Credit:

A unit by which the course work is measured. It determines the number of hours of instructions required per week. 'Credit' refers to the weightage given to a course, usually in terms of the number of instructional hours per week assigned to it. Credits assigned for a single course always pay attention to how many hours it would take for an average learner to complete a single course successfully.

ii. Grade Point:

Grade Point is a numerical weight allotted to each Grade Letter on a 10-point scale.

iii. Letter Grade:

Letter Grade is an index of the performance of students in a said paper of a particular course. Grades are denoted by letters O, A+, A, B+, B, C, P, F and Abs. Student obtaining Grade F / Grade Abs shall be considered failed/ absent and, will be required to appear in the subsequent ESE. The UGC recommends a 10-point grading system with the following (Table: 4) Letter Grades:

(i) A Letter Grade shall signify the level of qualitative/quantitative academic achievement of a student in a Course, while the Grade Point shall indicate the numerical weight of the Letter Grade on a 10-point scale.

(ii) There shall be 08 (eight) Letter Grades bearing specific Grade Points as listed in Table 4, where the Letter Grades 'O' to 'P' shall indicate successful completion of a course.

(iii) Apart from the 08 (eight) regular Letter Grades listed in Table 4, there shall be 03 (three) additional Letter Grades, which shall be awarded if a Course is withdrawn or spanned over the next Semester or remains incomplete.

Table 4: Letter Grades and Grade Points

Letter Grade	Grade Points	Description
O	10	Outstanding
A+	9	Excellent
A	8	Very Good
B+	7	Good
B	6	Above Average
C	5	Average
P	4	Pass
F	0	Fail
Abs	0	Absent
UFM	0	Unfair Means

iv. Grade Point Average:

a. SGPA (Semester Grade Point Average)

The SGPA of a student in a Semester shall be the weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered in that Semester, irrespective of whether he/she could or could not complete the Courses. More specifically, the calculation of SGPA shall take into account the Courses graded with Letter Grades 'O' to 'F' as given in Table 4.

$$\text{SGPA} = \frac{\Sigma_{i=1}^{n} C_i G_i}{\Sigma_{i=1}^{n} C_i} \quad (1.1)$$

The SGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.1) up to two decimal places, where n is the total number of Credit Courses registered by the student in that Semester, Gi is the Grade Point secured in the i^{th} registered Course and Ci is the Credit (weight) of that Course.

b. CGPA (Cumulative Grade Point Average)

(i) The CGPA of a student in a Semester of a Programme shall be the accumulated weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered and successfully completed so far starting from the enrolment in the Programme. In other words, taking into account all the Courses graded with 'O' to 'P' as given in Table 4, generally the CGPA of a student shall be calculated starting from the first Semester of his/her enrolled Programme, while the CGPA of a lateral-entry student shall be calculated starting from the Semester of his/her enrolment.

(ii) The CGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.2) up to two decimal places, where N is the total number of Credit Courses registered and successfully completed so far by the student, Gi is the Grade Point secured in the ith completed Course and Ci is the Credit (weight)of that Course.

$$\text{CGPA} = \frac{\Sigma_{i=1}^{N} C_i G_i}{\Sigma_{i=1}^{N} C_i} \quad (1.2)$$

(iii) The CGPA shall be convertible into equivalent percentage of marks using Equation Conversion of CGPA to percentage marks: = CGPA*10

D. Post-Examination

i. Transcript or Grade Card or Certificate:

A marking certificate shall be issued to all the registered students after every Semester. The Semester mark sheet will display the course details (code, title, number of credits, grade secured) along with total credit earned in that Semester.

ii. Grievance Readdress Mechanism:

Students with any dissatisfaction or grievance regarding the marks awarded in any of the Papers / Courses may appeal to the Controller of Examinations for remedial action such as Re-evaluation within 10 days of the declaration of result.

(i) A student has options to appeal for re-evaluation of his /her answer script to the Controller of Examination.

(ii) Application for re-evaluation / re-scrutiny of answer scripts shall be made in the definite proforma available with the Examination Office through the head of the respective departments within 10 days of declaration of the results of the respective examinations.

(iii) The Controller of Examination may appoint an examiner for re-evaluation and will consider and recognize the evaluation done by a University appointed examiner.

(iv) There shall be no provision for re-evaluation of the Practical Papers, Project Work, and Dissertation etc. However, the students fail in practical examination or viva voce and wish to appear again may apply to be evaluated can do so with the next schedule.

(v) After screening the application for re-evaluation, the CoE may send the answer scripts of the student to the examiners appointed by the CoE with the approval of Vice Chancellor.

(vi) The marks/grades achieved by the students after the re-evaluation shall be final and binding.

(vii) Fresh Marks – sheets / Grade Card shall be issued only if the candidate secures pass marks / passing grade in the re-evaluated paper.

(viii) Revaluation of answer scripts shall be deemed to be an additional facility provided to the students with a view to improving upon their results at the preceding examination result for any reason whatsoever shall not confer any right upon them for admission to next higher class which matters always be regulated in accordance with the relevant rules or regulations framed by the University.

(ix) If as a result of revaluation of the candidate attracts the provision of condonation of deficiency, the same may be applied to his/her only for fresh attempt.

INSTRUCTION TO TEACHERS AND STUDENTS

(Teaching and Learning Methods)

In all the courses the teacher has to select topics for teacher-method which should not be less than 20 percent. The approach will be direct classroom teaching through a series of lectures delivering concepts using ITC facilities, white or blackboard. Notes may also be circulated to the students; however, the students are to be involved in the preparation of the notes. The teacher will be responsible for selecting the best note for circulation. The teacher-centric methodology has recently fallen out of favour because this strategy for teaching is seen to favour passive students.

1. **Student- centric / Constructivist Approach:**

The topics of the courses may be selected at the start of the class and assigned one topic to each of the students for studying by themselves, prepare presentations, notes, etc., and present at respective class time after consultation and discussion with the course teachers. The teacher facilitates the learning of the students by guiding and providing input and explaining concepts. 60 percent of the course contents may be selected for this purpose. To avoid behaviour problems, teachers must lay a lot of groundwork in student-centric classrooms. Typically, it involves instilling a sense of responsibility in students. In addition, students must learn internal motivation.

a. Project-Based Learning: The teacher may select 5 percent of topics for the purpose and may conduct visits to the laboratory for experiments or field surveys. The selection of the topic may be done considering the available facility for the purpose. However, in the final semester of each of the programme the student has to undergo project-based learning at least 4 months duration. This approach will help the student to think critically, evaluate, analyze, make decisions, collaborate, and more.

b. Inquiry-Based Learning: The teacher/ students are supposed to list at least five questions in each contact hour and student solve these question or search for answer which becomes the home work for the students "question-driven" learning approach. The teacher may look for the correctness of the solution or the best possible answer and discuss in the successive class. This will help in the preparation for various competitive examination and develop a habit for search for solutions.

c. Flipped Classroom: About 10 percent of the course content has to be completed by this method. In this approach the students are asked to watch video or lecture prepared by the teacher or any video available (relevant to the course). A set of questions may be given to the

students for searching answers by the students. The idea is that students should have more time in-classroom focusing on achieving these higher levels of thinking and learning. The Flipped classroom is also an acronym. The letters FLIP represent the four pillars included in this type of learning: Flexible environment, Learning culture shift, Intentional content, and Professional educator. As you can see, the second pillar refers to a culture shift from the traditional approach where students are more passive to an approach where students are active participants. As a result, this approach is also a student- centric teaching method.

d. Cooperative Learning: The remaining five percent has to be completed by cooperative learning approach. In this approach, the students are allotted problems. During library hours the students along with the teacher visit the library and search for probable solutions for the assigned problem. The same has to be done in groups so that the students discuss among themselves for the appropriate answers. Essentially, cooperative learning believes that social interactions can improve learning. In addition, the approach recreates real-world work situations in which collaboration and cooperation are required.

The percentage categorization for the completion of a theory course

Teacher-centric or Direct Classroom Teaching: Delivery by series of lectures	20%
Student-centric Approach, Students present and deliver lectures in the presence of teacher and supervised by teacher	60%
Students visit fields or perform experiments or teachers perform demonstration	05%
Flipped Classroom approach	10%
Cooperative learning approach	05%

Inquiry-based approach has to be followed in all of the classes

The teacher has to distribute the topics to be considered for teaching by the above-mentioned approaches and prepare a lesson plan for execution and maintain a file.

B.Tech CSE Curriculum 2024-28 (IBM)

First Semester

Sl.No	Course Title	Course Code	Additional Category	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Linear Algebra and Univariate Calculus	24BCSM1101R	DSC (Minor)	4	3	1	0	0	0	0	40	60	0	100
2	Engineering Physics	24BCSM1102R	MDC	3	2	0	2	0	0	0	40	60	100	200
3	Basic Electrical and Electronics Engineering	24BCSM1103R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
4	Workshop for Engineers	24BCSM1104R	DSC (Minor)	1	0	0	2	0	0	0	0	0	100	100
5	1. Introduction to Java 2. International Leadership And Organizational Behaviour	24MOCS1101R 24MOCS1102R	VAC	2	0	0	0	8	0	0	100	0	0	100
6	PDP (Introductory English)	24UBPD1104R	AEC	2	0	0	4	0	0	0	0	0	100	100
7	Co Curricular Activities	24UBCC1101R	Co Curricular	1	0	0	0	4	0	0	100	0	0	100
8	Software Foundation and Programming (with C and C++)	24BCSM1105R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
Total Credit in 1 st Semester				**20**	**10**	**1**	**12**	**12**	**0**	**0**	**360**	**240**	**500**	**1100**

2nd Semester

Sl.No	Course Title	Course Code	Compulsory/Elective	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
2	Object Oriented Programming using JAVA	24BCSM1201R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Data Structure and Algorithms	24BCSM1202R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
5	Environmental Science	24UBES1201R	MDC	3	3	0	0	0	0	0	40	60	0	100
6	1. INTRODUCTION TO GIT AND GITHUB 2. REACT BASICS	24MOCS1201R 24MOCS1202R	VAC	2	2	0	0	0	0	0	100	0	0	100
7	Field Based Learning	24BCSM1203R	Field Visit	1	0	0	0	0	0	16	100	0	0	100
8	PDP	24UBPD1204R	AEC	2	0	0	4	0	0	0	0	0	100	100
9	Extra Curricular Activities	24UBEC1201	Extra Curricular Activities	1	0	0	0	4	0	0	100	0	0	100
10	Programming with Python	24BCSM1204R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
Total Credit in 2nd Semester				**21**	**14**	**0**	**10**	**4**	**0**	**16**	**460**	**240**	**400**	**1100**

3rd Semester

Sl.No	Course Title	Course Code	Compulsory/Elective	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Digital Electronics	24BCSM2101R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
3	Discrete Mathematics & Graph Theory	24BCSM2102R	DSC (Minor)	3	3	0	0	0	0	0	40	60	0	100
4	Computer Organization and Architecture	24BCSM2103R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
6	Field-Based Learning	24BCSM2104R	Field Visit	1	0	0	0	0	0	16	0	100	0	100
7	Financial Literacy (FL)	24UBPD2104R	MDC	1	0	0	2	0	0	0	0	0	100	100
8	Basic Life Saving Skills	24UULS2102R	MDC	1	0	0	2	0	0	0	0	0	100	100
9	1. Spring Framework Specialization 2. Javascript Basics	24MOCS2101R 24MOCS2102R	MDC	1	1	0	0	0	0	0	0	100	0	100
9	PDP	24UBPD2103R	AEC	2	0	0	4	0	0	0	0	0	100	100
10	Logical Reasoning for Computer Science	24BCSM2107R	SEC	2	1	0	2	0	0	0	0	0	100	100
11	Mini Project I	24BCSM2105R	DSC (Minor)	1	0	0	0	4	0	0	0	0	100	100
10	Cloud Fundamentals	24BCSM2106R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
Total Credit in 3rd Semester				**23**	**13**	**0**	**16**	**4**	**0**	**16**	**160**	**440**	**800**	**1400**

4th Semester

Sl.No	Course Title	Course Code	Compulsory/Elective	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Database Management System	24BCSM2201R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Operating Systems	24BCSM2202R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
3	Probability and Statistics	24BCSM2203R	DSC (Minor)	3	2	1	0	0	0	0	40	60	0	100
4	Formal Language and Automata Theory	24BCSM2204R	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100
5	PDP	24BCSM2205R	AEC	2	0	0	2	0	0	0	0	0	100	100
6	Quantitative Aptitude For Computer Science	24BCSM2206R	SEC	2	1	0	2	0	0	0	0	0	100	100
7	Digital Literacy	24UCDL2202R	VAC	1	0	0	2	0	0	0	0	0	100	100
8	Basic Acclimitizing Skills	24UULS2203R	VAC	1	0	0	2	0	0	0	0	0	100	100

9	Mini Project II	24BCSM2207R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100
10	Data Visualization	24BCSM2208R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
Total Credit in 4th Semester				**25**	**15**	**1**	**14**	**4**	**0**	**0**	**300**	**300**	**700**	**1300**

5th Semester

Sl.No	**Course Title**	**Course Code**	**Compulsory/Elective**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Data Communication and Computer Network	24BCSM3101R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Signals and Systems	24BCSM3102R	DSC (Minor)	3	2	0	2	0	0	0	40	60	100	200
3	UI/UX Design	24BCSM3103R	DSC (Major)	4	2	0	2	4	0	0	40	60	0	100
4	Software Engineering	24BCSM3104R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
6	PDP	24UBPD3104R	AEC	2	0	0	4	0	0	0	0	0	100	100
7	Design Thinking and Enterpreuship	24BCSM3105R	SEC	2	1	0	2	0	0	16	0	0	100	100
9	MOOCS / CBCS	24MOCS3101R 24MOCS3102R	VAC	1	0	0	0	4	0	0	0	0	100	100
10	Essence of Indian Traditional Knowledge	24BCSM3105R	VAC	2	2	0	0	0	0	0	40	60	0	100
11	Mini Project-III	24BCSM3106R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100
12	Predictive Analysis	24BCSM3107R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
Total Credit in 5th Semester				**27**	**16**	**0**	**16**	**12**	**0**	**16**	**340**	**360**	**700**	**1400**

6th Semester

Sl.No	**Course Title**	**Course Code**	**Compulsory/Elective**	**Credit**	**L**	**T**	**P**	**S**	**R**	**O**	**Maximum Marks**			**Total**
											IA	**SEE**	**PE**	
1	Web Technologies	24BCSM3201R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Compiler Design	24BCSM3202R	DSC (Major)	4	3	1	0	0	0	0	40	60	0	100
3	Design and Analysis of Algorithm	24BCSM3203R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Statistical Methods and Modelling	24BCSM3204R	DSC (Minor)	3	3	0	2	0	0	0	40	60	100	200
5	Mini Project-IV	24BCSM3205R	DSC (Minor)	1	0	0	0	4	0	0	100	0	0	100
6	Foundations of Data Science	24BCSM3206R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
7	PDP		AEC	2	0	0	4	0	0	0	0	0	100	100
Total Credit in 6th Semester				**22**	**15**	**1**	**12**	**4**	**0**	**0**	**300**	**300**	**500**	**1100**

7th Semester

Sl.No	Course Title	Course Code	Compulsory/Elective	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	PE - I	24BCSM4101R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
2	Project-I	24BCSM4102R	DSC (Major)	6	0	0	0	16	12	0	100	200	0	300
3	PE - II	24BCSM4103R	DSC (Major)	4	3	0	2	4	0	0	100	0	100	200
4	Summer Internship	24BCSM4104R	Internship	4	0	0	0	16	0	0	0	100	0	100
Total Credit in 7th Semester				**18**	**6**	**0**	**4**	**36**	**12**	**0**	**240**	**360**	**200**	**800**

8th Semester

Sl.No	Course Title	Course Code	Compulsory/Elective	Credit	L	T	P	S	R	O	Maximum Marks			Total
											IA	SEE	PE	
1	Cryptography & Network Security	24BCSM4201R	DSC (Major)	3	3	0	0	0	0	0	40	60	0	100
2	Machine Learning	24BCSM4202R	DSC (Major)	4	3	0	2	4	0	0	40	60	100	200
3	Image Processing and Pattern Recognition	24BCSM4203R	DSC (Major)	4	3	0	2	0	0	0	40	60	100	200
4	Project-II	24BCSM4204R	DSC (Major)	6	0	0	0	12	18	0	100	200	0	300
Total Credit in 8th Semester				**17**	**9**	**0**	**4**	**16**	**18**	**0**	**220**	**380**	**200**	**800**

***IA: Internal Assessment,**
***SEE: Semester End Examination,**
***PE: Practical Examination**

List of Program Electives (PE)Paper		
Semester	**Program Electives No.**	**Course Name**
VII	PE-I	1. Neural Networks 2. Artificial Intelligence
VII	PE-II	1. Data Mining and Analytics 2. Information Retrieval Systems

Semester	Credit	L	T	P	S	R	O	Maximum Marks			Total
								IA	SEE	PE	
1st Semester	20	10	1	12	12	0	0	360	240	500	1100
2nd Semester	21	14	0	10	4	0	16	460	240	400	1100
3rd Semester	23	13	0	16	4	0	16	160	440	800	1400
4th Semester	25	15	1	14	4	0	0	300	300	700	1300
5th Semester	27	16	0	16	12	0	16	340	360	700	1400
6th Semester	22	15	1	12	4	0	0	300	300	500	1100
7th Semester	18	6	0	4	36	12	0	240	360	200	800
8th Semester	17	9	0	4	16	18	0	220	380	200	800
Total	**173**	**98**	**3**	**88**	**92**	**30**	**48**	**2380**	**2620**	**4000**	**9000**

List of MOOCS				
Sl. No	**MOOCS Course No**	**Name of the Course**	**Course Code**	**Semester**
1	MOOCS I	Introduction to Java	24MOCS1101R	I
2		International Leadership And Organizational Behaviour	24MOCS1102R	
1	MOOCS II	Introduction To GIT And GITHUB	24MOCS1201R	II
2		React Basics	24MOCS1202R	
1	MOOCS III	Spring Framework Specialisation	24MOCS2101R	III
2		Javascript Basics	24MOCS2102R	
1	MOOCS IV	Introduction to Mobile Development	24MOCS3101R	V
2		Introduction to the Internet of Things and Embedded Systems	24MOCS2102R	

SEMESTER – I									
Course Title	**Linear Algebra and Univariate Calculus**								
Course code	**24BCSM1101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 48T	**3**	**1**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Mathematics	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. To make understand to evaluate definite and improper integrals. Apart from these some special functions like Beta and Gamma function must be introduced. 2. To provide the application of differential and integral calculus. 3. To make understand the convergence and divergence of sequence and series								
CO1	The students will be able to demonstrate the ability to understand the concepts of mathematics, logic, reasoning and coding								
CO2	Students will be able to apply of applications of concepts in other disciplines such as engineering, computer science, physics, etc.								
CO3	Solve and devise solutions to a range of elementary real-world problems in mathematics and programming								
CO4	Explore and apply key concepts in logical thinking to business problems.								
CO5	Enable students to critically analyze information to evaluate evidence and construct reasoned arguments.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Calculus:** Evaluates and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions.	10	Evaluate definite and improper integrals and understand their properties. Apply Beta and Gamma functions in various contexts. Use definite integrals to calculate surface areas and volumes of revolution.	1,2,3
II	**Calculus:** Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; indeterminate forms and L'Hospital's rule; Maxima and minima.	10	Understand and apply Rolle's Theorem and the Mean Value Theorem to determine properties of functions. Utilize Taylor and Maclaurin series expansions with remainders for function approximation. Evaluate indeterminate forms using L'Hospital's Rule, and identify and analyze local maxima and minima of functions.	3,4
III	**Sequences and series:** Convergence of sequence and series, tests for convergence; Power series, Taylor's series, series for exponential, trigonometric and logarithm functions; Fourier series: Half range sine and cosine series, Parseval's theorem.	10	Analyze the convergence of sequences and series using various tests. Develop and utilize power series, including Taylor's series, to represent functions such as exponential,	3,4

			trigonometric, and logarithmic functions. Apply Fourier series to decompose functions into their sine and cosine components and understand Parseval's theorem in the context of these series.	
IV	**Multivariable Calculus (Differentiation):** Limit, continuity and partial derivatives, directional derivatives, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers; Gradient, curl and divergence.	9	Understand and apply the concepts of limits, continuity, and partial derivatives, including directional and total derivatives. Analyze and solve problems involving tangent planes, normal lines, and optimization techniques such as maxima, minima, saddle points, and the method of Lagrange multipliers. Compute and interpret the gradient, curl, and divergence of vector fields.	4,5
V	**Matrices:** Inverse and rank of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Determinants; Eigenvalues and eigenvectors; Diagonalization of matrices; Cayley- Hamilton Theorem, and Orthogonal transformation.	9	Understand and calculate the inverse and rank of matrices, and apply the rank-nullity theorem. Solve systems of linear equations using matrix methods. Identify and work with symmetric and skew-symmetric matrices.	3,5

TEXT BOOKS:

T1: G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint,2002.

T2: Erwin kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley &Sons, 2006.

T3: Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi,11th Reprint, 2010

EFERENCE BOOKS:

R1:N.P. Bali and Manish Goyal, A text book of Engineering Mathematics,Laxmi Publications, Reprint,2008.

R2: B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition,2010.

OTHER LEARNING RESOURCES:

O1: Coursera online platform for Mathematical learning

O2: Math kind Quality *Math Education* portal

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	The students will be able to demonstrate the ability to understand the concepts of mathematics, logic, reasoning and coding.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
2	Students will be able to apply of applications of concepts in other disciplines such as engineering, computer science, physics, etc.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
3	Solve and devise solutions to a range of elementary real-world problems in mathematics and programming.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
4	Explore and apply key concepts in logical thinking to business problems.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12
5	Enable students to critically analyze information in order to evaluate evidence and construct reasoned arguments.	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PO11,PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM 1101R	**Linear Algebra and Univariate Calculus**	**CO1**	3	2	0	0	3				2	0	0	2
		CO2	3	3	0	0	2				0	0	0	2
		CO3	0	3	3	3	0				2	0	0	0
		CO4	2	0	3	0	0				0	3	2	1
		CO5	2	3	2	0	0				2	2	2	1

SEMESTER – I									
Course Title	**Engineering Physics**								
Course code	**24BCSM1102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:40T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**

Pre-requisite	Basic knowledge of Physics	**Co-requisite**	Mathematics	
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**			
Semester	**Summer/ I semester of the first year of the Programme**			
Course Objectives (Minimum 3)	1. To understand the theories of physics 2. To apply the concepts in practical problems 3. To understand the physics of any process			
CO1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.			
CO2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.			
CO3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.			
CO4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.			
CO5	Understand the basic characteristics of transformers and electrical machines.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Electrostatics in vacuum	**8**	Understand and apply Coulomb's Law to calculate the electrostatic force between point charges in a vacuum. Analyze electric fields and potentials created by various charge distributions using principles of electrostatics.	**2**
II	Magneto statics	**8**	Magnetostatics studies magnetic fields in systems with steady currents, focusing on the behavior and interactions of magnetic fields in static conditions. It explains phenomena like the magnetic field produced by a current-carrying conductor and the forces between magnetic dipoles.	**3**
III	Faraday's law	**8**	Faraday's law of electromagnetic induction states that a change in magnetic flux through a circuit induces an electromotive force (EMF) in the circuit. This induced EMF is proportional to the rate of change of the magnetic flux.	**3,5**
IV	Displacement current, Magnetic field due to time-dependent electric field and Maxwell's equations.	**8**	Understand the concept of displacement current as a term added to Ampere's law to account for the changing electric field in non-conductive regions. Recognize its role in ensuring the continuity of current in Maxwell's equations and the propagation of electromagnetic waves.	**3**
V	Electromagnetic waves	**8**	Electromagnetic waves are transverse waves composed of mutually perpendicular electric and magnetic fields oscillating perpendicular to the direction of wave propagation. They encompass a wide range of frequencies, from radio waves to gamma rays, each	**5**

			characterized by its wavelength and frequency.	
Practical Component				
Practical 1	To verify the ohm's law and hence determine the unknown resistance of the given material of the wire.	**4**	Verification of Ohm's Law through experimental data to determine the resistance of a given wire material	3
Practical 2	To find the value of a given resistance by using meter bridge.	**5**	Understanding how to determine the resistance of a given resistor using a meter bridge based on null point detection	4
Practical 3	To convert the galvanometer in to voltmeter and hence calibrate it with a standard resistance and ammeter.	**5**	To convert a galvanometer into a voltmeter, learn to connect a high resistance in series with the galvanometer and calibrate it using a standard resistance and ammeter to measure the voltage accurately.	5
Practical 4	To determine the internal resistance of a cell by using potentiometer.	**6**	Understand and apply the principle of a potentiometer to accurately measure the internal resistance of a cell	3
Practical 5	To determine the frequency of a tuning fork by Melde's apparatus.	**5**	Determine the frequency of a tuning fork using Melde's apparatus by observing the formation of stationary waves on a stretched string and 2calculating it from the wave parameters.	2
Practical 6	To determine the moment of inertia of a body about an axis passing through its center of gravity and perpendicular to its length.	**2**	To determine the moment of inertia of a body about an axis passing through its center of gravity and perpendicular to its length, the learning outcome is: "Calculate the moment of inertia using integral calculus, considering the mass distribution and geometric properties of the body."	4
Practical 7	To determine the ECE of copper by using copper voltmeter and ammeter.	**3**	To determine the Electrochemical Equivalent (ECE) of copper using a copper voltmeter and ammeter, students will measure the mass of copper deposited during electrolysis and relate it to the charge passed through the solution to calculate the ECE.	5

TEXT BOOKS:

T1: Introduction to Electrodynamics. David Griffiths. Prentice Hall, Upper Saddle River, New Jersey, 07458

T2: Basic Laws of Electromagnetism. IE IRODOV

REFERENCE BOOKS:
R1: Principles of physics. Halliday Resnick.

OTHER LEARNING RESOURCES:
O1: Coursera Physics Courses
O2: edX Physics Courses
O3: MIT Open Course Ware Physics

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.	PO1, PO2,PO3,PO4,PO5,PO10,PO12
5	Understand the basic characteristics of transformers and electrical machines.	PO1, PO2,PO3,PO4,PO5,PO10,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BCSM1102R	Engineering Physics	CO1	3	3	2	1	2						1	
		CO2	3	3	2	1	2						1	
		CO3	3	3	3	1	2						1	
		CO4	3	3	2	1	1						1	
		CO5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**Basic Electrical and Electronics Engineering**								
Course code	**24BCSM1103R**	**Total credits:3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives	1. To introduce the fundamental principles and concepts of electrical and electronic systems. 2. To develop the ability to analyze and solve basic electrical and electronic circuits. 3. To familiarize students with the working principles of electrical machines and semiconductor devices. 4. To provide an understanding of digital logic concepts and their applications in designing digital systems. 5. To impart knowledge of measurement techniques and instruments used in electrical and electronic systems.								
CO1	Understand and apply the basic principles of electrical circuits and AC/DC analysis.								
CO2	Demonstrate knowledge of the construction and working principles of electrical machines and devices.								
CO3	Analyze and design basic electronic circuits using diodes, transistors, and operational amplifiers.								
CO4	Apply digital logic techniques to solve problems and design simple digital circuits.								
CO5	Select appropriate measuring instruments for electrical and electronic applications and explain their working principles.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Basics of Electrical Engineering :** Fundamental Concepts: Charge, Current, Voltage, Power, and Energy. DC Circuits: Ohm's Law, Kirchhoff's Laws, Series and Parallel Circuits, Mesh and Nodal Analysis. AC Fundamentals: Sinusoidal Waveforms, RMS and Average Values, Power Factor, Single-Phase and Three-Phase Circuits	**7**	Understand the fundamental electrical concepts and laws. Analyze and solve basic DC and AC circuit problems.	**2,3**
II	**Electrical Machines:** Transformers: Principle, Construction, EMF Equation, and Efficiency. DC Machines: Construction, Working Principle, and Applications of DC Generators and Motors. Induction Motors: Basics of Single-Phase and Three-Phase Induction Motors.	**8**	Explain the working principles of electrical machines. Apply knowledge of electrical machines to real-world applications.	**2, 3**
III	**Basics of Electronics:** Semiconductor Theory: P-N Junction Diode, V-I Characteristics, Zener Diode. Transistors: Bipolar Junction Transistor (BJT) - Configuration, Characteristics, and Applications.	**7**	Understand the behaviour and characteristics of basic semiconductor devices. Utilize semiconductor devices in simple circuits.	**2, 3,4**

	Basics of Integrated Circuits and Operational Amplifiers (Op-Amps): Working and Applications.			
IV	**Digital Electronics:** Number Systems: Binary, Octal, Decimal, and Hexadecimal. Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR - Truth Tables and Applications. Boolean Algebra and Simplification of Boolean Expressions. Basic Digital Circuits: Half Adder, Full Adder, Multiplexers, and Flip-Flops.	**7**	Understand and apply digital logic concepts in basic circuits. Design simple digital circuits for computational tasks.	**2,3,4**
V	**Electrical and Electronic Measurements:** Measurement Basics: Measurement of Current, Voltage, Power, and Energy. Instruments: Working Principles of Ammeters, Voltmeters, Watt meters, and Multi meters	**7**	Explain the working principles of measurement devices. Select appropriate instruments for various electrical and electronic measurements.	**2,3,4**
Practical Component				
Practical 1	Verification of Ohm's Law	**2**	Verify Ohm's Law and understand its application in DC circuits.	1,2
Practical 2	Series and Parallel Circuit Analysis	**2**	Analyze series and parallel resistive circuits and compute voltage, current, and power.	3,4
Practical 3	Kirchhoff's Voltage and Current Laws	**2**	Validate Kirchhoff's Laws and apply them to solve electrical networks.	2,3
Practical 4	Measurement of Power and Power Factor in AC Circuits	**3**	Measure power and power factor in single-phase AC circuits using appropriate instruments.	3
Practical 5	Open-Circuit and Short-Circuit Tests on a Transformer	**3**	Perform OC and SC tests on a transformer and calculate efficiency and voltage regulation.	3,4
Practical 6	Speed Control of a DC Motor	**3**	Analyze the speed control methods of DC motors using armature and field control techniques.	1,2
Practical 7	Characteristics of PN Junction and Zener Diode	**3**	Understand the V-I characteristics of PN junction and Zener diodes and their applications.	1,3
Practical 8	Input and Output Characteristics of a BJT in CE Configuration	**3**	Analyze the input and output characteristics of a BJT and determine its parameters in CE mode.	2,3
Practical 9	Design and Implementation of Half-Wave and Full-Wave Rectifiers	**3**	Design and analyze rectifier circuits for AC to DC conversion and understand ripple factor.	3

Practical 10	Realization of Logic Gates Using Universal Gates	**2**	Construct and validate the truth tables of basic gates using NAND and NOR gates.	3,4
Practical 11	Implementation of Boolean Functions Using Logic Gates	**2**	Simplify and implement Boolean expressions using basic logic gates.	3
Practical 12	Measurement of Electrical Quantities Using Multi-meter	**2**	Use a digital multi-meter to measure voltage, current, resistance, and continuity.	3,4

Textbooks:

T1 : Hughes, E., Smith, I. M., Hiley, J., & Brown, K. "Electrical and Electronic Technology", 12th Edition, Pearson Education, 2016.

T2: Boylestad, R. L., & Nashelsky, L. "Electronic Devices and Circuit Theory", 11th Edition, Pearson Education, 2015.

T3: Mittle, V. N., & Mittal, A. "Basic Electrical Engineering", McGraw-Hill Education, 3rd Edition, 2017.

Reference Books:

R1 : Hayt, W. H., Kemmerly, J. E., & Durbin, S. M. "Engineering Circuit Analysis", 8th Edition, McGraw-Hill Education, 2012.

R2: Sedra, A. S., & Smith, K. C. "Microelectronic Circuits", 7th Edition, Oxford University Press, 2014.

R3: Floyd, T. L. "Digital Fundamentals", 11th Edition, Pearson Education, 2015.

R4: Sawhney, A. K. "A Course in Electrical and Electronic Measurements and Instrumentation", Dhanpat Rai & Co., 2015.

Other Learning Resources:

O1: NPTEL Lectures on Basic Electrical Circuits by Prof. L. Umanand, IISc Bangalore.
O2: NPTEL Lectures on Digital Circuits by Prof. S. Srinivasan, IIT Madras.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand Kirchoff's laws to solve complex DC circuits, employing Thevenin, Norton, and Superposition theorems effectively.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
2	Analyze single-phase AC circuits, applying phasor representation and power concepts with precision and understanding.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
3	Evaluate transformer performance, considering losses, regulation, and efficiency, and apply knowledge to three-phase transformer connections.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
4	Critically analyze three-phase induction motor operation, interpreting torque-slip characteristics, losses, and efficiency for practical applications.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12
5	Understand the design of power converters and demonstrate expertise in electrical installations, including safety devices and battery types.	PO1,PO2,PO3,PO4,PO5,PO6,PO9,PO10,PO11,PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM1103R	Basic Electrical and Electronics Engineering	CO 1	3	3	3	2	2	1			1	2	1	3
		CO 2	3	3	3	2	2	1			1	2	1	3
		CO 3	3	3	3	2	2	1			1	2	1	3
		CO 4	3	3	3	2	2	1			1	2	1	3
		CO 5	3	3	3	2	2	1			1	2	1	3

SEMESTER – I									
Course Title	**Software Foundation and Programming (with C and C++)**								
Course code	**24BCSM1105R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+ 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. Learning this course will lead you to understand basic concepts of workshop and manufacturing 2. Apply fundamental knowledge of workshop and manufacturing in day-to-day life 3. Recognize components using different materials.								
CO1	Introduction to various manufacturing methods like casting, Forming, machining etc.								
CO2	Application of computer coding in automation of Machines.								
CO3	Introduction to Carpentry & fitting operations and its application in industries.								
CO4	Different machining operations like turning, milling.								
CO5	Learning the different types of welding and its field of application.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Introduction to programming concepts: algorithms, flowcharts, and pseudocode Overview of C and C++ programming languages Structure of a C/C++ program and basic syntax Input and Output operations	8	Understand programming fundamentals and solve simple computational problems using structured approaches.	1,2,3
II	Decision-making constructs (if-else, switch-case) Loops (for, while, do-while) Functions in C/C++: declaration, definition, and parameter passing Recursion basics	8	Apply control structures and functions to write modular and efficient programs.	3,4
III	Arrays: single and multidimensional Strings and basic string operations Introduction to pointers and pointer arithmetic Dynamic memory allocation	8	Manipulate data using arrays, strings, and pointers to solve intermediate-level problems.	3,4
IV	Object-oriented concepts: classes, objects, and encapsulation Constructors and destructors Inheritance and polymorphism Operator overloading	8	Demonstrate object-oriented programming techniques to create reusable and efficient software solutions.	4,5
V	File operations in C and C++ (text and binary files) Exception handling in C++ Overview of templates and STL (Standard Template Library) Debugging and testing techniques	8	Develop programs with file handling and leverage advanced C++ features for robust software development.	3,5

Practical Component				
Practical 1	Write a program to take user input (name and age) and display a welcome message.	3	Understand basic input/output operations in C.	1,2
Practical 2	Write a program to determine whether a number is even or odd using if-else.	3	Apply decision-making constructs to solve simple problems.	3
Practical 3	Write a program to print the multiplication table of a given number using a loop.	3	Use loops to implement repetitive tasks efficiently.	3
Practical 4	Write a program to calculate the factorial of a number using a user-defined function.	3	Understand modular programming using functions.	3
Practical 5	Write a program to find the largest element in an array of integers.	3	Manipulate and process data stored in arrays	4
Practical 6	Write a program to swap two numbers using pointers.	3	Learn basic pointer operations and their application.	3
Practical 7	Create a class Student with attributes name and marks. Write methods to input and display details.	3	Demonstrate object-oriented programming concepts like classes and objects.	4
Practical 8	Write a C++ program to implement a class with a constructor to initialize values and a destructor to display a message.	3	Understand the use of constructors and destructors in managing object lifecycle.	3
Practical 9	Write a C++ program to create a file, write data to it, and then read the data back.	3	Perform file operations such as reading from and writing to files.	4
Practical 10	Write a C++ program to create a function template for swapping two variables of different data types.	3	Implement generic programming using templates in C++.	5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the concept of computer system, analyze a given problem, develop an algorithm, fundamental programming constructs, identify data representation formats, and describe operators and their precedence, associativity	PO1,PO2,PO3,PO4,PO5,PO11
2	Apply branching and loop statements in problem solving.	PO1,PO2,PO3,PO4,PO5,PO11
3	Apply the concepts of homogeneous derived data types, heterogeneous data types, strings and functions in programming.	PO1,PO2,PO3,PO4,PO5,PO11
4	Understand the concept of pointers and apply it in programming.	PO1,PO2,PO3,PO4,PO5,PO11
5	Apply the concept of file handling in C programming.	PO1,PO2,PO3,PO4,PO5,PO11

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM1105R	**Software Foundation and Programming (with C and C++)**	CO1	3	3	2	1	2						1	
		CO2	3	3	2	1	2						1	
		CO3	3	3	3	1	2						1	
		CO4	3	3	2	1	1						1	
		CO5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**Workshop for Engineers**								
Course code	**24BCSM1104R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 15T+ 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	4. Learning this course will lead you to understand basic concepts of workshop and manufacturing 5. Apply fundamental knowledge of workshop and manufacturing in day-to-day life 6. Recognize components using different materials.								
CO1	Introduction to various manufacturing methods like casting, Forming, machining etc.								
CO2	Application of computer coding in automation of Machines.								
CO3	Introduction to Carpentry & fitting operations and its application in industries.								
CO4	Different machining operations like turning, milling.								
CO5	Learning the different types of welding and its field of application.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Manufacturing Methods:** Casting, forming, machining, joining, advanced manufacturing methods	3	Manufacturing methods include casting, forming, machining, and joining, which involve shaping and assembling materials into final products. Advanced manufacturing methods leverage modern technologies such as additive manufacturing, laser machining, and automation to enhance efficiency, precision, and product quality.	1,2,3
II	**CNC machining, Additive Manufacturing: o**verview of CNC machining process, overview of additive manufacturing.	3	CNC Machining: An automated manufacturing process that uses pre-Programmed computer software to control machinery and tools to shape, cut, and drill materials with high precision. Additive Manufacturing: A process of creating objects by adding material layer by layer, commonly known as 3D printing, allowing for complex geometries and reduced material waste.	3,4
III	**Carpentry & Fitting operations:** Carpentry tools, carpentry operations, fitting tools, fitting operations	3	Understand the use of various carpentry tools and perform fundamental carpentry operations. Gain proficiency in using fitting tools and executing essential fitting operations.	3,4
IV	**Machining operations:** Turning, milling, turning processes, milling processes	3	Understand the principles, equipment, and techniques involved in turning and milling operations. Gain proficiency in identifying and performing specific processes related to turning and milling, including tool selection, setup, and operation.	4
V	**Welding:** Arcwelding & gas welding, brazing	3	Understand the principles, techniques, and applications of arc welding and gas welding. Develop skills in brazing, including proper material selection and	4

			joint preparation for effective metal joining.	
Practical Component				
Practical 1	**A brief introduction of workshop:** Machine shop, Fitting shop, carpentry shop, welding shop	5	Gain practical skills and foundational knowledge in machine operations, fitting, carpentry, and welding techniques essential for manufacturing and construction industries.	1,2
Practical 2	**Machine shop:** Plain turning, Taper turning, Step turning	10	Understand and execute the basic turning operations on a lathe, including plain turning, taper turning, and step turning, ensuring precision and adherence to specified dimensions. Develop skills in setting up and operating a lathe, selecting appropriate tools and parameters for each operation, and maintaining safety protocols throughout the machining process.	2,3
Practical 3	**Carpentry shop:** Dovetail joint, T-lap joint, Cross-lap joint, Corner-lap joint	5	Gain practical skills in creating dovetail, T-lap, cross-lap, and corner-lap joints, essential for woodworking projects. Understand the structural benefits and appropriate applications of each joint type in carpentry.	3,4
Practical 4	**Welding shop:** Arcwelding (Butt joint,T-joint, Lap joint, Corner joint)	5	The welding shop training aims to develop skills in performing arc welding on various joint types, including butt joints, T-joints, lap joints, and corner joints. Trainees will learn proper techniques, safety practices, and quality control measures to produce strong, reliable welds in different configurations.	4
Practical 5	**Fitting shop:** Angle fitting, square fitting, corner fitting	5	Participants will gain proficiency in identifying and selecting appropriate angle, square, and corner fittings for various construction needs. Participants will demonstrate competence in assembling and integrating angle, square, and corner fittings into structural frameworks effectively.	5

TEXT BOOKS:

T1: Elements of Workshop Technology, Vol. I 2008 and Vol. II 2010,
T2: Manufacturing Engineering and Technology

REFERENCE BOOKS:

R1: Principles Manufacturing Technology – I, Pearson Education, 2008

R2: Processes and Materials of Manufacture, Roy A. Lindberg 4th edition, Prentice Hall India, 1998

OTHER LEARNING RESOURCES:

O1: Coursera workshop & Manufacturing Courses
O2: edX manufacturing technology Courses

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply electric field and potential calculations, Compute the vectors and scalar representation of forces and nature of forces.	PO1,PO2,PO3,PO5,PO7
2	Analyze electrostatics in dielectric media, conservative and non-conservative forces, angular momentum and energy equations.	PO1,PO5,PO11,PO12
3	Compute basics of non-inertial frames, harmonic oscillator and forced oscillations.	PO1,PO3,PO9
4	Demonstrate understanding of magnetostatics in linear magnetic media, and the usage of common electrical measuring instruments.	PO1,PO2,PO5,PO9
5	Understand the basic characteristics of transformers and electrical machines.	PO1,PO2,PO3,PO5,PO6

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM1104R	**Workshop for Engineers**	**CO 1**	3	3	2	1	2						1	
		CO 2	3	3	2	1	2						1	
		CO 3	3	3	3	1	2						1	
		CO 4	3	3	2	1	1						1	
		CO 5	3	3	3	1	2						1	

SEMESTER – I									
Course Title	**MOOCS I: Introduction to Java**								
Course code	**24MOCS1101R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 40	**0**	**0**	**0**	**8**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the first year of the program**								

Course Objectives (Minimum 3)	1. To introduce students to the fundamental concepts of object-oriented programming using Java. 2. To familiarize learners with the syntax and basic constructs of the Java programming language. 3. To enable students to understand and apply key programming concepts such as variables, data types, operators, and control flow statements. 4. To provide hands-on experience in writing, compiling, and executing Java programs. 5. To develop problem-solving skills by implementing basic programming logic using Java.			
CO1	Understand the basic structure and syntax of Java programming language.			
CO2	Write simple Java programs to perform basic input and output operations.			
CO3	Apply fundamental programming concepts such as variables, data types, and control structures in Java.			
CO4	Develop small programs using loops and conditional statements in Java.			
CO5	Create and run basic Java programs using an Integrated Development Environment (IDE).			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Java :** Introduction to Java and its features, Overview of the Java Virtual Machine (JVM), Java Development Kit (JDK) and Integrated Development Environment (IDE) setup, Writing, compiling, and running your first Java program	**8**	Understand the evolution, features, and basic structure of Java programs. Set up the Java environment and execute simple programs.	**1,2,3**
II	**Basics of Java Programming:** Java syntax and data types, Variables, constants, and type casting, Operators (arithmetic, relational, logical, assignment, etc.), Input and output operations in Java	**8**	Describe the fundamental syntax and elements of Java programming. Apply basic operations using operators and variables in Java programs.	**2,3**
III	**Control Flow Statements:** Decision-making statements (if, if-else, switch), Looping constructs (for, while, do-while loops), Nested loops and the use of break/continue statements	**8**	Explain and use decision-making and looping constructs in Java programs. Write programs to solve problems using control flow statements.	**3,4**
IV	**Object-Oriented Programming in Java:** Introduction to Object-Oriented Programming (OOP) principles, Classes and objects in Java, Constructors and method overloading, Access modifiers and the this keyword	**8**	Describe the principles of OOP and their implementation in Java. Develop programs using classes, objects, and constructors.	**4**

V	**Arrays and Strings in Java :** Co Introduction to arrays (single-dimensional and multi-dimensional arrays), Basic operations on arrays (initialization, traversal, and manipulation), Strings in Java and common string operations, Introduction to the StringBuffer and StringBuilder classes	**8**	Demonstrate the use of arrays and strings in Java programs. Write programs to perform operations on arrays and strings.	**5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basic structure and syntax of Java programming language.	PO1,PO2,PO3,PO4,PO5,PO12
2	Write simple Java programs to perform basic input and output operations.	PO1,PO2,PO3,PO4,PO5,PO12
3	Apply fundamental programming concepts such as variables, data types, and control structures in Java.	PO1,PO2,PO3,PO4,PO5,PO12
4	Develop small programs using loops and conditional statements in Java.	PO1,PO2,PO3,PO4,PO5,PO12
5	Create and run basic Java programs using an Integrated Development Environment (IDE).	PO1,PO2,PO3,PO4,PO5,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24MOCS101R	**Introduction to Java**	**CO1**	2	2	3	1	1							1
		CO2	2	1	3	1	1							1

		CO3	2	1	2	1	1							1
		CO4	2	1	2	1	1							1
		CO5	2	2	2	1	1							1

SEMESTER – I									
Course Title	**PDP (Introductory English)**								
Course code	**24UBPD1104R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the first year of the Programme**								
Course Objectives (Minimum 3)	1. To capacitate the students with mastery over Basic English grammar. 2. To enable the students to communicate confidently with a focus on listening a speaking skills. 3. With the help of the basics of Phonetics, the students will be able to pronounce wor correctly. 4. To interact successfully and with decorum.								
CO1	Analyze and apply the rules of Parts of Speech in constructing grammatically correct sentences.								
CO2	Evaluate sentence structures, employ varied sentence types, and demonstrate effective comprehension skills								
CO3	Assess listening skills, identify factors influencing listening, and implement strategies for improved listening.								
CO4	Demonstrate effective speaking skills, including self-introduction, pronunciation, and extempore speech delivery.								
CO5	Apply communication theories, recognize barriers, and enhance skills for diverse communication situations.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar** 1. Parts of Speech 2. Articles 3. Auxiliary Verbs 4. Affirmative and Negative Sentences	**5**	You'll learn how different parts of speech function in sentences, including articles and auxiliary verbs, enabling you to construct both affirmative and negative sentences effectively.	**1**
II	**Grammar** 1. Determiners 2. Sentence Construction 3. Types of Sentences (Assertive, Imperative, etc.) 4. Degree of Comparison 5. Comprehension Exercises	**7**	Master determiners, sentence construction, and types of sentences (assertive, imperative, etc.), along with understanding degrees of comparison. Practice comprehension exercises to reinforce understanding and application of these concepts effectively.	**3**
III	**Listening Skills** 1. What is listening? 2. The Process of Listening 3. Factors that adversely affect Listening 4. Difference between	**6**	By studying listening skills, you will understand the distinction between listening and hearing, recognize factors that hinder effective listening, appreciate the importance of active listening, and	**5**

	Listening and Hearing, 5. Purpose and Importance of Effective Listening 6. How to Improve Listening Process.		learn strategies to enhance the listening process	
IV	**Speaking Skills** 1. Introducing yourself 2. Self-discovery 3. Basics of Phonetics, pronunciation 4. Extempore speech 5. Video Recording for Self reflection	**6**	Developing speaking skills through self-introduction, phonetics, and extempore speaking, enhanced by video reflections for self-improvement and self-discovery.	**4**
V	**Communication Skills** 1. Introduction to Communication, 2. Importance of Communication Skills, 3. Purpose of Communication, 4. Types of Communication, 5. Formal and informal communication 6. Importance of Communication, 7. Barriers to Communication, 8. How to improve/ tips to improve Communication skills. 9. Responding to different questions in various situations(formal/informal)	**6**	By studying communication skills, learners gain a comprehensive understanding of its importance, types, barriers, and strategies for improvement, enabling effective interaction in various contexts.	**3**

TEXT BOOKS:

T1: Chaturvedi, P.D., ChaturvediMukesh, 2011.Business Communication: Concepts, Cases and Applications, second edition, Pearson,Noida.

T2: Alex K., Chand, S, 2009. Soft Skills: Know Yourself and Know the World, first edition, S. Chand & Company Ltd.: NewDelhi

REFERENCE BOOKS:

R1: Quirk, Randolp. (2010) A Comprehensive Grammar of the English Language by Randolph Quirk, Sidney Greenbaum, Pearson EducationIndia

R2: Marks, Jonathan. (2017) IELTS Advantage Speaking and Listening Skills: A step-by-step guide to a high IELTS speaking and listening score. Book + CD-ROM, Delta Publishing by Klett

OTHER LEARNING RESOURCES:

O1: Clapingo english Courses
O2: English partner India's Largest Online English Learning Platform

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Analyze and apply the rules of Parts of Speech in constructing grammatically correct sentences.	PO10
2	Evaluate sentence structures, employ varied sentence types, and demonstrate effective comprehension skills	PO2,PO10
3	Assess listening skills, identify factors influencing listening, and implement strategies for improved listening.	PO9,PO10
4	Demonstrate effective speaking skills, including self-introduction, pronunciation, and extempore speech delivery.	PO10
5	Apply communication theories, recognize barriers, and enhance skills for diverse communication situations.	PO6,PO10

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24UBPD1104R	Introductory English for ngineers	CO 1										3		
		CO 2		2								3		
		CO 3									2	3		
		CO 4										3		
		CO 5						2				3		

SEMESTER – I									
Course Title	**Co-curricular Activities**								
Course code	**24UBCC1101R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30S	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the first year of the program**								
Course Objectives (Minimum 3)	1. It is to develop the social and soft skills and to promote a holistic development of the learners. 2. Participating in debate clubs, drama, or public speaking events enhances verbal and non-verbal communication skills. 3. Taking leadership roles in student organizations or event planning committees fosters leadership and teamwork abilities.								
CO1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.								
CO2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.								
CO3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.								
CO4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.								
CO5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Discover the exciting world of engineering through hands-on projects and workshops. Gain practical skills in robotics, coding, and innovation, preparing for future STEM success.	30	Explore the dynamic field of engineering through interactive projects and workshops. Develop practical skills in robotics, coding, and innovation. Prepare for future success in STEM disciplines.	1,2,3,4,5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Connect and adapt cultural diversity among communities.	PO1, PO6, PO8, PO10, PO12
2	Enhance team for working toward a shared vision	PO6, PO9, PO10, PO12
3	Demonstrate and apply interdisciplinary connections and Cultivate spirit of creative thought and curiosity to achieve goals	PO2, PO4, PO10, PO12
4	Learn to effectively communicate, delegate responsibilities and motivate team members.	PO2, PO6, PO10, PO12
5	Develop strong teamwork and collaboration skills by engaging in group activities.	PO1, PO2, PO6, PO8, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24UBCC1101R	**Co-Curricular Activities**	**CO1**	2					3		1		2		1
		CO2						1			3	3		1
		CO3		3		3						2		2
		CO4		3				3				2		2
		CO5	2	3				3		1				2

SEMESTER – II									
Course Title	**Object Oriented Programming using JAVA**								
Course code	**24BCSM1201R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1.To enable students to understand the basic object oriented programming concepts and apply them in problem solving. 2.Additionally, to illustrate among students the inheritance concepts for reusing the program and also to enable students. 3.To develop GUI-based applications using AWT, Swing and Event handling.								
CO1	Understand object-oriented programming concepts and execute them proficiently in Java.								
CO2	Apply building blocks of OOPs language, including inheritance, packages, and interfaces, analysing real-world problems in their context.								
CO3	Utilize multithreading, exception handling, and other OOPs concepts effectively in object-oriented programs.								
CO4	Apply various exception handling methods in programming, demonstrating competence in error management.								
CO5	Develop interactive and GUI-based Java applications through project-based learning, showcasing practical application of Java programming skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Java Overview:** Java Introduction, Platform Independence, JVM & JDK, Data types, Operators, If, else statement, Switch condition, while, do-while, for loop, break and continue statement. **Array and String:** Single Array & Multidimensional Array, Library Classes-String, String Buffer & Wrapper Class, Command line arguments and Various String Operations.	**5**	Describe the concept of platform independence in Java and how the Java Virtual Machine (JVM) enables it. Differentiate between the Java Virtual Machine (JVM) and the Java Development Kit (JDK), and understand their roles in Java application development. Define and declare single-dimensional arrays in Java, understanding array initialization, accessing elements, and iterating through array elements.	**1,2,3**
II	**Classes, Objects and Methods:** Class and Object, Object reference, Constructor: Constructor Overloading, Method: Method Overloading, Recursion,	**7**	Define classes and objects in Java, understanding the concept of encapsulation and how classes serve as blueprints for creating objects. Declare and define classes in Java, including instance variables, constructors, methods, and access modifiers (public, private, protected). Create objects	**3,4,5,6**

	Passing and Returning object form Method, new operator, this and static keyword, finalize () method, Import statement, Static import, Access control, Nested class, Inner class, Anonymous inner class.		of a class using constructors and understand the process of object initialization, including default constructors and parameterized constructors.	
III	**Inheritance and Interfaces in Java:** Overview of Inheritance, inheritance in constructor, Inheriting Data members and Methods, Multilevel Inheritance – method overriding Handle multilevel constructors Explain super keyword, Stop Inheritance, Explain Final keywords, Creation and Implementation of an interface, Interface reference, instance of operator, Interface inheritance, Dynamic method dispatch, Abstract class, Comparison between Abstract Class and interface, inside of System.out.println – statements.	**6**	Differentiate between types of inheritance in Java, including single inheritance, multiple inheritance through interfaces, and hierarchical inheritance. Implement method overriding in subclasses to provide specialized implementations of methods inherited from superclasses, understanding runtime polymorphism.	**2,3,4,5**
IV	**Exception Handling in Java:** Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. JAVA File Handling: Overview of Different Stream (Byte Stream, Character stream), Readers and Writers class, FileClass, File Input Stream, File Output Stream, Input Stream Reader and Output Stream Writer class, File reader and writer class, File Writer, Buffered Reader class.	**6**	Define exceptions in Java and understand their role in handling runtime errors and abnormal conditions that may occur during program execution. Implement try-catch blocks to handle exceptions gracefully, preventing program termination and providing alternative behaviors or error messages.	**1,3,4,5**
V	**Applet, AWT and Swing:** MVC Architecture, Applet: Applet Fundamental,	**6**	Define what Java applets are and understand their role in web-based applications, particularly in the context of early web development.	**4**

	Applet Architecture, Applet Skeleton, Requesting Repainting, Event Handling: various event handling mechanisms, Delegation Event Model, Events, Event Sources, Event Listeners, various classes related to event sources and event listeners, AWT: window fundamentals, creating frames, Adding removing various controls, Layout managers, Introduction To Swing, Applications and Pluggable look and feel, Basic swing components: Text Fields, Buttons, Toggle Buttons, Checkboxes, and Radio Buttons.		Describe the lifecycle of an applet, including initialization (init()), start (start()), stop (stop()), and destruction (destroy()), understanding how applets interact with web browsers. Implement event handling in Java applets to respond to user interactions such as mouse clicks, keyboard input, and window events, using event listeners and adapters.	
Practical Component				
Practical 1	Hands-on Introduction to Object-Oriented Programming: Practical exercises on creating classes and objects, Implementing encapsulation through practical examples, Coding exercises on constructors and destructors, Applying access modifiers (public, private, protected) in practical scenarios.	**6**	Define object-oriented programming (OOP) concepts such as classes, objects, inheritance, polymorphism, and encapsulation, understanding their role in software development. Create classes in Java, defining attributes (fields) and behaviors (methods) that represent real-world entities or concepts, and instantiate objects from these classes.	**1,2,4,5**
Practical 2	Practical Inheritance and Polymorphism: Coding exercises on implementing different types of inheritance (single, multi-level, and multiple), Practical examples of method overriding, Creating and using abstract classes and interfaces, Implementing static and dynamic binding through practical examples.	**8**	Implement different types of inheritance in Java: single inheritance (one subclass inherits from one superclass), multi-level inheritance (a subclass extends another subclass), and multiple inheritance (a subclass inherits from multiple superclasses using interfaces).	**2,4,5**
Practical 3	Advanced OOP Concepts in Practice: Practical exercises on creating and using packages and namespaces, Implementing exception	**8**	Organize Java classes into packages to manage code structure and namespace, demonstrating how packages encapsulate related classes and provide access control.	**1,2,3**

	handling in real-world scenarios, Creating and managing multithreaded applications, Practical exercises on generics and collections.		Develop practical exercises that involve creating, importing, and utilizing packages in Java applications, ensuring modularization, and facilitating code reuse and maintenance.	
Practical 4	Software Design and Development in Action: Hands-on Object-Oriented Analysis and Design (OOAD) exercises, Implementing design patterns (e.g., Singleton, Factory, Observer) in practical scenarios, Software testing and debugging exercises, Code optimization and refactoring practical sessions.	**8**	Conduct hands-on exercises on OOAD principles, including identifying objects, defining classes, specifying relationships (inheritance, composition), and modeling behavior using UML diagrams (class diagrams, sequence diagrams, etc.). Implement commonly used design patterns (e.g., Singleton, Factory, Observer, Strategy) in practical scenarios, demonstrating their application to solve recurring design problems and improve code flexibility and maintainability.	**2,3,4,5**

TEXT BOOKS:

T1: Herbert Schildt and Dale Skrien, ‖Java Fundamentals – A comprehensive Introduction‖, McGraw Hill.

T2: Herbert Schildt, ―Java the complete reference, McGraw Hill, Osborne.

T3: T. Budd, ―Understanding Object-Oriented Programming with Java‖, Pearson Education.

REFERENCE BOOKS:

R1: P.J. Dietel and H. M. Dietel,―Java How to program ‖,Prentice Hall.

R2: P. Radha Krishna, ―Object Oriented programming through Java‖, CRC Press.

OTHER LEARNING RESOURCES:

O1: http://java.sun.com

O2: http://www.oracle.com/technetwork/java/index.html)

O3: http://java.sun.com/javase

O4: http://www.oracle.com/technetwork/java/javase/overview/index.html
O5: http://download.oracle.com/javase/7/docs/api/index.htmlE-Text Books:
O6: http://docs.oracle.com/javase/tutorial/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand object-oriented programming concepts and execute them proficiently in Java.	PO1,PO2,PO3,PO4,PO5,PO12

2	Apply building blocks of OOPs language, including inheritance, packages, and interfaces, analysing real-world problems in their context.	PO1,PO2,PO3,PO4,PO5,PO12
3	Utilize multithreading, exception handling, and other OOPs concepts effectively in object-oriented programs.	PO1,PO2,PO3,PO4,PO5,PO12
4	Apply various exception handling methods in programming, demonstrating competence in error management.	PO1,PO2,PO3,PO4,PO5,PO12
5	Develop interactive and GUI-based Java applications through project-based learning, showcasing practical application of Java programming skills.	PO1,PO2,PO3,PO4,PO5,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM1201R	**Object Oriented Programming using JAVA**	**CO1**	2	2	3	1	1							1
		CO2	2	1	3	1	1							1
		CO3	2	1	2	1	1							1
		CO4	2	1	2	1	1							1
		CO5	2	2	2	1	1							1

SEMESTER – II									
Course Title	**Data Structure and Algorithms**								
Course code	**24BCSM1202R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Fundamentals of programming logic	**Co-requisite**	Basics of C Programming						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. To impart the basic concepts of data structures and algorithms. 2. To understand basic concepts about stacks, queues, lists, trees and graphs. 3. To understand concepts about searching and sorting techniques								
CO1	Understand analysis of algorithms using asymptotic notations, and learn search technique								
CO2	Analyse algorithms on stacks and queues and their applications.								
CO3	Implement and analyse operations on linked lists and its variations and their applications.								
CO4	Apply tree terminologies and operations on different types of trees, with a focus on algorithmic analysis and practical applications.								
CO5	Evaluate and compare various sorting algorithms, hashing techniques, and graph theoretic concepts along with their complexity analysis.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction:** Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.	**8**	Understanding fundamental data structures, algorithm analysis, and search techniques, including complexities, for efficient problem-solving and resource management in computing.	1,2,3
II	**Stacks and Queues**: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.	**10**	Learning outcomes: Mastery of stack and queue ADTs, including operations and complexities, and their practical applications in expression conversion, evaluation, and various queue types with corresponding algorithms and analyses.	3,4
III	**Linked Lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.	**10**	Gain expertise in linked list structures: singly linked lists, doubly linked lists, circular linked lists, stack and queue implementations, and algorithmic complexities.	3,4
IV	**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree,	**6**	Learning outcomes: Proficiency in basic tree	3,5

	Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.		terminology, operations on binary, threaded, AVL trees, and B/B+ trees, including algorithms and complexity analysis, with applications in various domains.	
V	**Sorting and Hashing:** Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing. **Graph:** Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.	**6**	Master sorting (Selection, Bubble, Insertion, Quick, Merge, Heap), hashing principles, and graph operations (search, traversal) with complexity analyses.	3,4,5
Practical Component				
Practical 1	Array Operations- Program to perform basic array operations (insertion, deletion, search, update).	3	Learning outcome: Proficiency in performing array operations (insertion, deletion, search, update) essential for efficient data manipulation and algorithm implementation in programming.	2,3
Practical 2	Linked List Operations- Program to implement singly linked list operations (insertion, deletion, traversal). Program to implement doubly linked list operations (insertion, deletion, traversal). Circular Linked List- Program to implement circular linked list operations (insertion, deletion, traversal).	3	Mastery of implementing and manipulating singly, doubly, and circular linked lists through operations like insertion, deletion, and traversal in programming contexts.	2,3,4
Practical 3	Stacks and Queues- Stack Using Array- Program to implement stack operations using arrays (push, pop, peek). Stack Using Linked List- Program to implement stack operations using linked lists. Infix to Postfix Conversion Program to convert infix expression to postfix expression using stack.	3	Proficiency in implementing stack operations using arrays and linked lists, and converting infix expressions to postfix using stacks, enhancing algorithmic understanding and programming skills.	3,4
Practical 4	Queue Using Array- Program to implement queue operations using arrays (enqueue, dequeue, front, rear). Queue Using Linked List- Program to implement queue operations using linked lists. Circular Queue-	3	Mastery in implementing queue operations using arrays and linked lists, including circular queue implementation, crucial for efficient data management and algorithmic	4,5

	Program to implement circular queue using arrays.		proficiency in programming.	
Practical 5	Trees Binary Tree Creation Program to create a binary tree and perform pre-order, in-order, and post-order traversals. Binary Search Tree (BST) Operations a) Program to implement BST operations (insertion, deletion, search).	3	Ability to create and traverse binary trees using preorder, in-order, and post-order methods, and proficiency in implementing BST operations (insertion, deletion, search) for efficient data organization and manipulation in programming.	2,3 ,4
Practical 6	AVL Tree Implementation Program to implement AVL tree operations (insertion with rotations). Heap Implementation a) Program to implement a max-heap or min-heap and perform heap operations (insert, delete, heap if y).	3	Mastering AVL tree operations with rotations for balance and implementing heap structures with insertions, deletions, and maintenance in programming contexts.	2,3 ,4
Practical 7	Sorting Programs- Merge Sort Implementation: Write a C program to implement the merge sort algorithm to sort an array of integers. Quick Sort Implementation: Implement the quick sort algorithm in C to sort an array of integers. Heap Sort Implementation: Write a C program to implement the heap sort algorithm to sort an array of integers.	3	Proficiency in implementing and understanding merge sort, quick sort, and heap sort algorithms for efficient array sorting in programming.	3,5
Practical 8	Hash Table with Chaining: Implement a hash table using chaining for collision resolution in C. Include functions to insert, search, and display elements.	3	Learning to implement a hash table with chaining for collision resolution, including functions for insertion, searching, and displaying elements, crucial for efficient data management in C programming.	3,5
Practical 9	Hash Table with Linear Probing: Write a C program to implement a hash table using open addressing with linear probing for collision resolution. Include functions to insert, search, and display elements.	3	Proficiency in implementing a hash table using linear probing for collision resolution, including functions for insertion, searching, and displaying elements in C programming, essential for efficient data management and retrieval.	3,4
Practical 10	Hash Table with Quadratic Probing: Implement a hash table using open addressing with quadratic probing for collision resolution in C. Include	3	Learning to implement hash tables with quadratic probing for collision resolution, including	4,5

	functions to insert, search, and display elements		insert, search, and display functions in C.	

TEXT BOOKS:

T1: "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.

T2: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein**,** "Introduction to Algorithms" by Cormen et al.

T3: Seymour Lipschutz , "Data Structures, with C" by McGraw-Hill Education, Schaum's Outlines series.

REFERENCE BOOKS:

R1: Algorithms, "Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company

R2: "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education.

OTHER LEARNING RESOURCES:

O1: Coursera: "Data Structures and Algorithms" by UC San Diego and National Research University Higher School of Economics

O2: Udemy: "Master the Coding Interview: Data Structures + Algorithms" by Andrei Neagoie

O3: GeeksforGeeks: Offers tutorials, explanations, and problem-solving sessions on data structures and algorithms.

O4: Tutorialspoint: Provides comprehensive tutorials on data structures and algorithms.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand analysis of algorithms using asymptotic notations, and learn search technique	PO1, PO2, PO3, PO10, PO12
2	Analyse algorithms on stacks and queues and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Implement and analyse operations on linked lists and its variations and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Apply tree terminologies and operations on different types of trees, with a focus on algorithmic analysis and practical applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Evaluate and compare various sorting algorithms, hashing techniques, and graph theoretic concepts along with their complexity analysis.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM1202R	**Data Structure & Algorithms**	**CO1**	2	2	1							1		1
		CO2	3	3	2	2	1					1		2
		CO3	3	3	3	1	1					1		2
		CO4	3	3	2	2	1					1		2
		CO5	3	3	2	2	1					1		2

SEMESTER – II									
Course Title	**Environmental Science**								
Course code	**24UBES1201R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**

		Total hours: 36T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the first year of the program**								
Course Objectives (Minimum 3)	1. Understanding Ecological Systems and Processes. 2. Analysing Environmental Issues and Solutions. 3. Promoting Environmental Stewardship and Sustainable Practices.								
CO1	Understand the relationships between natural and man-made systems.								
CO2	Develop critical thinking for shaping strategies (scientific, social, economic, administrative, and legal) for environmental protection, conservation of biodiversity, environmental equity, and sustainable development.								
CO3	Understand the consequences of human actions on the web of life, global economy, and quality of human life.								
CO4	Understand about Environment Pollution, Global Environment issues, Impact of Human Population and modern civilization on the Environment								
CO5	Analyse various aspects of human population, and the impact of the population growth on the environment.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Environment:** Definition and scope, components of environment, atmosphere, hydrosphere, lithosphere and biosphere, structure and composition, Life systems, pro and eukaryotic organizations, Metabolic principles; types of plants and animals. Producers, consumers and decomposers.	**7**	Identify and describe the major components of the environment: atmosphere (air), hydrosphere (water), lithosphere (earth's crust), and biosphere (living organisms), understanding their interactions and interdependencies. Define "environment" in the context of ecological and environmental sciences, exploring its scope as encompassing natural, built, and social environments.	**1,2,3**
II	**Ecology:** Terminology and approach, ecosystem, types of ecosystems; structure and function, mineral cycling, energy flow and trophic chains. Development and evolution.	**7**	Define key ecological terms such as ecosystem, community, population, niche, habitat, and biodiversity, understanding their roles in studying interactions between organisms and their environment. Explain the concept of an ecosystem as a functional unit consisting of biotic (living organisms) and abiotic (non-living environment) components interacting within a defined area.	**3,4**
III	**Environmental Pollution:** Sources, causes, assessment, effect, prevention and control of water pollution, air pollution noise and land pollution.	**7**	Identify and categorize sources of pollution such as industrial emissions, agricultural runoff, urban waste, and vehicular	**3,4**

	Strategies of management, concept of sustainability. Energy, e n v i r o n m e n t , and their relationship with human activities. Water Resources and utilization, forest resources.		emissions, understanding their contributions to environmental degradation. Conduct assessments of environmental pollution using scientific methods and tools, including monitoring techniques, data analysis, and environmental impact assessments (EIAs).	
IV	**Global Environmental Problems:** Human health, settlements, management of rivers, lakes, forests, wild life and catchments. Role of society, NGO and Govt. agencies. Concept of urbanization and green cities Global Warming, greenhouse causes and effects, carbon Sequestration.	**7**	Analyze the impact of environmental pollution and degradation on human health, including respiratory diseases, waterborne illnesses, and exposure to hazardous substances. Discuss the challenges and opportunities of urbanization, including urban sprawl, infrastructure development, resource consumption patterns, and sustainable urban planning practices.	**3,5**
V	**International agreements and protocols**, National forest policy and Environmental laws and acts. EIA.	**7**	Explain the purpose and significance of international environmental agreements and protocols, such as the Kyoto Protocol, Paris Agreement, Convention on Biological Diversity (CBD), and Montreal Protocol, analyzing their goals, implementation mechanisms, and global impact.	**3,5**

TEXT BOOKS:

T1: Sudhir Andrews (2013) Food and Beverage Service: A Training Manual, Tata McGraw Hill, 2013.

REFERENCE BOOKS:

R1: PradipDey, ManasGhosh, "Programming in C", 2nd Edition, 2018, Oxford University

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**

1	Understand the relationships between natural and man-made systems.	PO1,PO2,PO3,PO4,PO7,PO12
2	Develop critical thinking for shaping strategies (scientific, social, economic, administrative, and legal) for environmental protection, conservation of biodiversity, environmental equity, and sustainable development.	PO1,PO2,PO3,PO4,PO7,PO12
3	Understand the consequences of human actions on the web of life, global economy, and quality of human life.	PO2,PO7,PO12
4	Understand about Environment Pollution, Global Environment issues, Impact of Human Population and modern civilization on the Environment	PO2,PO3,PO7
5	Analyse various aspects of human population, and the impact of the population growth on the environment.	PO1,PO2,PO3,PO4,PO7,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24UBES1201R	**Environmental Science**	**CO1**	1	2	2	2			3					3
		CO2	2	2	2	2			3					3
		CO3		3					3					3
		CO4		1	2				3					
		CO5	1	2	2	2			3					3

SEMESTER – II									
Course Title	**Programming with Python**								
Course code	**24BCSM1204R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. To introduce the core concepts of functional programming and their implementation in Python.								

	2. To develop proficiency in using Python's functional constructs such as lambda functions, map, filter, and reduce. 3. To enable students to write modular, reusable, and efficient code using functional programming paradigms. 4. To explore the use of recursion, higher-order functions, and immutability in functional programming. 5. To apply functional programming techniques to solve real-world computational problems effectively.			
CO1	Understand about the history of Programming Languages, types of programming languages, Basic Syntax of a few programming languages.			
CO2	Understand the fundamentals of Python programming language and its syntax, enabling them to write basic to intermediate level programs.			
CO3	Demonstrate proficiency in using data types and conditional statements to make simple Python programs and manipulate data effectively.			
CO4	Understand about Loops, functions and various packages in Python.			
CO5	Understand Lambda and user defined functions in Python.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**History of Programming Languages** Overview of Programming Languages, Basic Syntax of a few programming languages, Advantages and disadvantages of Python, Real-life examples of application of Python	**6**	Understanding Python's versatility through its syntax, advantages in ease of use, and real-world applications in diverse fields like data science and automation.	**1,2**
II	**Introduction to Python Programming** Introduction to Python Language, Writing and Compiling Python Programs, Basic Structure of a Python Program, Data Types and Constants	**10**	Developing proficiency in Python basics: syntax, program structure, data types, constants, enabling effective programming and problem-solving skills.	**2,3**
I	**Data types and Variables** Data types, Variable declaration and use, Typecasting in Python, Operators in Python: Assignment, Logical, Arithmetic, Conditional statements: If statement, If else statement, Python Collections: List, Tuple, Sets and Dictionary	**10**	Mastering Python essentials: data types, variables, typecasting, operators, conditional statements, and for robust programming proficiency.	**1,2,3**
IV	**Loops and Functions** Loops in Python: For Loop, While Loop and Nested Loops, User Defined functions	**7**	Proficiency in Python loop and user-defined functions, enabling efficient problem-solving and scalable software development.	**1,2,3**
V	**Lambda Functions** Types of functions, Lambda functions	**7**	Understanding types of functions (built-in, user-defined) and lambda functions in Python for versatile	**1,2,3**

			programming and concise code implementation.	
Practical Component				
Practical 1	**Introduction to Python** Overview of Python Programming ● Setting up Python environment a) Installing Python b) Installing an IDE (e.g., PyCharm, VS Code, Jupyter Notebook) ● Writing your first Python program ● Basic syntax, comments, and documentation	**3**	Develop foundational skills in Python: setup environment, write code, apply basic syntax, comments, and documentation for effective programming.	**1,2,3**
Practical 2	**Basic Data Types and Variables** ● Variables and data types o Integers, floats, strings, and Booleans ● Type casting ● Basic input and output	**3**	Mastering Python basics: variables (integers, floats, strings, booleans), type casting, and basic input/output operations for practical programming proficiency.	**1,2,3**
Practical 3	**Operators and Expressions** ● Arithmetic operators ● Comparison operators ● Logical operators ● Assignment operators ● Bitwise operators	**3**	Understanding Python operators: arithmetic, comparison, logical, assignment, and bitwise, essential for comprehensive programming and problem-solving in diverse applications.	**3,4**
Practical 4	**Control Structures** ● Conditional statements a) if, else if, else ● Loops a) for loops b) while loops ● Break and continue statements	**3**	Mastering control structures in Python: conditional statements, loops, break, and continue statements for efficient program flow and logic implementation.	**1,2,3**
Practical 5	**Functions** ● Defining functions ● Calling functions ● Parameters and arguments ● Return values	**3**	Proficiency in Python functions: defining, calling, handling parameters/arguments, return values, variable scope, and utilizing	**1,2,3**

	• Scope of variables • Lambda functions		lambda functions for concise and efficient code implementation.	
Practical 6	**Data Structures** • Lists a) Creating and accessing elements b) List methods c) List comprehensions • Tuples • Dictionaries a) Creating and accessing elements b) Dictionary methods Sets	**3**	Proficiency in Python data structures: lists, tuples, dictionaries, and sets.	**1,2,3**
Practical 7	**Strings** • String operations • String methods • String formatting • Regular expressions	**3**	Mastering string operations, methods, formatting, and regular expressions in Python for robust text processing and pattern matching capabilities.	**1,2,3**
Practical 8	**File Handling** • Opening and closing files • Reading and writing files • Working with file modes • Using with statement	**3**	Proficiency in file handling in Python: opening, closing, reading, writing files, understanding file modes, and utilizing the with statement for resource management.	**1,2,3**
Practical 9	**Exception Handling** • Understanding exceptions • try, except, finally blocks • Handling multiple exceptions • Custom exceptions	**3**	Mastering exception handling in Python: understanding exceptions, using try-except-finally blocks, managing multiple exceptions, and creating custom exceptions for robust error management in programs.	**1,2,3**
Practical 10	**Modules and Packages** • Importing modules • Creating modules • Using standard libraries	**3**	Proficiency in module management in Python: importing modules, creating custom modules, and utilizing standard libraries like os, sys, math, datetime for enhanced functionality in applications.	**1,2,3**

TEXT BOOKS:

T1: BALAGURUSAMY, 1st Edition, Introduction To Computing And Problem-Solving Using Python, McGraw Hill, ISBN 9789352602582

REFERENCE BOOKS:

R1: Andrew Ngo, Introduction to Python Programming: Beginner to Advanced, Practical Guide, Tips and Tricks, Easy and Comprehensive, 1st Edition, 2017, Kindle Books

R2: Venugopal and Prasad, Python: The Complete Reference, 4th Edition, 2018, Tata McGraw Hill.

OTHER LEARNING RESOURCES:

O1: youtube: The youtube tutorial "Python Tutorial for Beginners" offers python programming tutorials for beginners, including functional programming principles.
O2: geeksforgeeks: Offers tutorials, including functional programming principles.
O3: Coursera: the course "Programming for Everybody (Getting Started with Python)" offered helps Gain a foundational understanding and learn new concepts from industry experts

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand about the history of Programming Languages, types of programming languages, Basic Syntax of a few programming languages.	PO1, PO2, PO3, PO5, PO11
2	Understand the fundamentals of Python programming language and its syntax, enabling them to write basic to intermediate level programs.	PO1, PO2, PO3, PO5, PO11
3	Demonstrate proficiency in using data types and conditional statements to make simple Python programs and manipulate data effectively.	PO1, PO2, PO3, PO5, PO11
4	Understand about Loops, functions and various packages in Python.	PO1, PO2, PO3, PO5, PO11
5	Understand Lambda and user defined functions in Python.	PO1, PO2, PO3, PO5, PO11

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BCSM1204R	**Functional Programming in Python**	**CO1**	3	1	1		2						1	
		CO2	1	2	2		2						1	
		CO3	1	1	1		2						1	
		CO4	1	2	1		2						1	
		CO5	2	1	1		2						1	

SEMESTER – II									
Course Title	**Field Based Learning**								
Course code	**24BCSM1203R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 16	**0**	**0**	**0**	**0**	**0**	**16**	**1**

Pre-requisite	Nil	Co-requisite	Nil	
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**			
Semester	**Winter/II Semester of First Year of the Programme**			
Course Objectives (Minimum 3)	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings. 4. To foster collaboration and communication skills through group projects and professional interactions. 5. To cultivate an appreciation for the complexities and challenges of professional practice in the field.			
CO1	Apply theoretical concepts to real-world situations.			
CO2	Conduct field-based research and gather data effectively.			
CO3	Analyze and interpret the university data to draw meaningful conclusions.			
CO4	Communicate findings clearly and effectively, both orally and in writing.			
CO5	Demonstrate professional behaviour and teamwork skills in field settings.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Field-Based Learning is designed to provide students with hands-on, practical experience in real-world environments related to Networking. This course emphasizes experiential learning through site visits, fieldwork, projects, and interactions with professionals in the field. Students will apply theoretical knowledge to practice, develop critical thinking skills, and gain a deeper understanding of their discipline.	**16**	Field-Based Learning provides students with hands-on experience in real-world networking environments. It emphasizes experiential learning through site visits, fieldwork, projects, and professional interactions. Students will apply theoretical knowledge, develop critical thinking skills, and gain a deeper understanding of networking.	**1,2,3,4,5**

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.	PO1, PO5, PO7, PO10, PO12
2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.	PO2, PO4, PO5, PO6, PO12
3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.	PO2, PO3, PO4, PO9, PO11
4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.	PO6, PO8, PO9, PO10, PO11
5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.	PO6, PO7, PO8, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM1203R	**Field-Based Training**	CO 1	3				2		1			1		1
		CO 2		3		3	2	1						2
		CO 3		3	3	2					2		1	
		CO 4						1		1	3	3	2	
		CO 5						3	3	2			1	2

SEMESTER – II									
Course Title	**PDP (Effective English for Engineers)**								
Course code	**24UBPD1204R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 40T	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**NIL**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/II Semester of First Year of the Programme**								
Course Objectives (Minimum 3)	1. This course will enable the students to acquire the important knowledge on grammar like the formation of sentences. 2. To enable the students to use vocabulary meaningfully for a successful conversation. 3. To establish Reputation and Rapport, a dress code session is much needed 4. The 3 P's (Planning, Prioritizing, and Performing) of Time Management will be taught to the students.								
CO1	Understand and identify common errors in English writing.								
CO2	Acquire skill of report writing.								
CO3	Develop the ability as critical readers and writers.								
CO4	Improve speaking ability in English both in terms of fluency and comprehensibility.								
CO5	Understand the correct usage of English grammar in writing and speaking.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Grammar** Interchange of Interrogative and Assertive Sentences, Interchange of Exclamatory and Assertive Sentences, Analysis of Sentences, Types of Tenses Exercises on Tense	**8**	Identify the structural differences between interrogative and assertive sentences. Convert assertive sentences into interrogative sentences and vice versa without altering the original meaning. Apply correct punctuation and syntax rules while transforming sentences from one form to another.	1,2,3
II	**Understanding Harmony in the Vocabulary** Synonyms, Antonyms, Homonyms	**12**	Define and understand what synonyms are and their role in language. Identify and provide examples of synonyms for given words. Use synonyms appropriately in various contexts to avoid repetition and enhance writing and speaking	3,4
III	**Reading Skills** Techniques of Effective Reading, Gathering ideas and information from a text, The SQ3RTechnique Interpret the text	**8**	Improve the ability to understand and interpret written texts across various genres and disciplines. Expand vocabulary by encountering and learning new words through reading diverse texts.	4,5
IV	**Dress Code Ethics** Introduction to Dress Code	**6**	Define dress code ethics and explain their relevance in various professional and social settings.	3,5

	Ethics, Purpose and Importance, How to Make first impression, What to Wear During Interviews or Any Other Formal Meetings.		Recognize how dress codes can vary across different cultures and the importance of being culturally sensitive in dress choices.	
V	**Time-Management Skills** Introduction To Time Management, Purpose And Importance of Time Management,Basic Tips to Maintain Time.	**5**	Define time management and explain its key principles. Explore the historical development of time management concepts and techniques.	3,5

TEXT BOOKS:

T1: Wren, P.C and Martin,H. 1995. High School English Grammar and Composition, S Chand Publishing.
T2: Barrett, Grant 2016 Perfect English Grammar: The Indispensable Guide to Excellent Writing and Speaking, Zephyros Press

REFERENCE BOOKS:

R1: Carthy. (2008) English Vocabulary in Use Upper - Intermediate with CD-ROM, Cambridge University Press
R2: Tracy, Brian. (2018) Time Management: The Brian Tracy Success Library, Manjul Publishing House

OTHER LEARNING RESOURCES: NIL

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and identify common errors in English writing.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
2	Acquire skill of report writing.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
3	Develop the ability as critical readers and writers.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
4	Improve speaking ability in English both in terms of fluency and comprehensibility.	PO1,PO2,PO3,PO4,PO5,PO10,PO12
5	Understand the correct usage of English grammar in writing and speaking.	PO1,PO2,PO3,PO4,PO5,PO10,PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24UBPD1204R	**PDP (Effective English for Engineers)**	**CO1**	1	1	1	1	1					3		2
		CO2	1	1	1	1	1					3		2
		CO3	1	1	1	1	1					3		2
		CO4	1	1	1	1	1					3		2
		CO5	1	1	1	1	1					3		2

SEMESTER – II									
Course Title	**Extra-Curricular Activities**								
Course code	**24UBEC1201**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30	**0**	**0**	**0**	**4**	**0**	**0**	**1**

Pre-requisite	Nil	**Co-requisite**	Nil	
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**			
Semester	**Winter/II Semester of First Year of the Programme**			
Course Objectives (Minimum 3)	1. It is to develop the social and soft skills and to promote a holistic development of the learners. 2. Participating in debate clubs, drama, or public speaking events enhances verbal and non-verbal communication skills. 3. Taking leadership roles in student organizations or event planning committees fosters leadership and teamwork abilities.			
CO1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.			
CO2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.			
CO3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.			
CO4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.			
CO5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Discover the exciting world of engineering through hands-on projects and workshops. Gain practical skills in robotics, coding, and innovation, preparing for future STEM success.	30	Explore the dynamic field of engineering through interactive projects and workshops. Develop practical skills in robotics, coding, and innovation. Prepare for future success in STEM disciplines.	1,2,3,4,5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Learn to a plan so that they can make meaningful contributions, maintain a commitment, and manage their time and priorities.	PO9
2	Transform passionate students who demonstrate leadership and pursue interests beyond their academics.	PO11,PO12
3	Learn to participate in various co-curricular activities leading to their multifaceted personality development.	PO12
4	Express their ideas, views, In-depth evaluation and analysis clearly in the topic of their interest.	PO10
5	Demonstrate and practices different activities, by Integrating learning experiences by demonstrating transferable skills.	PO9

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24UBEC1201	Extra-Curricular Activities	CO 1									2			
		CO 2											3	2
		CO 3												3
		CO 4										2		2
		CO 5									1			

SEMESTER – III									
Course Title	**Digital Electronics**								
Course code	**24BCSM2101R**	**Total credits: 3** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To familiarize with the different number systems, logic gates, and combinational and sequential circuits utilized in the different digital circuits and systems. 2. The course will help in design and analysis of the digital circuit and system.								
CO1	Understand Boolean algebra, analyze digital logic families, demonstrate IC interfacing.								
CO2	Design logic functions, implement digital circuits, showcase MSI chip expertise								
CO3	Analyse, design sequential circuits, demonstrate flip-flop and counter proficiency								
CO4	Evaluate digital-to-analog converters, understand quantization, A/D converter types.								
CO5	Analyse semiconductor memories, demonstrate proficiency in memory technology and PLDs.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Fundamentals of Digital Systems and logic families:** Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Boolean algebra, examples of IC gates, number systems-binary, signed binary, octal hexadecimal number, binary arithmetic, one's and two's complements arithmetic, codes, error detecting and correcting codes, characteristics of digital ICs, digital logic families, TTL, Schottky TTL and CMOS logic, interfacing CMOS and TTL, Tri-state logic.	**12**	Understand digital signals, circuits, logic gates, Boolean algebra, binary arithmetic, number systems, IC gates, logic families, and interfacing techniques.	**1,2,3**
II	**Combinational Digital Circuits:** Standard representation for logic functions, K-map representation, simplification of logic functions using K-map, minimization of logical functions. Don't care conditions, Multiplexer, De-Multiplexer/Decoders, Adders, Subtractors, BCD arithmetic, carry look ahead adder, serial adder, ALU, elementary ALU design, popular MSI chips, digital comparator, parity checker/generator, code converters, priority encoders, decoders/drivers	**10**	Understanding digital logic fundamentals including Boolean algebra, K-maps, multiplexers, adders, ALUs, and MSI chips for designing and optimizing digital circuits.	**1,2,3**

	for display devices, Q-M method of function realization.			
I	**Sequential circuits and systems**: A 1-bit memory, the circuit properties of Bistable latch, the clocked SR flip flop, J- K-T and D types flip flops, applications of flip flops, shift registers, applications of shift registers, serial to parallel converter, parallel to serial converter, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, counters design using flip flops, special counter IC's, asynchronous sequential counters, applications of counters.	**8**	Mastering digital memory elements (latches, flip-flops), shift registers, counters, and their applications for sequential logic design in digital systems.	**1,2,3,4,6**
IV	**A/D and D/A Converters:** Digital to analog converters: weighted resistor/converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter ICs, sample and hold circuit, analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D	**7**	Gaining expertise in digital to analog and analog to digital converters, encompassing specifications, operational principles, and practical applications in digital systems design.	**1,2,3,4,5**
V	**Semiconductor memories and Programmable logic devices:** Memory organization and operation, expanding memory size, classification and characteristics of memories, sequential memory, read only memory (ROM), read and write memory(RAM), content addressable memory (CAM), charge de coupled device memory (CCD), commonly used memory chips, ROM as a PLD, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).	**8**	Developing comprehensive knowledge of memory organization, types (ROM, RAM, CAM), memory characteristics, and applications in digital systems and programmable logic devices like PLDs and FPGAs.	**1,2,3,4,6**
Practical Component				
Practical 1	To study and verify the truth table of logic gates.	**2**	To effectively study and validate the truth	**2,3**

			tables of logic gates, ensuring proficiency in understanding their operational principles and applications in digital circuit design.	
Practical 2	To design and implementation using NAND gate and NOR gate as Universal Gate.	**2**	Designing and implementing circuits using NAND and NOR gates as universal gates, enhancing versatility in digital logic synthesis and problem-solving.	**3,4,5**
Practical 3	To study about Ex-OR gates and verify their Truth Table.	**2**	Studying XOR gates and verifying their truth table enhances understanding of their unique logic behavior critical for digital circuit design.	**2,3**
Practical 4	To study about adder circuits, half adder circuits and verify their truth table	**2**	Studying adder circuits, including half adders, and verifying their truth tables enhances proficiency in binary arithmetic fundamental to digital logic design.	**2,3**
Practical 5	To study about adder circuits, full adder circuits and verify their truth table.	**2**	Studying full adder circuits and verifying their truth tables strengthens understanding of binary addition, critical for designing complex arithmetic units in digital systems.	**2,3,4**
Practical 6	To study half subtractor using basic gates.	**2**	Studying half subtractors using basic gates enhances understanding of binary subtraction fundamentals, essential for digital logic design and arithmetic operations.	**2,3,4**
Practical 7	To study full subtractor using basic gates.	**2**	Studying full subtractors using basic gates enhances understanding of binary subtraction with borrow, crucial for designing efficient	**2,3,4**

			arithmetic circuits in digital systems.	
Practical 8	To design and set up a 4:1 and 2:1 Multiplexer.	**2**	Designing and setting up 4:1 and 2:1 multiplexer enhances understanding of data selection and routing in digital circuits, critical for signal processing applications.	**2,3,4,5,6**
Practical 9	To design and set up a 1:4 Demultiplexer (DE-MUX) .	**2**	Designing and setting up a 1:4 demultiplexer enhances understanding of data distribution and signal routing, crucial for digital communication and control systems.	**2,3,4,5,6**
Practical 10	Implementation and verification of decoder and encoder using logic gates.	**2**	Implementing and verifying decoders and encoders using logic gates enhances understanding of signal decoding and encoding techniques crucial for digital communication and control systems.	**2,3,4,5,6**
Practical 11	Mini-Project			

TEXT BOOKS:

T1: J.B Gupta, Electronic Device and Circuits, 2nd Ed., Katson Books 6th edition 2018.
T2: Dr. Sanjay Sharma, Digital Electronics and Logic Design, 6th Ed., Katson Books 2010.

REFERENCE BOOKS:

R1: Floyd, Electronic Devices, Pearson Education, 2nd edition, 9th edition, 2012.
R2: RP Jain, Modern Digital Electronics, Tata Mc Graw Hill, 3rd Edition, 2007.
R3: Frenzel, Communication Electronics: Principles and Applications, Tata Mc Graw Hill, 3rd Edition, 2001.
R4: Kumar, Fundamentals of Digital Circuits, Prentice Hall India, 2016.

OTHER LEARNING RESOURCES:

O1: NPTEL: Platforms like NPTEL offer courses such as “Digital Circuits and Systems” which cover topics ranging from digital circuits to system design.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand Boolean algebra, analyse digital logic families, demonstrate IC interfacing.	PO1, PO2, PO3, PO10, PO12
2	Design logic functions, implement digital circuits, showcase MSI chip expertise	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Analyse, design sequential circuits, demonstrate flip-flop and counter proficiency	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Evaluate digital-to-analog converters, understand quantization, A/D converter types.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Analyse semiconductor memories, demonstrate proficiency in memory technology and PLDs.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2101R	**Digital Electronics**	**CO1**	2	2	1							1		1
		CO2	3	3	2	2	1					1		2
		CO3	3	3	3	1	1					1		2
		CO4	3	3	2	2	1					1		2
		CO5	3	3	2	2	1					1		2

SEMESTER – III									
Course Title	**Discrete Mathematics and Graph Theory**								
Course code	**24BCSM2102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To provide students with a solid understanding of the fundamental concepts in discrete mathematics and graph theory. 2. To develop students' ability to solve problems related to discrete structures and graph algorithms. 3. To enable students to apply discrete mathematical concepts and graph theory techniques to computer science and related fields.								
CO1	Understand set operations, analyse relations, and demonstrate problem-solving in mathematics.								
CO2	Evaluate propositional logic, demonstrating analytical reasoning and deduction skills.								
CO3	Analyse algebraic structures, showcasing proficiency in groups, rings, and fields.								
CO4	Apply counting techniques, demonstrate problem-solving capabilities in mathematics.								
CO5	Analyse graph properties, apply graph theory concepts effectively in problem-solving.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Foundations of Discrete Mathematics** • Set Theory: Sets, subsets, operations on sets, Venn diagrams, Cartesian products. • Logic: Propositional logic, predicates, quantifiers, logical connectives, truth tables. • Proof Techniques: Direct proof, proof by contradiction, induction, contrapositive.	**4**	In Foundations of Discrete Mathematics, students will understand the basics of set theory, including sets, subsets, operations, Venn diagrams, and Cartesian products. They will learn propositional logic, predicates, quantifiers, logical connectives, and truth tables, along with mastering proof techniques such as direct proof, proof by contradiction, induction, and contrapositive.	**1,2,3**
II	**Combinatorics and Discrete Probability** • Combinatorics: Permutations, combinations, Pigeonhole principle, inclusion-exclusion principle. • Discrete Probability: Probability theory, conditional probability, Bayes' theorem, expected value.	**8**	Students will understand permutations, combinations, the Pigeonhole principle, and the inclusion-exclusion principle. They will also learn probability theory, conditional probability, Bayes' theorem, and how to calculate the expected value.	**2,5**

I	**Graph Theory Basics** • Graphs: Definitions, types of graphs, subgraphs, isomorphism. • Graph Representation: Adjacency matrix, adjacency list, incidence matrix. Trees: Properties, spanning trees, minimum spanning trees.	**10**	Students will learn the definitions and types of graphs, subgraphs, and graph isomorphism. They will understand graph representation using adjacency matrices, adjacency lists, and incidence matrices, and explore tree properties, spanning trees, and minimum spanning trees.	**1,2,3,4**
IV	**Advanced Graph Theory** • Graph Algorithms: Breadth-first search (BFS), depth-first search (DFS), Dijkstra's algorithm, Floyd-Warshall's algorithm. • Planarity and Colouring: Planar graphs, Euler's formula, graph colouring, chromatic number. Network Flows: Max-flow min-cut theorem, Ford-Fulkerson algorithm.	**7**	In Advanced Graph Theory, students will master graph algorithms such as BFS, DFS, Dijkstra's, and Floyd-Warshall. They will understand planar graphs, Euler's formula, graph colouring, chromatic number, and network flows including the max-flow min-cut theorem and the Ford-Fulkerson algorithm.	**2,4,6**
V	**Applications of Discrete Mathematics** • Applications in Computer Science: Data structures, algorithms, complexity. • Applications in Network Theory: Internet graphs, social networks, communication networks. • Additional Applications: Cryptography, coding theory, combinatorial optimization.	**7**	In Applications of Discrete Mathematics, students will explore its use in computer science for data structures, algorithms, and complexity, and in network theory for internet graphs, social networks, and communication networks. They will also understand applications in cryptography, coding theory, and combinatorial optimization.	**2,3**

TEXT BOOKS:

T1: Discrete Mathematics and Its Applications" by Kenneth H. Rosen

REFERENCE BOOKS:

R1: "Introduction to Graph Theory" by Douglas B. West.

OTHER LEARNING RESOURCES:

O1*:* *Discrete Mathematics for Computer Science Specialization* on Coursera: Offered by UC San Diego

and National Research University Higher School of Economics.

O2: MIT Open Course Ware's lecture notes on Discrete Mathematics and Graph Theory.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand set operations, analyse relations, and demonstrate problem-solving in mathematics.	PO1, PO2,PO12
2	Evaluate propositional logic, demonstrating analytical reasoning and deduction skills.	PO1, PO2,PO12
3	Analyse algebraic structures, showcasing proficiency in groups, rings, and fields.	PO1, PO2,PO12
4	Apply counting techniques, demonstrate problem-solving capabilities in mathematics.	PO1, PO2,PO12
5	Analyse graph properties, apply graph theory concepts effectively in problem-solving.	PO1, PO2,PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BCSM2102R	Discrete Mathematics and Graph Theory	CO1	1	3										1
		CO2	2	3										2
		CO3	2	3										2
		CO4		3										2
		CO5	2	3										2

SEMESTER – III									
Course Title	**Computer Organization and Architecture**								
Course code	**24BCSM2103R**	**Total credits: 4** **Total hours:** **45T + 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Programming For Problem Solving	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. Discuss the organization and architecture of computer systems and electronic computers. 2. Analyse the basic components of computer systems besides the computer arithmetic. 3. Discuss input-output organization, memory organization and management, and pipelining.								
CO1	Understand the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes.								
CO2	Analyse the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers.								
CO3	Analyse the control unit design approaches, memory design technologies and I/O transfers.								
CO4	Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.								
CO5	Summarize the concepts of memory organization with mapping functions and replacement algorithms.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Functional blocks of a computer:** CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU – registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs.	**8**	Understanding computer architecture components: CPU, memory, I/O subsystems, control unit. Mastering CPU's instruction set architecture, addressing modes, and interpreting instructions for diverse applications.	**1,2,3**
II	**Data representation:** signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic. Introduction to x86 architecture.	**8**	Understanding data representation (signed numbers, fixed/floating-point, character encoding) and computer arithmetic (addition, multiplication, division techniques, floating-point operations) for comprehensive computer system understanding.	**1,2,3**
I	**CPU control unit design:** hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU. Memory system design: semiconductor memory technologies, memory organization. Peripheral devices and their characteristics: Input-output	**10**	Understanding CPU control unit design (hardwired vs micro-programmed), memory system architecture, I/O subsystems, interfaces, interrupts, and exceptions for comprehensive	**1,2**

	subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB		computer system understanding and design.	
IV	**Pipelining:** Basic concepts of pipelining, throughput and speedup, pipeline hazards. Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.	**8**	Mastering pipelining fundamentals: concepts, throughput, speedup, and handling pipeline hazards. Understanding parallel processors, concurrent memory access, and cache coherency for advanced computational efficiency and performance.	**1,2**
V	**Memory organization:** Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.	**8**	Understanding memory organization: interleaving, hierarchical structure, cache memory, mapping techniques, replacement algorithms, and write policies for optimized memory access and performance in computer systems.	**1,2,3**

Practical Component				
Practical 1	**Exploring CPU Components:** Simulate the functional blocks of a CPU (ALU, registers, and control unit) using a simulator.	**3**	Understand the role and functionality of various CPU components. Analyse how instructions are executed within a CPU.	**1,2,4**
Practical 2	**Instruction Set Simulation:** Implement a subset of a CPU's instruction set (e.g., arithmetic and logical instructions) using a simulated environment.	**3**	Identify and explain the operation of basic instructions in a CPU. Simulate and interpret the results of instruction execution cycles.	3,4
Practical 3	**Data Representation and Conversion:** Write a program to convert between signed number representations (e.g., two's complement) and perform fixed and floating-point arithmetic.	**3**	Interpret different number systems and their representations. Solve arithmetic operations involving fixed and floating-point numbers.	**3,4**

Practical 4	**Designing Adders:** Design and simulate a ripple carry adder and a carry look-ahead adder using a hardware description language (HDL) like Verilog or VHDL.	**3**	Understand the concept of integer addition and analyse the performance of different adder designs.	**2,3**
Practical 5	**Hardwired vs. Microprogrammed Control Unit Design:** Design and compare a hardwired control unit and a microprogrammed control unit for a simple hypothetical CPU.	**3**	Understand the differences between hardwired and microprogrammed control units. Analyse the trade-offs in terms of speed and flexibility.	**4,5**
Practical 6	**Memory Hierarchy Simulation:** Simulate the organization of hierarchical memory (cache, main memory) and analyse the impact of cache size, block size, and replacement policies.	**3**	Proficiency in Python data structures: lists, tuples, dictionaries, and sets.	**1,2,3**
Practical 7	**Booth's Multiplier Implementation:** Implement Booth's multiplication algorithm in a programming language like Python or C++.	**3**	Understand and apply Booth's algorithm for efficient integer multiplication. Evaluate the advantages of Booth's method over basic shift-and-add techniques.	**3,5**
Practical 8	**Pipelining Simulation:** Simulate a basic instruction pipeline using a programming language or simulation tool to understand pipeline hazards (data, control, structural).	**3**	Understand pipelining concepts, hazards, and their impact on performance. Analyse how hazards affect throughput and suggest techniques to mitigate them.	**2,4**
Practical 9	**Exploring x86 Architecture:** Use an x86 assembler (e.g., NASM) to write and execute simple programs demonstrating arithmetic, logical, and control flow instructions.	**3**	Explore the x86 instruction set and its addressing modes. Develop basic assembly language programs.	**1,2,3**
Practical 10	**I/O Device Interface:** Simulate an I/O transfer mechanism using program-controlled, interrupt-driven, and DMA modes, and analyse their performance.	**3**	Understand the differences between program-controlled, interrupt-driven, and DMA-based I/O transfers.	**2,5**

			Evaluate the efficiency of different I/O modes.	

TEXT BOOKS:

T1: David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", 5th Edition, Elsevier.

REFERENCE BOOKS:

R1: "Computer Organization and Embedded Systems", 6th Edition by CarlHamacher, McGraw Hill Higher Education.
R2: "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill.
R3: "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

OTHER LEARNING RESOURCES:

O1: javatpoint: the tutorial "Computer Architecture and Organization"
O2: NPTEL: "**Computer Organization and Architecture"** by Prof. Mainak Chaudhuri (IIT Kanpur)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the interaction of the components of a computer system with the instruction set architecture of a CPU and addressing modes.	PO1, PO2, PO3, PO10, PO12
2	Analyse the various data representation techniques, perform computer arithmetic and discuss the various adders and multipliers.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Analyse the control unit design approaches, memory design technologies and I/O transfers.	PO1, PO2, PO3, PO10, PO12
4	Demonstrate the concepts of pipelining, parallel processing and concurrent access to memory.	PO1, PO2, PO3, PO4, PO10, PO12
5	Summarize the concepts of memory organization with mapping functions and replacement algorithms.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**

24BCSM2103R	**Computer Organization and Architecture**	**CO1**	2	3	1							1		3
		CO2	3	3	3	2	2					1		3
		CO3	2	2	2							1		3
		CO4	2	2	3	2						1		3
		CO5	2	2	2	2	1					1		3

SEMESTER – III									
Course Title	**Field-Based Learning**								
Course code	**24BCSM2104R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 16	**0**	**0**	**0**	**0**	**0**	**16**	**1**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To develop problem-solving skills in practical settings. 4. To foster collaboration and communication skills through group projects and professional interactions.								

	5. To cultivate an appreciation for the complexities and challenges of professional practice in the field.			
CO1	Apply theoretical concepts to real-world situations.			
CO2	Conduct field-based research and gather data effectively.			
CO3	Analyse and interpret the university data to draw meaningful conclusions.			
CO4	Communicate findings clearly and effectively, both orally and in writing.			
CO5	Demonstrate professional behaviour and teamwork skills in field settings.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	The organization will provide the students one field visit for this program. The students will be taken to a specific blocks of the university for learning purposes. The students will visit all department block site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners of IT department of the university. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.	16	Enhanced understanding through immersive field visits, interaction with IT professionals, and reflective journaling to integrate theoretical knowledge with practical insights effectively.	1,2,3,4,5,6

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.	PO1, PO5, PO7, PO10, PO12
2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.	PO2, PO4, PO5, PO6, PO12
3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.	PO2, PO3, PO4, PO9, PO11
4	Develop and deliver professional presentations using appropriate engineering language and communication	PO6, PO8, PO9, PO10, PO11

	techniques.	
5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.	PO6, PO7, PO8, PO11, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM2104R	Field-Based Training	CO 1	3				2		1			1		1
		CO 2		3		3	2	1						2
		CO 3		3	3	2					2		1	
		CO 4						1		1	3	3	2	
		CO 5						3	3	2			1	2

SEMESTER – III									
Course Title	**PDP (English For Employability For Engineers)**								
Course code	**24UBPD2103R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives	1. To introduce the types of sentences and their significance. 2. To strengthen the vocabulary of the students to enhance student' vocabulary to enhance their speaking and writing skills it the importance of dress codes in various organizations. 3. To overcome fear of Public Speaking 4. To understand the process of Effective Reading techniques and Listening skills.								
CO1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.								
CO2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.								
CO3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.								
CO4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.								
CO5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Reading Skills:** Exercise 1: Interpreting different texts Exercise 2: Importance and significance of Punctuation Exercise 3: Story -reading Exercise 4: News-reading Exercise 5: Comprehending texts of different genres	**6**	Enhancing reading skills: interpreting texts, understanding punctuation, reading stories and news, and comprehending various genres for improved literacy.	**1,2,3**
II	**Non-Verbal Communication:** i. Understanding Non-Verbal Communication and Body Language Types of Body Language ii. Importance and Impact of Body Language and introduction to Haptics, Kinesics and Proxemics	**6**	Understanding non-verbal communication: types and impact of body language, and introduction to haptics, kinesics, and proxemics for effective interaction.	**1,2,3**
I	**Grammar (Flipped Classroom):** i. Types of Sentences (using Assertive, Imperative, exclamatory in respective contexts), ii. Degrees of Comparison	**4**	Understanding grammar through flipped classroom: identifying types of sentences (assertive, imperative, exclamatory) and mastering degrees of comparison in context.	**2,3**
IV	**Public Speaking Skills:** i. Introduction to public speaking Preparation for Public speaking (scripts, non-verbal cues) ii. Understanding and overcoming Fear of Public Speaking	**7**	Developing public speaking skills: preparation (scripts, non-verbal cues), overcoming fear, and applying effective public speaking tips for confident presentations.	**2,3**

	Tips Public Speaking			
V	**Listening Skills:** i. Process of listening ii. Understanding listening barriers I. Difference between Listening and Hearing, iv. Importance of Effective Listening Practice Session: Listening to podcasts, Lectures, Audio Books etc.	**5**	Enhancing listening skills: understanding the listening process, identifying barriers, distinguishing listening from hearing, and practicing with podcasts, lectures, and audiobooks.	**2,3,4**

TEXT BOOKS:

T1: Wren,P.C and Martin,H. 1995. High School English Grammar and Composition, S Chand Publishing.
T2: Barrett,Grant.2016.PerfectEnglishGrammar:TheIndispensibleGuidetoExcellent Writing and Speaking, Zephyros Press

REFERENCE BOOKS:

R1: Carthy. (2008) English Vocabulary in Use Upper - Intermediate with CD-ROM, Cambridge University Press
R2: Tracy, Brian. (2018) Time Management: The Brian Tracy Success Library, Manjul Publishing House

OTHER LEARNING RESOURCES:

O1: NPTEL- the course "Technical English for Engineers" by Prof. Aysha Iqbal, IIT Madras
O2: youtube- "Your body language may shape who you are" Amy Cuddy ,TED
O3: Coursera- "Training and Practicing in English Public Speaking" by Shanghai Jiao Tong University

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Create effective written communication for professional scenarios, incorporating engineering terminology and conventions.	PO1, PO5, PO9, PO10
2	Demonstrate proficient verbal communication skills, applying engineering vocabulary and technical language appropriately.	PO6, PO9, PO10, PO12
3	Evaluate and apply effective teamwork strategies in engineering contexts, fostering collaboration and innovation.	PO6, PO9, PO10, PO11
4	Develop and deliver professional presentations using appropriate engineering language and communication techniques.	PO5, PO10, PO12
5	Analyse and adapt communication strategies for diverse engineering workplace scenarios, demonstrating cross-cultural competence.	PO6, PO8, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24UBPD2103R	**English For Employability for Engineers**	**CO1**	1				1				1	3		
		CO2						2			1	3		1
		CO3						2			3	2	1	
		CO4					1					2		3
		CO5						3		1		3		2

SEMESTER – III									
Course Title	**Financial Literacy**								
Course code	**24UBPD2104R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	Basic Financial knowledge	**Co-requisite**	**NIL**						
Programme	**Bachelor of Technology and Computer Science and Engineering**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. The course would offer an inclusive approach to understand the relevant concepts of money, borrowing, lending, taxes and their application to financial planning. 2. Assess the personal financial planning process, the life cycle of financial plans, and methods of goal achievement. 3. Formulate a budget, record-keeping system, and tax planning strategy based on current financial goals.								
CO1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.								
CO2	Design a diversified investment portfolio that addresses several different investment objectives.								
CO3	Differentiate between open and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.								
CO4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.								
CO5	Apply financial knowledge and skills to make informed decisions, ensuring long-term financial stability and security.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Fundamentals of Financial Planning - i. Functions of money; ii. Inflation- Meaning, causes, how it can be controlled; I.process official planning, iv. Time value of money-simple and compound interest; v.Net Present Value and Future value, vi. Power of Compounding; vii. Doubling period and Rule of 72.	**8**	Fundamentals of Financial Planning covers understanding money's functions, inflation causes and control, financial planning processes, time value of money including simple and compound interest, Net Present Value (NPV) and Future Value (FV), compounding power, doubling periods, and the Rule of 72 for effective financial decision-making.	**1,2, 3**
II	Income Tax Planning- i. Meaning of Income, ii. Direct and Indirect Taxes, Taxable Income, various heads of Income for tax Calculation, I. Non-taxable Income, iv. Tax evasion and tax avoidance, v.GST, Tax Planning Strategies.	**8**	Learning Outcome: Gain proficiency in understanding income, taxes (direct, indirect), taxable and non-taxable income, tax evasion, avoidance, GST, and effective tax planning strategies.	**3,4**
I	Entrepreneurial planning - i. Meaning of Entrepreneurship, prerequisites for becoming an entrepreneur, ii. Entrepreneurship Support Systems in India,	**10**	Understand entrepreneurship fundamentals, support systems in India, institutional and financial support mechanisms (venture capital, business angels, government	**3,4**

	I. Institutional support systems for entrepreneurs, iv. Financial support systems for entrepreneurs; v. Venture Capital, Business Angels, vi. Assistant of Government, vii. Commercial Bank Loans and Overdraft.		assistance, commercial bank loans), enabling effective entrepreneurial planning and implementation.	
IV	Planning for investing in securities market - i. Investment avenues offered by Securities Markets.. Primary Market and Secondary Market, ii . Stock market- meaning, features, functions of NSE, BSE DEMAT trading account, I. Security repository, stock brokers. Operational aspects of securities markets: placement of orders, contract note, pay-in and pay-out, trading and settlement cycle, iv. Various risks involved in investing in securities markets; Role of Financial Intermediaries; Stock indices. v. Mutual Funds- meaning concept, definition, types, importance and drawbacks of mutual funds, mutual funds in India, investing in mutual funds, vi. Systematic Investment Plan (SIP) and its advantages.	**10**	Gain knowledge of investment avenues in securities markets, primary and secondary markets, stock market functions (NSE, BSE), DEMAT trading, security repositories, stock brokers, operational aspects (order placement, contract notes, trading cycles), risks, financial intermediaries, stock indices, mutual funds (types, importance, drawbacks, investing in India), and advantages of Systematic Investment Plans (SIPs).	**4**
V	Planning for debts and Retirement i. Consumer credit Introduction to consumer credit; choosing a source of credit, the cost of credit alternatives, ii. Consumer Legal Protection; I. Housing Decision: Factors and Finance: Vehicle Decisions. iv. Retirement planning Meaning of cost of living; retirement need analysis; development of retirement plan, various retirement schemes. v. Estate Planning; Pension and Medicare Planning; Wills.	**9**	Develop skills in consumer credit management, including choosing credit sources and evaluating costs. Understand consumer legal protections and factors influencing housing and vehicle decisions. Master retirement planning, including cost-of-living considerations, retirement needs analysis, retirement scheme evaluation, estate planning, pension, Medicare planning, and wills.	**4**

TEXT BOOKS:

T1: Sinha Pradeep K. and Priti Sinha. Computer Fundamentals: Concepts Systems andThe Million-Dollar Financial Advisor: Powerful Lessons and Proven Strategies from Top Producers by David J. Mullen Jr
T2: Personal Finance and Planning by Dr. Rajni

REFERENCE BOOKS:

R1: Be Your Own Financial Advisor: Financial Planning, Investment Options, Risk Management, Tax Management, Succession Planning Kindle Edition y Sushil Bali
R2: The Dumb Things Smart People Do with Their Money: Thirteen Ways to Right Your Financial Wrongs Kindle Edition y Jill Schlesinger

OTHER LEARNING RESOURCES:

O1: Financial Planning Association (FPA):Offers resources, workshops, and events for financial planners and individuals seeking financial advice.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop a cash management strategy and a plan to facilitate the home or automobile buying process.	PO2, PO9, PO10, PO11
2	Design a diversified investment portfolio that addresses several different investment objectives.	PO2, PO5, PO11, PO12
3	Differentiate between open and closed-end mutual funds, exchange-traded funds, and direct or indirect real estate investments.	PO1, PO2, PO6, PO7
4	Create a financial plan that covers your income needs in retirement and helps protect you and your estate.	PO2, PO6, PO8, PO11, PO12
5	Apply financial knowledge and skills to make informed decisions, ensuring long-term financial stability and security.	PO2, PO6, PO8, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24UBPD2104R	**Financial Literacy**	**CO 1**		2							1	1	3	
		CO 2		3			2						3	1
		CO 3	2	2				1	1					
		CO 4		2				1		1			3	2
		CO 5												

SEMESTER – III									
Course Title	**Basic Life Saving Skills**								
Course code	**24UULS2102R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I Semester of Second Year of the Programme**								
Course Objectives (Minimum 3)	1. To equip students with essential knowledge of emergency response protocols. 2. To develop skills for administering CPR and first aid in critical situations. 3. To familiarize students with the use of automated external defibrillators 4. To train students in recognizing and responding to medical emergencies effectively. 5. To promote confidence and preparedness in handling life-threatening scenarios.								
CO1	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives								
CO2	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life								
CO3	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders, and anyone involved in emergency care								
CO4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate ,delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.								
CO5	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction(heart attack),stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Basic Life Support(BLS)** ● Introduction of BLS ● Chain of survival ● ABCs Assessment ● CPR and Ventilation Technique ● AED ● Choking for adult and children	5	Understanding chain of survival, ABCs assessment, CPR techniques, ventilation, AED usage, and choking management for adults and children.	**1,2,3**
II	**Soft skills** ● Introduction ● Communications Skills ● Situational Skills ● Team Work	5	Enhanced communication abilities, improved situational awareness, strengthened teamwork skills, and mastery of other interpersonal competencies crucial for professional success.	**3,4**

I	**Trauma emergencies** • Scene safety • Primary assessment • Bleeding control • Helmet removal • Care of amputated body part • Extrication of victims and safe transfer • Cervical spine stabilizatio n • Cervical collar application • Splinting of broken Limbs	10	Mastering scene safety assessment, prioritizing initial approaches in pre-hospital care such as primary assessment and bleeding control, and executing procedures like helmet removal, care of amputated body parts, extrication, cervical spine stabilization and collar application, and splinting of broken limbs for effective patient management and safety.	**3,4**
IV	**Triage system** • Introduction • Flow chart approach of Triage • Triage of Multiple Casualties in Pre-Hospital setting • Triage of Single casualty	5	Understanding the introduction to triage, mastering the flowchart approach for triage, effectively triaging multiple casualties in a pre-hospital setting, and appropriately triaging a single casualty to optimize emergency response and patient care outcomes.	**1,2,3,4**
V	**Medical emergencies** • Introduction • Victim centred approach in medical emergency • Diabetic emergencies • emergency child birth stroke recovery position	5	Learning victim-centered medical emergency approaches, managing seizures, heart attacks, asthma, diabetic emergencies, emergency childbirth, and stroke recovery positioning for effective patient care.	**2,3,4**

TEXT BOOKS:

T1: Nancy Caroline, Emergency Care in the streets, Seventh edition by Jones and Bartlett

T2: LC Gupta, First Aid book

REFERENCE BOOKS:

R1: Advance Cardio vascular life support and Basic life support provider manual, American Heart Association (AHA)

R2: Wikipedia - Basic life support

OTHER LEARNING RESOURCES:

O1: youtube- "Basic Life Support (BLS) CPR and AED Training" by the American Heart Association

O2: **youtube- "First Aid Training"** by St John Ambulance

O3: **Coursera- "Introduction to First Aid"** by the University of Michigan

O4: **Udemy- "Complete BLS Certification Course - Basic Life Support"** by First Aid For Free

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Basic Life Support (BLS) are the specific skills and knowledge that individuals should acquire and demonstrate after completing a BLS training course. These outcomes are essential for anyone who may encounter a medical emergency and need to provide immediate assistance before professional medical help arrives	PO1, PO6, PO8, PO10, PO12
2	Soft skills are a set of personal attributes and abilities that enable students to interact effectively and harmoniously with others, both personally and professionally. These skills are highly valued and can lead to better communication, collaboration, and overall success in various aspects of life	PO6, PO9, PO10, PO12
3	Focus on the knowledge and skills needed to effectively recognize, assess, and manage various types of traumatic injuries and emergencies. Essential for healthcare professionals, first responders, and anyone involved in emergency care	PO2, PO4, PO10, PO12
4	Understand the principles and purpose of the Triage system in healthcare settings. Describe the different levels of triage (e.g., immediate ,delayed, minimal, expectant) and their significance in prioritizing patient care. Identify common medical conditions and injuries that require immediate attention and those that can be managed at a later stage. Recognize the importance of accurate and timely patient assessment during triage.	PO2, PO6, PO10, PO12
5	Identify common medical emergency conditions: Learners should be able to recognize and differentiate between common medical emergencies, including myocardial infarction(heart attack),stroke, diabetic emergencies, anaphylaxis, respiratory distress, seizures, and allergic reactions.	PO1, PO2, PO6, PO8, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
23UULS211R	Basic Life Saving Skills	CO 1	2					3		1		2		1
		CO 2						1			3	3		1
		CO 3		3		3						2		2
		CO 4		3				3				2		2
		CO 5	2	3				3		1				2

SEMESTER III									
Course Title	**Logical Reasoning for Computer Science**								
Course code	**24BCSM2107R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the Second year of the Programme**								
Course Objectives	1. To introduce the fundamental concepts of logical reasoning and its importance in problem-solving for computer science. 2. To develop mathematical, analytical, and abstract reasoning skills for computational thinking. 3. To enable students to analyze critical thinking scenarios and evaluate solutions for data sufficiency problems. 4. To enhance students' ability to recognize patterns and relationships for effective decision-making. 5. To apply logical reasoning techniques in programming, debugging, and optimization of algorithms.								
CO1	Understand the foundational concepts of logical reasoning and their applications in computer science.								
CO2	Apply mathematical and analytical reasoning skills to solve complex problems.								
CO3	Analyze and evaluate critical thinking problems and data sufficiency scenarios.								
CO4	Recognize and utilize abstract reasoning and pattern recognition for logical problem-solving.								
CO5	Integrate logical reasoning techniques in programming and algorithm optimization for real-world applications.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Introduction to logical reasoning, types, and applications in computer science. Number Series, Alphabet Series, Coding-Decoding, and Direction Sense problems. Logical Deductions: Syllogisms, Statements, and Assumptions.	**8**	Understand the fundamentals of logical reasoning and apply them to basic problems.	**1,2,3**
II	Blood Relations, Seating Arrangements (Linear and Circular), Puzzles. Basic Arithmetic Problems: Percentage, Ratio and Proportion, Profit and Loss. Analytical Reasoning: Ranking, Order, and Arrangement Problems.	**12**	Apply mathematical and analytical reasoning skills to solve computational problems.	**3,4**
I	Data Sufficiency Problems: Understanding assumptions and evaluating statements.	**12**	Analyze data sufficiency scenarios and solve critical thinking problems for decision-making.	**3,4,5**

	Critical Thinking: Logical Cause and Effect, Evaluating Arguments, Identifying Summeracies. Logical Connectives: AND, OR, NOT, and implications in real-world applications.			
IV	Pattern Completion, Figure Series, Mirror and Water Images, and Shape Analogy. Logical Problems Involving Matrices, Odd One Out, and Logical Grouping of Objects. Identifying Relationships in Structured and Unstructured Data.	**12**	Develop abstract reasoning skills to identify patterns and relationships in problem-solving.	**3,4**
V	Problem Solving with Algorithms: Understanding Flowcharts, Decision Trees, and Logical Conditions in Programming. Real-World Scenarios: Scheduling, Network Flow, Resource Allocation, and Optimization Problems. Case Study: Application of logical reasoning in debugging and optimizing computational problems.	**12**	Apply logical reasoning techniques in programming, algorithms, and problem-solving for computer science.	**4,5**

Textbooks

T1: Aggarwal, R. S. - "A Modern Approach to Logical Reasoning", S. Chand Publishing, 2021.

Reference Books

R1: Sharma, A. - "Logical Reasoning and Data Interpretation for CAT", McGraw-Hill, 2022.
R2: Skiena, S. S. - "The Algorithm Design Manual", Springer, 3rd Edition, 2020.

Other Learning Resources

O1: Online Platforms:
- Brilliant.org: Interactive modules on logical reasoning and problem-solving.
- GeeksforGeeks: Logical reasoning for competitive programming and interviews.

O2: Video Tutorials:
- NPTEL: "Problem Solving and Programming" series by IITs.
- YouTube Channels: "Unacademy Logical Reasoning" for practical applications.

O3: Practice Tools:
- LeetCode: Logical reasoning problems in coding.
- HackerRank: Challenges on data sufficiency and pattern recognition.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping

SN	Course Outcome (CO)	Mapped Program Outcome
1	Understand the foundational concepts of logical reasoning and their applications in computer science.	PO1, PO5, PO12
2	Apply mathematical and analytical reasoning skills to solve complex problems.	PO1, PO5, PO12
3	Analyze and evaluate critical thinking problems and data sufficiency scenarios.	PO1, PO5, PO12
4	Recognize and utilize abstract reasoning and pattern recognition for logical problem-solving.	PO1, PO5, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM2107R	Logical Reasoning for Computer Science	CO1	3				3							2
		CO2	2				2							1
		CO3	2				2							1
		CO4	2				2							1
		CO5												

SEMESTER – III									
Course Title	**Cloud Fundamentals**								
Course code	**24BCSM2106R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the second year of the programme**								
Course Objectives (Minimum 3)	1. To introduce the basic concepts, characteristics, and deployment models of cloud computing. 2. To familiarize students with cloud platforms and their core services like virtual machines, storage, and containers. 3. To enable students to deploy, monitor, and manage cloud-based resources. 4. To understand cloud security practices, including identity and access management. 5. To explore emerging trends such as serverless computing and cost optimization strategies in the cloud.								
CO1	Understand the fundamental concepts and deployment models of cloud computing.								
CO2	Deploy and manage virtual machines, storage, and containers on a cloud platform.								
CO3	Monitor and optimize the usage of cloud resources effectively.								
CO4	Implement basic security practices and access control for cloud resources.								
CO5	Apply serverless computing and cost management techniques in cloud-based applications.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to cloud computing:** - Definition and characteristics of cloud computing (on-demand, scalability, resource pooling, etc.). - Types of cloud services: IaaS, PaaS, SaaS. - Cloud deployment models: Public, Private, Hybrid, and Community Cloud.	**8**	Understand the basic concepts, characteristics, and types of cloud computing.	2,3
II	**Cloud Infrastructure and Virtualization :** - Basics of virtualization: Virtual machines, Hypervisors, and Containers. - Components of cloud infrastructure: Servers, storage, and networking. - Introduction to cloud platforms (e.g., AWS, Azure, Google Cloud).	**8**	Understand the role of virtualization in cloud infrastructure and its components.	2,3
III	**Cloud Services and Deployment:** - Overview of popular cloud services (e.g., AWS EC2, S3, RDS). - Application hosting in the cloud. - Pricing models and cost management in cloud computing.	**7**	Explore the features and use cases of cloud services and deployment strategies.	3,4

IV	**Security and Governance in the Cloud:** - Cloud security fundamentals: Data protection, identity management, and access control. - Compliance and regulatory considerations (e.g., GDPR, HIPAA). - Shared responsibility model in cloud security.	**7**	Analyze the security challenges in cloud computing and implement governance models.	4,5
V	**Future Trends and Emerging Technologies in the Cloud:** - Edge computing and hybrid cloud integration. - Cloud-native applications: Microservices and serverless computing. - Role of Artificial Intelligence (AI) and Machine Learning (ML) in cloud computing.	**7**	Identify the trends and technologies shaping the future of cloud computing.	4,5
Practical Component				
Practical 1	**Introduction to Cloud Platforms:** Create a free account on a cloud platform (e.g., AWS, Azure, or Google Cloud) and explore its dashboard.	**3**	Understand the basic interface and services offered by a cloud platform.	**1,2,3**
Practical 2	**Deploying a Virtual Machine in the Cloud:** Launch a virtual machine instance on a cloud platform, configure its operating system, and connect to it remotely.	**3**	Learn to deploy and manage virtual machines on a cloud platform.	**3,4**
Practical 3	**Exploring Cloud Storage Services:** Upload, organize, and manage files using a cloud storage service (e.g., AWS S3 or Google Cloud Storage).	**3**	Understand cloud storage concepts and learn to use storage services for file management.	**3**
Practical 4	**Creating and Using Container:** Set up and deploy a container using Docker on a cloud platform and test its functionality.	**3**	Learn the basics of containerization and how it integrates with cloud platforms.	**3,4**
Practical 5	**Hosting a Static Website in the Cloud:** Host a simple static website on a cloud platform using a storage service (e.g., AWS S3 or Azure Blob Storage).	**3**	Understand the process of hosting websites on cloud platforms.	**3,4**
Practical 6	**Configuring Identity and Access Management** Create user roles and policies on a cloud platform and test access control for different users.	**3**	Learn how to manage users and permissions for cloud resources securely.	3,4,5
Practical 7	**Cost Estimation and Budgeting:** Use a cloud platform's cost estimation tool to estimate the cost of running a specific workload and set budget alerts.	**3**	Analyze pricing models and implement cost management strategies for cloud services.	**3,4**

Practical 8	**Monitoring Cloud Resource Usage:** Set up monitoring tools (e.g., CloudWatch on AWS) to track resource usage and performance metrics.	**3**	Understand how to monitor and optimize cloud resource utilization.	**3,5**
Practical 9	**Securing Data in the Cloud:** Implement basic encryption and enable data protection features for a storage service.	**3**	Learn the basics of securing data in the cloud using encryption and access control.	**4,5**
Practical 10	**Exploring Serverless Computing:** Deploy a serverless function (e.g., AWS Lambda or Google Cloud Functions) to process a simple event-driven task.	**3**	Understand the concept of serverless computing and its applications.	**3,4**

Textbooks

T1: Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi - *"Mastering Cloud Computing: Foundations and Applications Programming"*, McGraw-Hill, 2013.
T2: Thomas Erl - *"Cloud Computing: Concepts, Technology & Architecture"*, Pearson Education, 2013.

Reference Books

R1: Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra - *"Distributed and Cloud Computing: From Parallel Processing to the Internet of Things"*, Morgan Kaufmann, 2012.
R2: Anthony Velte, Toby Velte, and Robert Elsenpeter - *"Cloud Computing: A Practical Approach"*, McGraw-Hill, 2010.

Other Learning Resources

1. **Online Platforms:**
   - *AWS Cloud Practitioner Essentials* (AWS Training).
   - *Google Cloud Fundamentals* (Coursera).
2. **Web Resources:**
   - **Microsoft Learn**: Free cloud computing tutorials for Azure.
   - **AWS Documentation**: Comprehensive guides and use cases for Amazon Web Services.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand fundamental cloud computing concepts and architectures.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Analyze and implement RESTful APIs and data services on cloud platforms.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Develop and deploy applications using Python and related frameworks.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BCSM2106R	**Cloud Fundamentals**	**CO1**	3	3	3	2	3					1		3
		CO2	2	2	2	2	2					1		2
		CO3	3	2	3	2	3					1		3
		CO4	3	3	3	2	3					1		3
		CO5	3	3	3	3	3					1		3

<table>
<tr><th colspan="10">SEMESTER – III</th></tr>
<tr><td>Course Title</td><td colspan="9">Mini Project-I</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BCSM2105R</td><td>Total credits: 1</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>Total hours: 48</td><td>0</td><td>0</td><td>0</td><td>4</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="7">Nil</td></tr>
<tr><td>Programme</td><td colspan="9">Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)</td></tr>
<tr><td>Semester</td><td colspan="9">Summer/I Semester of Second Year of the Programme</td></tr>
<tr><td>Course Objectives</td><td colspan="9">1. To provide hands-on experience in identifying, analyzing, and solving real-world problems using the knowledge acquired during the course.
2. To enhance project planning, execution, and time management skills through teamwork and collaboration.
3. To enable students to apply theoretical concepts and technical skills to develop a functional solution or prototype.
4. To improve research, critical thinking, and problem-solving skills in a practical setting.
5. To strengthen communication and documentation skills by preparing project reports and presenting project outcomes effectively.</td></tr>
<tr><td>CO1</td><td colspan="9">Apply IT principles for real-world problem-solving using programming languages and network solutions.</td></tr>
<tr><td>CO2</td><td colspan="9">Contribute to complex projects as a team member, demonstrating effective personal and team management.</td></tr>
<tr><td>CO3</td><td colspan="9">Develop programming skills to create and assess software, hardware, and network solutions.</td></tr>
<tr><td>CO4</td><td colspan="9">Apply personal and team management skills as a professional software developer.</td></tr>
<tr><td>CO5</td><td colspan="9">Cultivate employability skills and uphold professionalism in software development.</td></tr>
<tr><td>Unit-No.</td><td colspan="5">Content</td><td>Contact Hour</td><td colspan="2">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="5">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
a) theoretical studies
b) computer simulations
c) hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;</td><td>48</td><td colspan="2">Learning about independent research, technical expertise application, and effective communication through comprehensive project execution and presentation in specialized technical areas.</td><td>1,2,3,4, 5,6</td></tr>
</table>

	5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2105R	**Mini Project-I**	**CO 1**	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

SEMESTER – IV									
Course Title	**Database Management Systems**								
Course code	**24BCSM2201R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	File Systems	**Co-requisite**	NIL						

Programme	Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)			
Semester	**Winter/ II semester of the second year of the Programme**			
Course Objectives (Minimum 3)	1. To understand and use data manipulation language to query, update, and manage a database. 2. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency 3. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.			
CO1	Understand the fundamental concepts of database management systems with DBMS languages and data models.			
CO2	Apply the concepts of query languages such as DDL and DML for designing a relational database.			
CO3	Apply the ACID properties and concurrency control schemes to perform transaction processing with database recovery.			
CO4	Apply security to the database by checking for authentication and authorization			
CO5	Apply advanced topics of data warehousing and data mining, distributed databases and web databases.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Database system architecture:** Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). **Data models:** Entity-relationship model, network model, relational and object-oriented data models, integrity constraints, data manipulation operations.	**6**	In studying Database System Architecture, students will understand data abstraction, data independence, and the roles of DDL and DML. They will also learn various data models, including entity-relationship, network, relational, and object-oriented models, and comprehend integrity constraints and data manipulation operations.	**1,2**
II	**Relational query languages:** Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. **Relational database design:** Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. **Query processing and optimization:** Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.	**10**	Students will gain proficiency in relational query languages (relational algebra, tuple and domain relational calculus, SQL3), understand DDL and DML constructs across MySQL, Oracle, DB2, and SQL Server. They will also learn effective relational database design (domain and data dependency, Armstrong's axioms, normal forms, dependency preservation, lossless design) and optimize query processing through evaluation, equivalence,	**2,3,4**

			join strategies, and algorithms.	
III	**Transaction processing:** Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Multi-version and optimistic, Concurrency Control schemes, Database recovery.	**8**	Students will comprehend transaction processing concepts such as concurrency control, ACID properties, serializability of scheduling, and locking mechanisms in timestamp-based and multi-version concurrency control schemes. They will also master database recovery techniques essential for maintaining data integrity and availability.	**2,3,4,5**
IV	**Database Security:** Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.	**6**	Students will understand database security fundamentals including authentication, authorization, DAC, MAC, and RBAC models, as well as techniques for intrusion detection and prevention of SQL injection attacks.	**2,3**
V	**Advanced topics:** Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.	**5**	Students will explore advanced topics including object-oriented and object-relational databases, logical databases, web databases, distributed databases, and data warehousing, alongside principles and techniques of data mining for comprehensive data management solutions.	**1,2,3**
Practical Component				
Practical 1	Designing and implementing a relational database schema based on business requirements, ensuring normalization up to 3NF.	**2**	Students will achieve proficiency in designing and implementing relational database schemas tailored to business requirements, ensuring data organization adheres to normalization principles up to the third normal form (3NF).	2,3
Practical 2	Writing and executing SQL queries to retrieve, update, and delete data from a relational database, using DML commands such as SELECT,	**2**	Students will master the skill of writing and executing SQL queries to efficiently retrieve, update,	3,5

	INSERT, UPDATE, DELETE.		and delete data from relational databases, employing DML commands (SELECT, INSERT, UPDATE, DELETE) to manipulate and manage data effectively.	
Practical 3	Implementing concurrency control mechanisms like locking and timestamp-based protocols to manage simultaneous access to data in multi-user environments.	**2**	Students will develop the capability to implement concurrency control mechanisms such as locking and timestamp-based protocols, ensuring consistent and reliable management of data access in multi-user environments to prevent conflicts and maintain data integrity.	3,4
Practical 4	Performing query optimization by analyzing query plans, selecting appropriate join strategies, and applying optimization algorithms (e.g., cost-based optimization).	**2**	Students will learn to optimize query performance by analyzing query plans, selecting optimal join strategies, and applying algorithms such as cost-based optimization, enhancing database efficiency and responsiveness.	4,5
Practical 5	Securing database access through authentication mechanisms and defining authorization roles using RBAC models to enforce access controls.	**2**	Students will gain proficiency in securing database access through authentication mechanisms and implementing RBAC models to define and enforce granular authorization roles, ensuring data confidentiality and integrity in diverse organizational settings.	4,5
Practical 6	Managing transactions to ensure ACID properties (Atomicity, Consistency, Isolation, Durability) using transaction management techniques and protocols.	**2**	Students will develop expertise in managing transactions to guarantee ACID properties (Atomicity, Consistency, Isolation, Durability), employing robust transaction management techniques and protocols for reliable data operations in database systems.	5,6
Practical 7	Designing and deploying distributed database systems to handle data across multiple nodes or locations while ensuring consistency and fault tolerance.	**2**	Students will learn to design and deploy distributed database systems, ensuring data consistency and fault	3

			tolerance across multiple nodes or locations, crucial for scalable and resilient data management in modern applications.	
Practical 8	Implementing data integrity constraints such as primary keys, foreign keys, and check constraints to maintain data accuracy and reliability.	**2**	Students will gain proficiency in implementing data integrity constraints like primary keys, foreign keys, and check constraints, ensuring data accuracy and reliability in relational database systems.	3
Practical 9	Designing and optimizing database indexes to enhance query performance and facilitate efficient data retrieval.	**2**	Students will master designing and optimizing database indexes to improve query performance and streamline data retrieval operations, enhancing overall database efficiency.	3
Practical 10	Implementing database recovery techniques such as backup and restore procedures to ensure data availability and resilience against failures.	**2**	Students will learn to implement database recovery techniques, including backup and restore procedures, to ensure data availability and resilience against failures, maintaining database integrity and continuity.	3
Practical 11	Exploring object-oriented and object-relational database features for modeling complex data structures and relationships using inheritance and encapsulation.	**2**	Students will explore object-oriented and object-relational database features to effectively model complex data structures and relationships using inheritance and encapsulation, enhancing flexibility and scalability in database design.	3
Practical 12	Utilizing data warehousing techniques to integrate and consolidate data from heterogeneous sources for analytical processing and decision support.	**2**	Students will learn to utilize data warehousing techniques to integrate and consolidate data from diverse sources, enabling effective analytical processing and decision support in organizational contexts.	3
Practical 13	Applying data mining algorithms to extract patterns and insights from large datasets stored in databases, supporting business intelligence and	**2**	Students will apply data mining algorithms to extract valuable patterns and insights from large	3

	predictive analytics.		databases, empowering business intelligence and facilitating predictive analytics for informed decision-making.	
Practical 14	Developing logical database designs to map conceptual models (e.g., ER diagrams) to physical database schemas optimized for performance and storage.	**2**	Students will develop proficiency in translating conceptual models (e.g., ER diagrams) into optimized physical database schemas, ensuring efficient performance and storage in logical database designs.	3
Practical 15	Managing and optimizing database performance by monitoring resource usage, tuning database parameters, and analyzing execution plans for optimization opportunities.	**2**	Students will learn to manage and optimize database performance through effective monitoring of resource usage, tuning database parameters, and analyzing execution plans, ensuring efficient operation and responsiveness in database systems.	3

TEXT BOOKS:

T1: . "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F.Korth, S. Sudarshan, McGraw-Hill.

REFERENCE BOOKS:

R1: "Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.

OTHER LEARNING RESOURCES:

O1: https://www.jbiet.edu.in/coursefiles/cse/HO/cse2/DBMSI-III.pdf

O2: https://cs.ulb.ac.be/public/_media/teaching/infoh303/dbmsnotes.pdf

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental concepts of database management systems with DBMS languages and data models.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
2	Apply the concepts of query languages such as DDL and DML for designing a relational database.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
3	Apply the ACID properties and concurrency control schemes to perform transaction processing with database recovery.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12

4	Apply security to the database by checking for authentication and authorization	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
5	Apply advanced topics of data warehousing and data mining, distributed databases and web databases.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2201R	**Database Management Systems**	**CO1**	2	2	3	2	2				3		3	3
		CO2	3	2	3	2	2				2		3	3
		CO3	3	2	3	3	3				3		3	2
		CO4	3	2	3	1	3				2		3	2
		CO5	3	2	3	3	3				2		2	2

SEMESTER – IV									
Course Title	**Operating Systems**								
Course code	**24BCSM2202R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To learn the mechanisms of OS to handle processes and threads and their communication and the mechanisms involved in memory management in contemporary OS. 2. To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols 3. To know the components and management aspects of concurrency management.								
CO1	Understand operating system evolution, types, services, structures (UNIX, Windows).								
CO2	Analyse process, thread concepts, CPU scheduling, relationships, and states effectively.								
CO3	Evaluate IPC solutions, evaluate deadlock conditions, implement prevention, detection, recovery.								
CO4	Analyse memory management, mapping, allocation, paging, virtual memory, page replacement.								
CO5	Understand I/O hardware, controllers, DMA, secondary storage, disk scheduling, file management.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction:** Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**7**	Students will grasp the concepts of operating systems, including their generations, types, services, system calls, and structural architectures (layered, monolithic, microkernel), as well as understand the concept of virtual machines for efficient resource management and isolation.	**1,2**
II	**Processes:** Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. **Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, **Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.	**8**	Students will understand processes and threads in operating systems, including their definitions, states, relationships, and management mechanisms like context switching and PCBs, alongside exploring process scheduling foundations, criteria, algorithms, and multiprocessor considerations for efficient resource utilization and system responsiveness.	**2,3,4**
III	**Inter-process Communication:**	**8**	Students will master inter-process communication	**2,3,4,5**

	Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution **Deadlocks:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker‘s algorithm, Deadlock detection and Recovery.		concepts including critical sections, race conditions, mutual exclusion, and hardware-based solutions, alongside understanding deadlocks, their conditions, prevention strategies like the Banker's algorithm, and techniques for detection and recovery in operating systems.	
IV	**Memory Management:** Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging. **Virtual Memory**: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).	**8**	Students will comprehend memory management in operating systems, covering concepts like logical and physical address mapping, memory allocation strategies, including contiguous and paging methods, and virtual memory essentials such as locality of reference, demand paging, and page replacement algorithms like FIFO, LRU, and optimal for efficient resource utilization and system performance.	**2,3**
V	**I/O Hardware:** I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure.. **File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management. **Disk Management:** Disk structure, Disk scheduling, Disk reliability, Disk formatting, Boot-block, Bad blocks.	**8**	Students will grasp I/O hardware principles including devices, controllers, and DMA, understand I/O software goals such as interrupt handling and device drivers, explore file management concepts like access methods, types, operations, and directory structures, and learn about disk management covering structure, scheduling, reliability, and formatting for efficient data storage and retrieval in operating systems.	**1,2,3**
Practical Component				
Practical 1	Basic Linux Commands and Overview.	**4**	Gain proficiency in fundamental Linux commands and system navigation.	2,3
Practical 2	Write Shell Script for followings a) To find the global complete path for any file.	**10**	Develop proficiency in shell scripting through practical exercises in file management, system	2,3,5

	b) To broadcast a message to a specified user or a group of users logged on any terminal. c) To copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories. d) To compare identically named files in two different directories and if they are same, copy one of them in a third directory e) To delete zero sized files from a given directory (and all its sub- directories). f) To display the name of those files (in the given directory) which are having multiple links. g) To display the name of all executable files in the given directory. h) Write a script to display the date, time and a welcome message (like Good Morning etc.). The time should be displayed with "a.m." or "p.m." and not in 24 hours notation. i) Write a script to display the directory in the descending order of the size of each file.		administration, and user interaction on Linux systems	
Practical 3	Implementation of FCFS (First Come First Serve) CPU Scheduling.	**2**		3
Practical 4	Implementation of SJF (Shortest Job First) CPU Scheduling.	**2**		3
Practical 5	Implementation of Round Robin (RR) CPU Scheduling.	**2**		3
Practical 6	Implementation of Priority CPU Scheduling Algorithm.	**2**		3
Practical 7	Implementation of FIFO Replacement Algorithm.	**2**		3
Practical 8	Implementation of Optimal Page Replacement Algorithm.	**2**		3
Practical 9	Implementation of LRU Page Replacement Algorithm by Stack method	**2**		3
Practical 10	Implement the producer-consumer problem using threads	**2**		3

TEXT BOOKS:

T1: Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia
Student Edition.

T2: Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

REFERENCE BOOKS:

R1: Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing.

R2: Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley

OTHER LEARNING RESOURCES:

O1: "Operating System Concepts" by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne.

O2: "Modern Operating Systems" by Andrew S. Tanenbaum.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand operating system evolution, types, services, structures (UNIX, Windows).	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Analyse process, thread concepts, CPU scheduling, relationships, and states effectively.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Evaluate IPC solutions, evaluate deadlock conditions, implement prevention, detection, recovery.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Analyse memory management, mapping, allocation, paging, virtual memory, page replacement.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Understand I/O hardware, controllers, DMA, secondary storage, disk scheduling, file management.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2202R	**Operating Systems**	**CO 1**	3	2	2	3	2					3		2
		CO 2	3	2	2	2	2					2		1
		CO 3	3	3	3	3	2					3		1
		CO 4	2	2	3	2	3					2		2
		CO 5	2	2	2	3	2					3		1

SEMESTER – IV									
Course Title	**Probability and Statistics**								
Course code	**24BCSM2203R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T	**2**	**1**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Mathematics**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To make understand to solve first and second order ordinary differential equation. 2. To provide conceptual understanding on higher order ODE with constant coefficients. 3. To provide an overview of probability to engineers. 4. To understand the concept of basic statistical measurements. 5. To analyse the data by applying statistical methods.								
CO1	Enabling solving skills of ordinary differential equations of various order.								
CO2	Enabling solving skills of higher order ODE with constants coefficients.								
CO3	Enable to formulate and solve problems involving random variables.								
CO4	Enable to solve problems of various probability distribution.								
CO5	Analysing experimental data by applying statistical methods.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**First order ordinary differential equations:** Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.	**7**	Understanding first-order ODEs: exact, linear, Bernoulli's, Euler's equations, and equations solvable for p, y, x, and Clairaut's type.	**1,2**
II	**Ordinary differential equations of higher orders:** Second order linear differential equations with variable coefficients, method of variation of parameters, Cauchy-Euler equation, Higher order ODE with constant coefficients.	**9**	Understanding higher-order ODEs: second-order linear equations with variable coefficients, variation of parameters, Cauchy-Euler equation, and constant coefficient ODEs.	**2,3**
III	**Probability:** Basic definitions on Probability, conditional probability, independence; Discrete random variables, Independent random variables., Probability distributions: Binomial, Poisson and Normal, Poisson approximation to the binomial distribution.	**8**	Understanding probability basics: definitions, conditional probability, independence, discrete random variables, and distributions (Binomial, Poisson, Normal) including Poisson approximation.	**2,3,5**
IV	**Basic Statistics:** Measures of central tendency: Moments, skewness and Kurtosis, Correlation and regression.	**8**	Mastering basic statistics: measures of central tendency, moments, skewness, kurtosis, and understanding correlation and regression for data analysis.	**2,3,4**
V	**Applied Statistics:**	**8**	Understanding applied statistics: significance testing for large samples,	**2,3**

	Test of significance: Large sample test for single proportion. Difference of proportions, single mean, difference of means and difference of standard deviations.		including tests for single proportion, difference of proportions, means, and standard deviations.	

TEXT BOOKS:

T1: Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
T2: N. P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.

REFERENCE BOOKS:

R1: S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.
R2: S. L. Ross, Differential Equations, 3rd Ed. Wiley India, 1984.

OTHER LEARNING RESOURCES:
O1: tutorialspoint- offers a course "Statistics-Probability" for all beginners
O2: youtube- Statistics and Probability full course.
O3: NPTEL- a video lecture on "Probability and Statistics". Course by Prof. Somesh Kumar, IIT Kharagpur.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enabling solving skills of ordinary differential equations of various order.	PO1, PO2, PO5, PO12
2	Enabling solving skills of higher order ODE with constants coefficients.	PO1, PO2, PO5, PO12
3	Enable to formulate and solve problems involving random variables.	PO1, PO2, PO4, PO6, PO12
4	Enable to solve problems of various probability distribution.	PO1, PO4, PO5, PO10, PO12
5	Analysing experimental data by applying statistical methods.	PO2, PO4, PO5, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**

24BCSM2203R	**Probability and Statistics**	**CO1**	3	3			2							1
		CO2	3	3			2							2
		CO3	2	2		2		1						1
		CO4	2			2	2					1		1
		CO5		3		3	2						1	2

SEMESTER – IV									
Course Title	**Formal Language and Automata Theory**								
Course code	**24BCSM2204R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To gain a solid understanding of formal languages, grammars, automata, and their theoretical foundations in computer science. 2. Learn to design, analyse, and compare different computational models such as finite automata, pushdown automata, and Turing machines. 3. Develop skills to apply formal methods and automata theory concepts to solve complex problems in language processing, compiler design, and algorithm development.								
CO1	Understand the fundamental characteristics of formal languages and formal grammars, showcasing foundational linguistic knowledge.								
CO2	Recognize the similarity between deterministic and non-deterministic finite automata, demonstrating comparative analysis skills.								
CO3	Evaluate the minimization processes of both deterministic and non-deterministic finite automata, applying critical thinking skills.								
CO4	Analyse the resemblance between non-deterministic push-down automata and context-free grammars, demonstrating advanced understanding and synthesis of concepts.								
CO5	Examine the fundamental characteristics of Turing machines and their computational applications, showcasing analytical skills in computing theory.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction** Definitions: Language, Grammar, Automata, Relation between language, Grammar and automata, Importance of automata theory. **Finite Automata (FA)** Introduction of FA and its example. DFA: Definition, Processing strings, Transition functions, Language of a DFA, NDFA: Non-determinism, Definition, Processing strings, Extended transition functions, Language of a NFA, Kleene's theorem, Epsilon transitions, Applications of FA in text search.	**8**	To grasp finite automata theory, design deterministic/nondeterministic automata, understand regular languages, and apply these concepts to problem-solving.	**1, 2,3**
II	**Regular expressions and regular languages**	**8**	learn to construct and interpret regular expressions, understand regular languages, and apply	**2,3**

	Memory required to recognize a language, Regular expressions, Regular expression to finite automata & vice versa, Algebraic laws for regular expressions, Applications of regular expressions, Criterion for regularity, Regular languages, Properties of Regular languages		these concepts to pattern matching.	
III	**Context Free Grammars and Languages** Definition, Left most and right most grammars, Parse trees, Ambiguity: Ambiguous grammar, Removing ambiguity, Normal forms, Applications of context free grammars: Parsers	**10**	To comprehend context-free grammars, parse trees, Chomsky hierarchy, and apply these concepts to analyze and generate context-free languages.	**2,4**
IV	**Pushdown automata (PDA) and context free languages (CFL)** Definition & representation of pushdown automata, Acceptance by PDA: By final state, By empty stack, Deterministic PDA, Equivalence of PDA and CFL, Pumping Lemma for CFL Closure properties of CFL, Testing membership of context free, Decision problems for CFLs.	**10**	To master pushdown automata theory, understand context-free languages, parse trees, and apply these concepts to language recognition and parsing.	**2,3**
V	**Turing machines** Definition, Language of a Turing machine, Programming Turing machines, The Church-Turing thesis, A simple programming language, Extensions of the basic Turing machine. **Recursively enumerable languages & Undesirability** The halting problem, The post correspondence problem, Time and space complexity of Turing machines, Complexity classes	**9**	To grasp Turing machine theory, recursively enumerable languages, undecidability, and apply these concepts to analyze computability and algorithmic problems.	**2,3,4**

Text Books:

T1: John. E. Hopcroft, Rajeev Motwani, Jeffry Ullman,‖ Introduction to Automata Theory, Languages and Computation‖, Pearson Education.
T2: John Martin,—Introduction to Languages and the Theory of Computation, Tata McGraw Hill

Reference Books:

R1: Adesh K. Pandey,―An Introduction to Automata Theory & Formal Languages‖, Katson Books.
R2: Peter Linz,―An Introduction to Formal Languages and Automata‖, Narosa.
R3: Zvi Kohavi and Niraj K. Jha,―Switching and Finite Automata Theory‖, Tata McGraw Hill.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental characteristics of formal languages and formal grammars, showcasing foundational linguistic knowledge.	CO1, CO2, CO3, CO10, CO12
2	Recognize the similarity between deterministic and non-deterministic finite automata, demonstrating comparative analysis skills.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Evaluate the minimization processes of both deterministic and non-deterministic finite automata, applying critical thinking skills.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Analyze the resemblance between non-deterministic push-down automata and context-free grammars, demonstrating advanced understanding and synthesis of concepts.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
5	Examine the fundamental characteristics of Turing machines and their computational applications, showcasing analytical skills in computing theory. .	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM2204R	Formal Language and Automata Theory	CO 1	2	2	1							1		1
		CO 2	3	3	2	2	1					1		2
		CO 3	3	3	3	1	1					1		2
		CO 4	3	3	2	2	1					1		2
		CO 5	3	3	2	2	1					1		2

SEMESTER IV									
Course Title	**Quantitative Aptitude For Computer Science**								
Course code	**24BCSM2206R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours:30T	**1**	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives	1. To develop the mathematical foundation required for solving quantitative problems in computer science. 2. To enhance logical reasoning and analytical skills for computational thinking. 3. To introduce the concepts of data interpretation and statistical analysis for decision-making. 4. To familiarize students with combinatorial and probability techniques relevant to algorithm design. 5. To understand optimization methods and computational models for real-world applications in computer science.								
CO1	Apply mathematical principles and logic to solve quantitative problems in computer science.								
CO2	Analyze and interpret data effectively for informed decision-making in technical scenarios.								
CO3	Demonstrate problem-solving skills using combinatorial techniques and probability models.								
CO4	Develop optimization solutions for computational problems using mathematical tools.								
CO5	Evaluate and analyze the complexity of algorithms and systems using appropriate models.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Mathematical Foundations** - Number Theory: Divisibility, Modular Arithmetic, GCD, LCM - Linear Algebra: Matrices, Determinants, Eigenvalues, Eigenvectors - Permutations and Combinations: Principles of Counting, Probability Basics	**6**	Understand basic mathematical concepts and their applications in computational problems.	**1,2,3**
II	**Logical Reasoning and Problem Solving** - Logical Deductions, Syllogisms - Puzzles and Series: Number, Letter, and Symbol Series - Coding-Decoding, Blood Relations	**6**	Develop logical thinking and problem-solving skills essential for algorithm design.	**3,4**
III	**Data Interpretation and Analysis** - Tabular Data: Representation and Analysis - Graphs and Charts: Bar, Line, Pie - Basic Statistical Measures: Mean, Median, Mode, Standard Deviation	**6**	Analyse and interpret data efficiently for decision-making and pattern recognition.	**4,5**
IV	Combinatorics and Probability - Advanced Probability: Bayes' Theorem, Conditional Probability - Binomial Theorem, Principle of Inclusion-Exclusion - Applications in Algorithm Analysis	**6**	Apply combinatorial techniques and probability concepts in computational problems.	**4,5**

V	Optimization and Computational Models - Optimization: Linear Programming, Graph Theory Basics - Time and Space Complexity: Big O, Big Theta, Big Omega - Applications in Computer Science (e.g., Scheduling, Search, Sort Algorithms)	**6**	Understand optimization techniques and computational complexity for problem-solving.	**4,5**

TEXT BOOKS:

T1: Greene, D. E. - "Mathematical Structures for Computer Science: A Modern Treatment of Discrete Mathematics", W. H. Freeman, 7th Edition, 2014.

T2: Rosen, K. H. - "Discrete Mathematics and Its Applications", McGraw-Hill, 7th Edition, 2011.

T3: Babu, R. S. - "Quantitative Aptitude for Competitive Examinations", S. Chand Publishing, 2018.

REFERENCE BOOKS:

R1: Mazumdar, S. - "Probability, Statistics, and Random Processes for Engineers", Pearson, 4th Edition, 2015.

R2 : Papadimitriou, C. H., & Steiglitz, K. - "Combinatorial Optimization: Algorithms and Complexity", Dover Publications, 1998.

OTHER LEARNING RESOURCES:

O1: NPTEL Lectures:

Discrete Mathematics for Computer Science by Prof. Kamala Krithivasan, IIT Madras.

Probability and Statistics by Prof. Prathap Haridoss, IIT Madras.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply mathematical principles and logic to solve quantitative problems in computer science.	PO1, PO5, PO12
2	Analyze and interpret data effectively for informed decision-making in technical scenarios.	PO1, PO5, PO12
3	Demonstrate problem-solving skills using combinatorial techniques and probability models.	PO1, PO5, PO12
4	Develop optimization solutions for computational problems using mathematical tools.	PO1, PO5, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2206R	**Quantitative Aptitude For Computer Science**	**CO1**	3				3							2
		CO2	2				2							1
		CO3	2				2							1
		CO4	2				2							1
		CO5												

SEMESTER – IV									
Course Title	**Digital Literacy**								
Course code	**24UCDL2202R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**All Degree / Diploma programme of Engineering, Computer Technology, Management, Humanities, Science, and Agricultural Sciences**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. Students will be able to understand the fundamentals of computer systems and Internet search along with advanced features of MS-Office. 2. Students will be able to understand about the introduction to Social Media and E-Commerce and utility software. 3. Students will develop a solid foundation in computational thinking, which includes problem-solving, algorithmic design, and logical reasoning.								
CO1	Fundamentals of Computer Systems, Office Automation and Internet Search.								
CO2	Know more about the Internet & Cyber World								
CO3	Know Social Media, E-Commerce, and apply the same for digital branding								
CO4	Know to use the digital payments and digital transactions, and other utility software								
CO5	Able to build basic level websites								

Practical Component

Unit-No.	Content	Contact Hour	Learning Outcome	BL
Practical 1	Disassemble and reassemble a desktop computer, identifying and explaining the function of each component. Discuss the importance of hardware compatibility.	**3**	By disassembling and reassembling a desktop computer, learners understand component functions, emphasize hardware compatibility for system stability, and gain practical troubleshooting skills.	3
Practical 2	Install an operating system (e.g., Windows or Linux) on a virtual machine. Configure essential settings, such as user accounts, network connections, and system updates.	**3**	Creating and configuring a virtual machine with an operating system develops skills in installation, setup of user accounts, network configurations, and system maintenance, enhancing proficiency in virtualized environments.	3
Practical 3	Use office software (e.g., Microsoft Office or Google Workspace) to create documents, spreadsheets, and presentations. Teach formatting, inserting images, and collaboration features.	**3**	Using office software for document creation, formatting, image insertion, and collaboration enhances proficiency in productivity tools, fostering effective communication and collaborative skills in professional settings.	3
Practical 4	Instruct students on effective internet searching, including the use of search	**3**	Teaching effective internet searching covers using	3

	engines, keywords, and advanced search operators.		search engines proficiently, employing appropriate keywords, and leveraging advanced search operators to refine results, enhancing research skills and information retrieval efficiency.	
Practical 5	Set up email accounts, compose and send emails, attach files, and organize emails into folders. Discuss email etiquette and best practices.	**3**	Setting up email accounts, composing, sending emails, attaching files, organizing with folders, and understanding email etiquette enhances communication efficiency and professionalism, emphasizing clarity, conciseness, and proper etiquette in digital correspondence.	3
Practical 6	Use cloud storage services (e.g., Google Drive or Dropbox) to store and synchronize files. Create a backup of important data and discuss data recovery options.	**3**	Using cloud storage services for file storage and synchronization ensures accessibility and collaboration, while creating backups of important data safeguards against loss, with data recovery options providing security and continuity in digital operations.	3
Practical 7	Instruct students to create accounts on popular social media platforms (e.g., Facebook, Twitter, Instagram). Guide them through profile setup, privacy settings, and content posting.	**3**	Guiding students to create social media accounts, set up profiles, configure privacy settings, and post content fosters digital presence management, emphasizing responsible sharing, online safety, and personal branding awareness.	3
Practical 8	Have students create a content calendar for a fictional business or brand. Plan posts, including text, images, and hashtags, and use social media management tools to schedule posts.	**3**	Creating a content calendar for a fictional business includes planning posts with text, images, and hashtags, using social media management tools to schedule and optimize engagement, fostering skills in strategic content planning and digital marketing execution.	3
Practical 9	Guide students in building a basic e-commerce website using platforms like Shopify or WooCommerce. They should add products, set up	**3**	Guiding students to build a basic e-commerce website involves adding products, setting up payment gateways, and configuring	3

	payment gateways, and configure the online store.		the online store using platforms like Shopify or WooCommerce, fostering skills in online retail setup and management for digital entrepreneurship.	
Practical 10	Set up a payment gateway for the e-commerce website created in Experiment 3. Test payment transactions and discuss security protocols such as SSL encryption.	**3**	Setting up a payment gateway for the e-commerce website involves integrating services like PayPal, Stripe, or others, testing transactions to ensure functionality, and emphasizing security protocols such as SSL encryption to protect customer data during online transactions, ensuring safe and secure e-commerce operations.	3

TEXT BOOKS:

T1: Sinha, Pradeep K. and Priti Sinha. *Computer Fundamentals: Concepts Systems Applications*. 3rd ed. New Delhi: BPB Publications.

T2: Goel, A. 2010. *Computer Fundamentals*. Pearson India.

REFERENCE BOOKS:

R1: Balaguruswamy, E. 2009. *Fundamentals of Computers*. Tata McGraw-Hill Education.

R2: Balaguruswamy, E. 2014. *Fundamentals of Computer and Programming (Updated Ed Sem. I, Au)*. Tata McGraw-Hill Education.

OTHER LEARNING RESOURCES:

O1: https://www.w3schools.com

O2: https://edu.gcfglobal.org

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will have basic knowledge of cooking methods.	PO1, PO5, PO12
2	Students will gain the knowledge of organizing & Cleaning of Rooms.	PO1, PO5, PO12

3	Students will be able to gain the travel management concept.	PO1, PO5, PO12
4	Students will be able to acquire the knowledge of basic household's amenities for day- to-day use.	PO1, PO5, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24UCDL1002R	Digital Literacy	CO 1	3				3							2
		CO 2	2				2							1
		CO 3	2				2							1
		CO 4	2				2							1
		CO 5												

SEMESTER – IV									
Course Title	**Basic Acclimatizing Skills**								
Course code	**24UULS2203R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 52	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**All the Under Graduate Programmes**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To impart knowledge of the fundamentals of Hospitality industry and its applications. 2. Students will be able to familiarize with the cooking equipment's & Utensils. 3. Students will be able to handle different modes of reservations.								
CO1	Students will have basic knowledge of cooking methods.								
CO2	Students will gain the knowledge of organizing & Cleaning of Rooms.								

CO3	Students will be able to gain the travel management concept.			
CO4	Students will be able to acquire the knowledge of basic household's amenities for day- to-day use.			
CO5	Students will be able to gain the hospitality management concept.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Accommodation Management** Telephone handling technique Organizing of Rooms. Cleaning equipment's and uses. Bed making Process.	**12**	Introduction to Accommodation Management covers telephone handling techniques, room organization, cleaning equipment usage, and bed making processes, ensuring comprehensive skills in guest service and room maintenance.	**1,2**
II	**Fundamental of Cooking** Uses of basic cooking equipment's Uses of fire & Fuel Different cuts of vegetables Uses of herbs & spices Regional Food Habits	**10**	Fundamentals of Cooking include mastering basic cooking equipment, understanding fire and fuel usage, learning various vegetable cuts, utilizing herbs and spices effectively, and exploring regional food habits for diverse culinary knowledge.	**2,3,4**
III	**Food and Beverage skills** Introduction to catering industry Types menus and beverages Identifications of Cutlery, crockery & glassware Table etiquettes or manners Customer handling skills or Situation Handling	**12**	Food and Beverage skills encompass an introduction to the catering industry, knowledge of menu and beverage types, identification and use of cutlery, crockery, and glassware, understanding table etiquette, and developing customer handling and situational management abilities.	**2,3,5**
IV	**Travel management** Travel Documentation (Types) Application of passport & Visa Tourism products (UNESCO sites) Types of logistics in travel and tourism management	**10**	Travel Management involves understanding travel documentation types, applying passport and visa procedures, exploring tourism products such as UNESCO sites, and managing logistics in travel and tourism operations effectively.	**2,3**
V	**Basic Hospitality Skills** Various Egg Preparations Canapés preparations Mock tail & Shakes Preparations Butter Rice / Lemon Rice	**8**	Basic Hospitality Skills include mastering various egg preparations, creating canapés, preparing mocktails and shakes,	**1,2,3**

	Various Lentils Preparations 1 non-veg preparation/ 1 veg preparation		making butter rice or lemon rice, mastering various lentil dishes, and preparing one non-vegetarian and one vegetarian dish to ensure a diverse culinary repertoire.	

TEXT BOOKS:

T1: Arora K (2011). Theory of cookery, Frank brothers & company (pub) pvt.ltd.-New Delhi.
T2: Bruce H. Axler, Carol A. Litrides (2010) Food and Beverage Service Volume 1 of Wiley Professional Restauranteur, Guides.

REFERENCE BOOKS:

R1: Mohammed Zulfikar (2010) - Introductions to Tourism and Hotel Industry Introduction to Tourism and Hotel Industry. Vikas Publishing.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Students will have basic knowledge of cooking methods.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
2	Students will gain the knowledge of organizing & Cleaning of Rooms.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
3	Students will be able to gain the travel management concept.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
4	Students will be able to acquire the knowledge of basic household's amenities for day- to-day use.	PO1, PO2, PO3, PO4, PO7, PO10,PO12
5	Students will be able to gain the hospitality management concept.	PO1, PO2, PO3, PO4, PO7, PO10,PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24UULS2201R	**Basic Acclimati**	**CO1**	1	1	2	2			3			1		2
		CO2	1	1	2	2			3			1		2

	zing Skills	**CO3**	1	2	3	2			3			1		2
		CO4	2	3	2	3			3			1		2
		CO5	3	3	3	3			3			1		2

SEMESTER – IV									
Course Title	**Data Visualization**								
Course code	**24BCSM2205R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Knowledge of Data Handling and Programming	**Co-requisite**	Basic programming skills (e.g., Python)						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To introduce the fundamental principles and techniques of data visualization. 2. To enable students to preprocess and transform raw data for visualization. 3. To teach the creation and customization of basic and advanced visualizations. 4. To develop skills for building interactive dashboards and geospatial visualizations. 5. To foster the ability to tell impactful stories using data visualizations.								
CO1	Understand the fundamentals and tools of data visualization.								
CO2	Preprocess and clean data for effective visualization.								
CO3	Create and customize various types of charts and advanced visualizations.								
CO4	Develop interactive dashboards and geospatial visualizations.								
CO5	Communicate insights effectively through storytelling with data.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	• Basics of Data Visualization: Importance, Types, and Applications • Visualization Process: From Data to Insights • Principles of Effective Data Visualization: Clarity, Accuracy, and Aesthetics • Tools for Data Visualization: Tableau, Power BI, Matplotlib, Seaborn	7	Understand the fundamentals of data visualization and identify appropriate tools and techniques for visualizing data.	1,2,3
II	• Data Cleaning and Transformation for Visualization • Handling Missing Data and Outliers • Types of Data: Categorical, Numerical, and Time-Series • Aggregating and Summarizing Data	8	Prepare and preprocess data to make it suitable for creating effective visualizations.	3,4
III	• Basic Charts: Bar Chart, Pie Chart, Line Chart, and Scatter Plot • Customizing Charts: Titles, Labels, Colors, and Legends • Comparing Data: Grouped Bar Charts, Stacked Bar Charts, and Multi-Line Charts • Best Practices for Chart Selection	8	Create and customize basic charts to effectively represent different types of data.	3,4
IV	• Advanced Charts: Heatmaps, Box Plots, Histograms, and Tree Maps • Geospatial Data Visualization: Maps and Geographic Charts	8	Develop advanced visualizations and interactive dashboards for comprehensive data analysis.	4,5

	• Creating Dashboards: Combining Multiple Visualizations • Adding Interactivity to Visualizations			
V	• Designing Visual Narratives: Storyboarding and Wireframing • Telling Stories with Visuals: Choosing the Right Visualization • Case Studies: Analyzing Real-World Data Visualization Examples • Common Pitfalls and How to Avoid Them	8	Create compelling stories using data visualizations to communicate insights effectively.	5,6
Practical Component				
Practical 1	Explore and compare features of tools like Tableau, Power BI, Matplotlib, and Seaborn by creating simple visualizations.	2	Familiarize with different data visualization tools and understand their basic functionalities.	2,3
Practical 2	Perform data cleaning and transformation using Python (Pandas) or Excel. Handle missing values, outliers, and normalize data.	2	Preprocess raw data to make it suitable for visualization.	3,4
Practical 3	Create bar charts, line charts, pie charts, and scatter plots using Matplotlib/Seaborn for a given dataset.	6	Develop and customize basic visualizations to represent different types of data.	3
Practical 4	Enhance basic charts by adding titles, labels, legends, and colors. Create grouped and stacked bar charts using Python.	3	Learn to customize visualizations for better clarity and presentation.	3,4
Practical 5	Develop heatmaps, histograms, and box plots using Seaborn and interpret the insights gained from these visualizations.	3	Create and interpret advanced visualizations for data analysis.	3
Practical 6	Visualize geospatial data on maps using Tableau, Power BI, or Python (Plotly). Mark regions and analyze geographic trends.	3	Represent and analyze spatial data using geographic visualizations.	4,5
Practical 7	Design an interactive dashboard in Tableau or Power BI by combining multiple charts and adding filters.	3	Develop dashboards to present comprehensive insights interactively.	4,5
Practical 8	Choose a dataset, identify key insights, and create a storyboard using a sequence of visualizations to communicate findings.	3	Use data visualizations to tell compelling and impactful stories.	5,6
Practical 9	Visualize and analyze time-series data using line charts, area charts, and annotations for trends and patterns.	3	Analyze temporal patterns in data using appropriate visualizations.	4,5
Practical 10	Create visualizations that illustrate common pitfalls (e.g., misleading axes, overloading visuals) and fix them for clarity.	2	Identify and resolve issues in data visualizations to ensure clarity and accuracy.	4,5

Textbooks

T1: Tamara Munzner - "Visualization Analysis and Design", CRC Press, 2014.

T2: Cole Nussbaumer Knaflic - "Storytelling with Data: A Data Visualization Guide for Business Professionals", Wiley, 2015.

Reference Books

R1: Ben Fry - "Visualizing Data", O'Reilly Media, 2008.

R2: Nathan Yau - "Data Points: Visualization That Means Something", Wiley, 2013.

Other Learning Resources

O1: "Data Visualization with Python" (Coursera by IBM).

O2: Tableau for Beginners" (Udemy).

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Recognize the characteristics of machine learning strategies	PO1, PO2, PO3, PO4, PO10, PO12
2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations	PO1, PO2, PO3, PO4, PO10, PO12
3	Analyze and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Understand the concept of ensemble learning and its applications	PO1, PO2, PO3, PO4, PO10, PO12
5	Evaluate the performance of various algorithms	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
24BCSM2205R		**CO1**	2	2	1	3						1		2

	Data Visualization Learning	**CO2**	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – IV									
Course Title	**Mini Project II**								
Course code	**24BCSM2207R**	**Total credits:1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 48	**0**	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the second year of the Programme**								
Course Objectives (Minimum 3)	1. To develop the knowledge, skills and attitudes of a professional Computer engineering professional. 2. To become confident in designing engineering solutions to complex software problems utilising a systems approach. 3. Apply theoretical knowledge gained in earlier semesters to solve real-world problems through the development and implementation of a software project.								
CO1	Apply IT principles for real-world problem-solving using programming languages and network solutions.								
CO2	Contribute to complex projects as a team member, demonstrating effective personal and team management.								
CO3	Develop programming skills to create and assess software, hardware, and network solutions.								
CO4	Apply personal and team management skills as a professional software developer.								
CO5	Cultivate employability skills and uphold professionalism in software development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a) theoretical studies b) computer simulations c) hardware construction 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 4. Deliver a seminar on the general area of work being undertaken	48	This course involves individual or small group projects supervised by academic staff, focusing on topics aligned with student and supervisor interests. Students will conduct extensive literature reviews to understand current developments in their chosen technical area. They will then engage in detailed technical work, employing theoretical studies, computer simulations, or hardware construction. Throughout the project, students will document their progress through regular reports or a professional journal, ensuring accountability and effective time management. They will present their findings and contributions in a seminar, preparing a formal report that outlines the project's scope, methodologies, and results. The culmination will involve showcasing	1,2,3,4,5

	and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.		their work in a public forum, utilizing poster presentations and operational demonstrations of hardware and software. This comprehensive approach aims to enhance students' research, technical, communication, and presentation skills in a real-world project environment.	

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM2207R	**Mini Project-II**	**CO 1**	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

SEMESTER – V									
Course Title	**Data Communication and Computer Network**								
Course code	**24BCSM3101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Understand the fundamentals of data communications principles of media access, switching, routing and flow control. 2. Understand the basics of network protocol design and analysis. 3. Be familiar with the TCP/IP protocol suite and with application layer protocols. 4. Be able to write network-capable programs using the socket libraries. 5. Understand the structure of client-server systems and be able to build client-server programs.								
CO1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.								
CO2	Analyze data link layer techniques, flow control, and error protocols.								
CO3	Analyse network layer protocols along with routing issues.								
CO4	Summarize transport and application layer operations and protocols along with QoS services.								
CO5	Design and execute computer network programming projects, showcasing advanced skills in network application development.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Introduction to networks, internet, protocols and standards, the OSI model, layers in OSI model, TCP/IP suite, Addressing, Analog and digital signals. Physical Layer: digital transmission, multiplexing, transmission media, circuit switched networks, Datagram networks, virtual circuit networks, switch and Telephone network.	**8**	Understand the OSI model, TCP/IP suite, and basic concepts of addressing, signals, and transmission media.	**2**
II	Data link layer: Introduction, Block coding, cyclic codes, checksum, framing, flow and error control, Noiseless channels, noisy channels, HDLC, point to point protocols	**8**	Learn error detection and correction techniques, framing, and protocols like HDLC and point-to-point protocols.	**2,3**
III	Network Layer: Logical addressing, internetworking, tunnelling, address mapping, ICMP, IGMP, forwarding, uni-cast routing protocols, multicast routing protocols.	**10**	Gain knowledge on logical addressing, routing protocols, and address mapping techniques.	**2,4**
IV	Transport Layer: Process to process delivery, UDP and TCP protocols, SCTP, data traffic, congestion, congestion control, QoS, integrated services, differentiated services, QoS in switched networks.	**10**	Comprehend the fundamentals of process-to-process delivery, TCP/UDP protocols, and congestion control mechanisms.	**2,3**
V	Application Layer – Domain name space, DNS in internet, electronic	**9**	Understand domain name space, DNS operations, email	**2**

	mail, FTP, WWW, HTTP, SNMP, multi-media, network security		protocols, FTP, HTTP, and basics of network security.	
Practical Component				
Practical 1	Configure a simple network consisting of multiple devices such as routers, switches, and computers. Test connectivity and troubleshoot any issues encountered.	2	Demonstrate the ability to configure basic network settings and troubleshoot connectivity problems.	1,2,3
Practical 2	Use network analysis tools like Wireshark to capture and analyze network traffic. Identify different protocols, their headers, and their roles in the OSI model.	2	Gain proficiency in protocol analysis and understanding of the OSI model layers.	3,4
Practical 3	Use oscilloscopes or signal analysers to examine analog and digital signals transmitted over the network. Interpret signal characteristics and identify any anomalies.	2	Understand the fundamentals of analog and digital signals in networking and their analysis	3,4,5
Practical 4	Implement a block coding technique such as Hamming codes to detect and correct errors in transmitted data frames. Analyse its effectiveness in error detection and correction.	2	Gain hands-on experience in implementing error detection and correction mechanisms in the data link layer.	3,4,5
Practical 5	Implement CRC error detection mechanism in a data link layer protocol such as Ethernet. Calculate CRC values for given data frames and verify error detection capability.	2	Understand the theory and practical implementation of CRC for error detection.	3,4,5
Practical 6	Implement and analyze flow control mechanisms such as Stop-and-Wait and Sliding Window protocols. Measure throughput and efficiency under different network conditions.	2	Understand the concepts of flow control and error control in data link layer protocols.	3,4,5
Practical 7	Configure IP addresses using both IPv4 and IPv6 addressing schemes for a network topology. Verify connectivity and troubleshoot addressing conflicts.	2	Demonstrate proficiency in configuring logical addresses andunderstanding IP addressing schemes.	3,4,5
Practical 8	Implement and compare different routing protocols such as RIP, OSPF, and BGP in a simulated network environment. Analyze routing tables and convergence times.	2	Understand the operation of routing protocols and their impact on network performance.	3,4,5
Practical 9	Configure Address Resolution Protocol (ARP) and Neighbor Discovery Protocol (NDP) for mapping network layer addresses to data link layer addresses. Verify mappings and troubleshoot resolution issues.	2	Gain practical experience in address resolution and mapping techniques.	3,4,5
Practical 10	Capture and analyse UDP and TCP traffic using packet analysis tools. Compare their characteristics, such	2	Understand the differences between UDP and TCP protocols and their suitability for different applications.	3,4,5

	as connection establishment, reliability, and overhead.			
Practical 11	Configure QoS mechanisms such as traffic prioritization, traffic shaping, and congestion management in a network. Measure and analyze QoS metrics under different traffic loads.	2	Understand the importance of QoS in ensuring reliable and efficient network performance.	3,4,5
Practical 12	Implement and evaluate congestion control algorithms such as TCP's congestion avoidance and congestion control mechanisms. Measure throughput and packet loss under congested conditions.	2	Gain practical insights into congestion control mechanisms and their impact on network performance.	3,4,5
Practical 13	Configure and troubleshoot DNS servers and clients. Analyze DNS queries and responses using network diagnostic tools.	2	Understand the operation of DNS and its role in translating domain names to IP addresses.	4,5,6
Practical 14	Set up email servers (SMTP, IMAP) and FTP servers. Demonstrate the exchange of emails and file transfers between clients and servers. Implement security measures such as encryption and authentication.	2	Gain practical experience in configuring and securing email and FTP services.	4,5,6
Practical 15	Deploy web servers (HTTP, HTTPS) and web applications. Implement security measures such as SSL/TLS encryption, HTTPS, and web application firewalls to protect against common attacks.	2	Understand the principles of web service security and implement security measures to protect against threats.	4,5,6

TEXT BOOKS:

T1: Data Communications and Networking–Behrouz A. Forouzan, Fourth EditionTMH,2006.

T2: Computer Networks—Andrew S Tanenbaum, 4th Edition, Pearson Education.

REFERENCE BOOKS:

R1: An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education

R2: UnderstandingcommunicationsandNetworks,3rd Edition, W.A. Shay, Cengage Learning.

R3: Computer and Communication Networks, Nader F. Mir, Pearson Education

R4: Computer Networking: A Top-Down Approach Featuring the Internet, James F. Kurose, K. W. Ross,3rd Edition, Pearson Education.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**

1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.	PO1, PO2, PO3, PO4, PO5, PO12
2	Analyze data link layer techniques, flow control, and error protocols.	PO1, PO2, PO3, PO4, PO5, PO12
3	Analyse network layer protocols along with routing issues.	PO1, PO2, PO3, PO4, PO5, PO12
4	Summarize transport and application layer operations and protocols along with QoS services.	PO1, PO2, PO3, PO4, PO5, PO12
5	Design and execute computer network programming projects, showcasing advanced skills in network application development.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM3101R	**Data Communication and Computer Network**	**CO1**	3	1	2	1	1							3
		CO2	3	2	3	1	1							3
		CO3	3	2	3	1	1							3
		CO4	3	2	3	1	2							3
		CO5	3	3	3	3	2					1		3

SEMESTER – V									
Course Title	**Signals and Systems**								
Course code	**24BCSM3102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Introduces students for an intermediate level of fluency with signals and systems in both continuous time and discrete time. 2. Prepares students for more advanced subjects in digital signal processing (including audio, image and video processing). 3. Prepares students in communication theory, and system theory, control and robotics.								
CO1	Understand different types of signals-continuous and discrete, odd and even, periodic and aperiodic etc.								
CO2	Understand Classifications of the standard forms of Signals with respect to systems based on their properties.								
CO3	Analyse the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.								
CO4	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.								
CO5	Analyzation of Z-transform in continuous time signals and systems by using appropriate mathematical tools.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Signal and System:** signal and system as seen in everyday life, signal properties, periodicity, determinism and stochastic characteristics, special signal, system properties.	**7**	To Explain different types of signals- continuous and discrete, odd and even, periodic and aperiodic etc.	1,2
II	**Behaviour of continuous and discrete time LTI system:** Impulse response, step response, convolution, input-output behaviour, Characterization of causality and stability of LTI system.	**8**	Classifications of the standard forms of Signals with respect to systems based on their properties	1,2
III	**Laplace Transform:** Laplace Transform and its inverse: Definition, existence conditions, Region of Convergence and properties, Application of Laplace transform for the analysis of continuous time LTI system (stability etc.) Significance of poles & zeros.	**10**	Analysing the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.	1,2
IV	**Fourier Analysis of Continuous and discrete Time Signals and Systems:** Fourier Series, Fourier Transform and properties, Parseval‟s theorem, Frequency response of LTI systems. Sampling Theorem.	**12**	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.	1,2
V	**Z-Transform:** Z-Transform and its inverse: Definition, Existence Region of convergence and properties. Application of Z-Transform for the analysis of	**8**	Analyzation of Z-transform in continuous time signals and systems by using	1,2

	Discrete time LTI systems, Significance of poles and zeros.		appropriate mathematical tools.	
Practical Component				
Practical 1	**Introduction to Signals and Their Properties:**Generate and analyze basic signals (e.g., sinusoidal, exponential, and unit step signals) using MATLAB/Python. Study their properties such as periodicity, determinism, and energy/power.	**3**	Understand and classify different types of signals and their properties.	**1,2,3**
Practical 2	**Impulse and Step Response of LTI Systems:** Determine the impulse and step response of a continuous-time LTI system using MATLAB/Python.	**3**	Analyze the behavior of LTI systems using impulse and step responses.	**3,4**
Practical 3	**Convolution of Signals:** Perform convolution of two continuous-time or discrete-time signals using MATLAB/Python to compute the system's output.	**3**	Understand and apply the convolution operation to analyze input-output relationships in LTI systems.	**3,4**
Practical 4	**Stability and Causality of LTI Systems:** Verify the stability and causality of an LTI system using its impulse response and step response.	**3**	Determine the stability and causality of LTI systems based on input-output characteristics.	**3,4**
Practical 5	**Laplace Transform and System Analysis:** Compute the Laplace transform of continuous-time signals and analyze the stability of an LTI system using poles and zeros.	**3**	Apply Laplace transform techniques to analyze the stability of continuous-time systems.	**3,5**
Practical 6	**Fourier Series Representation of Periodic Signals:** Compute the Fourier series coefficients of a periodic signal and reconstruct the signal using MATLAB/Python.	**3**	Analyze and represent periodic signals using Fourier series.	3,4
Practical 7	**Fourier Transform of Continuous-Time Signals:** Compute the Fourier transform of given continuous-time signals and analyze their frequency components.	**3**	Understand the frequency domain representation of signals using Fourier transform.	**3,4**
Practical 8	**Sampling Theorem and Reconstruction of Signals:** Demonstrate the sampling theorem by sampling a continuous-time signal at different rates and reconstructing it using MATLAB/Python.	**3**	Verify the sampling theorem and understand its significance in signal processing.	**3,5**
Practical 9	**Z-Transform and System Analysis:** Compute the Z-transform of discrete-time signals and analyze the stability of a discrete-time LTI system using poles and zeros	**3**	Apply Z-transform techniques for analyzing discrete-time LTI systems.	**3,4**

Practical 10	**Frequency Response of LTI Systems:** Analyze the frequency response of continuous and discrete-time LTI systems using MATLAB/Python.	**3**	Evaluate the frequency response of LTI systems and interpret system behavior in the frequency domain.	**4,5**

TEXTBOOKS:

T1: Oppenheim Alan, V., Willsky Alan. S., and Nawab, H., "Signals and Systems", Prentice Hall, 1997.

REFERENCE BOOKS:

R1: Haykin Simon, "Communication Systems", 3rd Edition, John Wiley, 1995.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basics of data communication, networking, internet, physical layer techniques, and circuit switching.	PO1, PO2, PO3, PO4, PO5, PO12
2	Understand Classifications of the standard forms of Signals with respect to systems based on their properties.	PO1, PO2, PO3, PO4, PO5, PO12
3	Analyse the concepts of Laplace transform based continuous time and discrete time analysis of signals and systems.	PO1, PO2, PO3, PO4, PO5, PO12
4	Analyzation of the concepts of Fourier transform based continuous time and discrete time analysis of signals and systems.	PO1, PO2, PO3, PO4, PO5, PO12
5	Analyzation of Z-transform in continuous time signals and systems by using appropriate mathematical tools.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM3102R	Signals and systems	CO 1	3	1	2	1	1							3
		CO 2	3	2	3	1	1							3
		CO 3	3	2	3	1	1							3
		CO 4	3	2	3	1	2							3
		CO 5	3	3	3	3	2					1		3

SEMESTER – V									
Course Title	**UI/UX Design**								
Course code	**24BCSM3103R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**2**	**0**	**2**	**4**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives (Minimum 3)	1. To understand the fundamental principles of User Interface (UI) and User Experience (UX) design. 2. To develop skills in creating user-centered designs through user research, wireframing, and prototyping. 3. To learn the application of visual design elements such as typography, color theory, and layout in creating intuitive interfaces. 4. To explore tools and techniques for usability testing and improving accessibility in digital products. 5. To enable students to design responsive and interactive interfaces that adapt to various platforms and devices.								
CO1	Demonstrate an understanding of the core principles of UI/UX design and their application in creating user-friendly interfaces								
CO2	Conduct user research and create personas, user journey maps, and scenarios to represent user needs and behaviors.								
CO3	Design low-fidelity wireframes and interactive prototypes using modern design tools.								
CO4	Evaluate designs for usability, accessibility, and responsiveness using testing methods and feedback analysis.								
CO5	Create visually appealing, functional, and responsive interfaces using advanced UI/UX design techniques.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Definition and Importance of UI/UX Differences between UI and UX Principles of Good Design: Usability, Accessibility, and Aesthetics Design Thinking Process: Empathize, Define, Ideate, Prototype, and Test UX Research: Methods and Techniques	**7**	Understand the role of UI/UX in product design. Explain the principles of design thinking and its application. Conduct basic UX research to gather user insights	**1,2,3**
II	Visual Design Principles: Alignment, Contrast, Repetition, Proximity Typography and Color Theory in UI Design Layout and Grids for Responsive Design Tools for UI Design: Figma, Adobe XD, Sketch (Introduction) Low-Fidelity Prototyping	**8**	Identify and apply visual design principles in UI design. Use prototyping tools to create low-fidelity wireframes. Analyze and improve the visual hierarchy of existing designs.	**2,3,4**
III	UX Design Phases: Research, Ideation, and Testing	**7**	Create effective user flows and interaction designs for applications.	**3,4**

	User Flow and Interaction Design Information Architecture and Navigation Design Wireframing and Mockups: Best Practices Case Studies in UX Design		Design intuitive navigation structures and information architecture.	
IV	Basics of typography, color theory, and layouts Designing for responsiveness (mobile and desktop) Creating intuitive interaction designs	**8**	Design visually appealing and interactive user interfaces while considering responsiveness.	**4**
V	Basics of usability testing methods Gathering and analyzing user feedback Iterating designs based on feedback	**6**	Perform basic usability testing and refine designs based on feedback.	**5**
Practical Component				
Practical 1	**Understanding Good vs. Bad Design:** Analyze and compare good and bad user interfaces in real-world websites/apps.	**3**	Identify design principles such as usability, accessibility, and aesthetics by evaluating examples.	1,2,4
Practical 2	**Creating User Personas** Perform user research and create 2-3 personas for a target application (e.g., e-commerce, social media)	**3**	Develop user personas to represent diverse user needs and behaviors.	3,4
Practical 3	**Designing User Journey Maps** Map out the user journey for a specific task (e.g., purchasing a product online).	**3**	Understand and visualize the steps a user takes to achieve a goal in a system.	3,5
Practical 4	**Creating Low-Fidelity Wireframes** Sketch low-fidelity wireframes for a simple mobile app (e.g., weather app or task manager).	**3**	Build simple and clear wireframes representing basic layouts and user flows.	3,4
Practical 5	**Prototyping :** Create an interactive prototype for a single feature of a mobile app using a prototyping tool.	**3**	Translate wireframes into clickable prototypes to demonstrate interactions.	3,4
Practical 6	**Visual Design: Typography and Color Scheme:** Design a login page using appropriate typography, colors, and alignment.	**3**	Apply typography and color theory principles to create visually appealing UI designs.	3,5
Practical 7	**Responsive Design:** Design a simple webpage that adapts to mobile and desktop views.	**3**	Identify usability issues and refine the design based on test results.	4,5
Practical 8	**Interaction Design: Micro-interactions:** Add animations or micro-interactions to a prototype (e.g., button hover effects or loading animations).	**3**	Design intuitive and engaging micro-interactions for enhancing user experience.	3,4

Practical 9	**Accessibility Testing:** Test an existing app or website for accessibility compliance.	**3**	Evaluate the accessibility of digital products and suggest improvements for inclusivity.	5
Practical 10	**Usability Testing on a Prototype:** Conduct usability testing on an app prototype and gather user feedback.	**3**	Identify usability issues and refine the design based on test results	4,5

Textbooks

T1: "Don't Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug (3rd Edition, 2014)

T2: "The Design of Everyday Things" by Don Norman (Revised Edition, 2013)

Reference Books

R1: "About Face: The Essentials of Interaction Design" by Alan Cooper, Robert Reimann, and David Cronin(4th Edition, 2014)

R2: "Lean UX: Designing Great Products with Agile Teams" by Jeff Gothelf and Josh Seiden (2nd Edition, 2016)

Other Learning Resources

O1: "Google UX Design Professional Certificate" on Coursera

O2: Figma's Official Learning Platform (https://figma.com/resources/learn/)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate an understanding of the core principles of UI/UX design and their application in creating user-friendly interfaces	PO2, PO3, PO4, PO6, PO8, PO9, PO10, PO12
2	Conduct user research and create personas, user journey maps, and scenarios to represent user needs and behaviors.	PO2, PO3, PO4, PO6, PO8, PO9, PO10, PO12
3	Design low-fidelity wireframes and interactive prototypes using modern design tools.	PO2, PO3, PO4, PO6, PO8, PO9, PO10, PO12
4	Evaluate designs for usability, accessibility, and responsiveness using testing methods and feedback analysis.	PO2, PO3, PO4, PO6, PO8, PO9, PO10, PO12
5	Create visually appealing, functional, and responsive interfaces using advanced UI/UX design techniques.	PO2, PO3, PO4, PO6, PO8, PO9, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM3103R	UI/UX Design	CO 1		2	2	3		1		3	1	1		2
		CO 2		3	2	3		1		2	1	1		2
		CO 3		3	2	2		1		2	1	1		2
		CO 4		2	3	2		1		2	1	1		2
		CO 5		2	3	3		1		3	1	1		2

SEMESTER – V									
Course Title	**Software Engineering**								
Course code	**24BCSM3104R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Computer Science	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives (Minimum 3)	1. Master software development methodologies. 2. Design scalable and maintainable software architectures. 3. Implement effective software testing and quality assurance practices.								
CO1	Demonstrate ethical software development practices.								
CO2	Apply systems development lifecycle phases effectively.								
CO3	Elicit, analyze, and specify software requirements collaboratively.								
CO4	Create and evaluate standard procedures and documentation.								
CO5	Collaborate productively in interdisciplinary software project teams.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	Introduction: Lifecycle models Function-oriented software design: Structured analysis and structured design.	**7**	Students will gain proficiency in life cycle models and function-oriented software design through structured analysis and design methodologies, preparing for effective software development practices.	**1,2**
II	Software requirements, analysis, and specification: Informal and formal specification.	**8**	Develop skills in software requirements, analysis, and specification, encompassing both informal and formal specification methods to ensure comprehensive software development understanding and practice.	**1,2,3**
III	Object-Oriented Design: User interface design, GUI design primitives, Window management system, and the X Windows system. Coding and Testing: Coding standards and unit testing.	**7**	Master Object-Oriented Design principles including user interface and GUI design primitives, along with Window management systems like X Windows. Understand coding standards and practice effective unit testing methods to ensure robust software development.	**1,2,3**
IV	Project management: Estimation, scheduling, risk management, and configuration management.	**8**	Acquire proficiency in project management essentials such as estimation, scheduling, risk management, and	**1,2,3**

			configuration management, crucial for effective planning and execution of software projects.	
V	Software reliability and quality assurance: Reliability metrics and growth modeling, ISO-9000, SEI, and CMM.	**6**	Explore software reliability and quality assurance, focusing on reliability metrics, growth modelling, and standards like ISO-9000, SEI, and CMM to ensure high-quality software development practices and processes.	**1,2**
Practical Component				
Practical 1	Compare and analyse various SDLC models (e.g., WaterSummer, Agile, Spiral) for different project scenarios.	**3**	Understand the features, advantages, and limitations of different lifecycle models.	1,2,4
Practical 2	Develop a Data Flow Diagram (DFD) for a given system to represent its functional components and data flow.	**3**	Apply structured analysis techniques to create functional models for software systems.	3,4
Practical 3	Prepare a formal SRS document for a given problem statement using IEEE standards.	**3**	Write clear, concise, and comprehensive software requirement specifications.	3,5
Practical 4	Create UML diagrams (class, sequence, activity, and use case diagrams) for a software system.	**3**	Understand and apply object-oriented design principles using UML.	3,4
Practical 5	Design and develop a user-friendly graphical user interface (GUI) for a given application using GUI primitives.	**3**	Design intuitive and accessible user interfaces for software systems.	3,4
Practical 6	Write code following industry-standard coding guidelines and perform unit testing using tools like JUnit or PyTest.	**3**	Develop high-quality code adhering to coding standards and validate it using unit tests.	3,5
Practical 7	Estimate project effort and duration using techniques like Function Point Analysis (FPA) and create a Gantt chart for scheduling.	**3**	Apply estimation and scheduling techniques to plan software projects effectively.	4,5
Practical 8	Identify potential risks in a software project, assess their impact, and develop a risk mitigation plan.	**3**	Understand risk management processes and apply them to reduce project uncertainties.	3,4
Practical 9	Implement quality assurance techniques and explore ISO-9000 standards and SEI CMM levels for software quality management.	**3**	Demonstrate knowledge of software quality standards and assurance techniques.	2,4
Practical 10	Calculate reliability metrics (MTBF, MTTR) and use a reliability growth model to assess software reliability.	**3**	Apply reliability metrics to evaluate and improve software robustness.	4,5

TEXT BOOKS:

T1: An Integrated Approach to Software Engineering by Jalote. Narosa Publishing House.
T2: Software Engineering by R. Mall, PHI.

REFERENCE BOOKS:

R1: Software Engineering by R.S.Pressman, McGraw Hill

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate ethical software development practices.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
2	Apply systems development lifecycle phases effectively.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
3	Elicit, analyze, and specify software requirements collaboratively.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
4	Create and evaluate standard procedures and documentation.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12
5	Collaborate productively in interdisciplinary software project teams.	PO1, PO2, PO3, PO4, PO5, PO9, PO11, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BCSM3104R	Software Engineering	CO1	2	2	3	2	2				3		3	3
		CO2	3	2	3	2	2				2		3	3
		CO3	3	2	3	3	3				3		3	2
		CO4	3	2	3	1	3				2		3	2
		CO5	3	2	3	3	3				2		2	2

SEMESTER – V									
Course Title	**Design Thinking and Entrepreneurship**								
Course code	**24BCSM3105R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**1**	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer / I semester of the third year of the programme**								
Course Objectives (Minimum 3)	1. To introduce students to the principles and processes of design thinking for solving real-world problems empathetically and innovatively. 2. To enable students to conduct contextualized research and define user-centric needs. 3. To develop creative problem-solving skills through brainstorming, idea evaluation, and iterative prototyping. 4. To foster entrepreneurial mindsets by integrating innovation strategies into the design and development processes. 5. To equip students with the ability to test, refine, and launch user-centric solutions effectively.								
CO1	Compare and select problems suitable for DT projects and use techniques for empathetic research.								
CO2	Identify and document insights, user habits and identify user needs.								
CO3	Visualize solutions, evaluate solution concepts and able to create rough prototypes, gather feedback.								
CO4	Able to create high-fidelity prototypes. Able to test user experience								
CO5	Able to identify a business model for a solution concept. Able to estimate financial results2. Apply the various techniques of parsing to construct a syntax analyzer for a specific programming language.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Design Thinking for Contextualized Problem-Solving and Empathetic Research.	8	Master design thinking for empathetic research and contextualized problem-solving, enhancing innovation and user-centered solutions.	1,2
II	Analysis of Research, Defining Needs	8	Gain proficiency in analyzing research data to accurately define needs, facilitating effective problem-solving and decision-making processes.	2,3,4
III	Brainstorming and Evaluation,	8	Develop skills in brainstorming and evaluating ideas to foster creativity and critical thinking for effective decision-making and problem-solving.	2,5
IV	Prototyping, Testing and testing to Launch.	8	Students will acquire proficiency in prototyping, testing, and launching processes to validate ideas and products efficiently, ensuring successful implementation and user satisfaction.	3,4,5
V	Entrepreneurial Innovation.	8	Develop entrepreneurial skills and strategies for fostering innovation, enabling effective identification, development, and implementation of business opportunities.	4,5

TEXT BOOKS:

T1: Ries, E. - "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses", Crown Business, 2011.

REFERENCE BOOKS:

R1: Van Der Pijl, P., Lokitz, J., & Solomon, L. K. (2016). Design a better business: New tools, skills, and mindset for strategy and innovation. John Wiley & Sons.

R2: Kelley, T., & Kelley, D. - "Creative Confidence: Unleashing the Creative Potential Within Us All", Crown Business, 2013.

OTHER LEARNING RESOURCES:

O1: "Design Thinking for Innovation" by the University of Virginia (Coursera).

O2: "Entrepreneurship 10"1by MIT OpenCourseWare.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Compare and select problems suitable for DT projects and use techniques for empathetic research.	PO1, PO2, PO3, PO4, PO10, PO12
2	Identify and document insights, user habits and identify user needs.	PO1, PO2, PO3, PO4, PO10, PO12
3	Visualise solutions, evaluate solution concepts and able to create rough prototypes, gather feedback.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Able to create high-fidelity prototypes. Able to test user experience	PO1, PO2, PO3, PO4, PO10, PO12
5	Able to identify a business model for a solution concept. Able to estimate financial results2. Apply the various techniques of parsing to construct a syntax analyzer for a specific programming language.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**

24BCSM3105R	**Design Thinking**	**CO 1**		2	2	3		1		3	1	1		2
		CO 2		3	2	3		1		2	1	1		2
		CO 3		3	2	2		1		2	1	1		2
		CO 4		2	3	2		1		2	1	1		2
		CO 5		2	3	3		1		3	1	1		2

SEMESTER - V									
Course Title	**Essence of Indian Traditional Knowledge**								
Course code	**24BCSM3105R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the programme**								
Course Objectives	1. To introduce the concept and relevance of Indian traditional knowledge in various domains of life. 2. To explore the foundational Indian philosophical traditions and their influence on ethics and values. 3. To study the contributions of traditional Indian knowledge to science, technology, and sustainability. 4. To appreciate the cultural heritage and traditional art forms of India as a source of knowledge and creativity. 5. To promote the application of traditional knowledge systems for holistic living and addressing modern challenges.								
CO1	Understand the scope and importance of Indian traditional knowledge in contemporary contexts.								
CO2	Analyze Indian philosophical traditions and their relevance to ethics and value systems.								
CO3	Examine the contributions of traditional knowledge systems to science, technology, and environmental conservation.								
CO4	Appreciate the role of cultural heritage and art forms in shaping India's identity and global recognition.								
CO5	Apply principles of Indian traditional knowledge to promote sustainable and holistic living.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	- Definition and importance of traditional knowledge. - Ancient systems of knowledge in India: Vedic, Upanishadic, and other ancient texts. - Scope and relevance of traditional knowledge in contemporary society.	**7**	Understand the concept and relevance of Indian traditional knowledge in various domains of life.	**1,2,3**
II	- Introduction to six schools of Indian philosophy (Shad-Darshanas). - Key ideas of Advaita Vedanta, Sankhya, and Yoga. - Influence of Indian philosophy on ethics and values in modern life.	**7**	Understand the fundamental Indian philosophical schools and their contributions to knowledge systems.	**2,3,4**
III	- Ancient Indian contributions to mathematics, astronomy, and medicine (Ayurveda). - Traditional agricultural practices and environmental conservation techniques.	**7**	Explore the contributions of traditional Indian knowledge to science, technology, and environmental sustainability.	**3,4**

	- Integration of traditional and modern science for sustainable development.			
IV	- Indian classical music and dance (Bharatanatyam, Kathak, etc.). - Traditional crafts, architecture, and their socio-economic significance. - Role of cultural heritage in fostering national identity and global recognition.	**7**	Appreciate Indian cultural heritage and art forms as expressions of traditional knowledge and creativity.	**3,4**
V	- Yoga and Ayurveda for physical and mental well-being. - Ethical principles and sustainability in traditional Indian practices. - Relevance of traditional knowledge in solving modern-day problems.	**7**	Understand the role of traditional knowledge in promoting a holistic and balanced way of life.	**3,5**

Textbooks

T1: Kapil Kapoor & Michel Danino - "Knowledge Traditions and Practices of India", CBSE, 2012.

T2: Dayalbagh Educational Institute - "Essence of Indian Knowledge Tradition (EIKT)", New Age International Publishers, 2019.

Reference Books

R1: Satya Prakash et al. - "Foundations of Indian Culture", Bharatiya Vidya Bhavan, 1994.

R2: M. Hiriyanna - "Essentials of Indian Philosophy", Motilal Banarsidass Publishers, 2000.

Other Learning Resources

O1: NPTEL Course: "Introduction to Indian Knowledge Systems" by Prof. B. Mahadevan, IIM Bangalore.

O2: SWAYAM Course: "Understanding Indian Culture" by IGNOU.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the scope and importance of Indian traditional knowledge in contemporary contexts.	PO6, PO12
2	Analyze Indian philosophical traditions and their relevance to ethics and value systems.	PO6, PO12
3	Examine the contributions of traditional knowledge systems to science, technology, and environmental conservation.	PO6, PO12
4	Appreciate the role of cultural heritage and art forms in shaping India's identity and global recognition.	PO6, PO12
5	Apply principles of Indian traditional knowledge to promote sustainable and holistic living.	PO6, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM3105R	**Essence of Indian Traditional Knowledge**	**CO1**						3						2
		CO2						3						2
		CO3						2						2
		CO4						2						1
		CO5						2						1

<table>
<tr><th colspan="8">SEMESTER – V</th></tr>
<tr><td>Course Title</td><td colspan="7">Mini Project III</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BCSM3106R</td><td rowspan="2">Total credits: 1
Total hours: 48</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>4</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Pre-requisite</td><td>Nil</td><td>Co-requisite</td><td colspan="6">Nil</td></tr>
<tr><td>Programme</td><td colspan="7">Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)</td></tr>
<tr><td>Semester</td><td colspan="7">Summer/I semester of the third year of the Programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="7">1. To develop the knowledge, skills and attitudes of a professional Computer engineering professional.
2. To become confident in designing engineering solutions to complex software problems utilising a systems approach.</td></tr>
<tr><td>CO1</td><td colspan="7">Apply IT principles for real-world problem-solving using programming languages and network solutions.</td></tr>
<tr><td>CO2</td><td colspan="7">Contribute to complex projects as a team member, demonstrating effective personal and team management.</td></tr>
<tr><td>CO3</td><td colspan="7">Develop programming skills to create and assess software, hardware, and network solutions.</td></tr>
<tr><td>CO4</td><td colspan="7">Apply personal and team management skills as a professional software developer.</td></tr>
<tr><td>CO5</td><td colspan="7">Cultivate employability skills and uphold professionalism in software development.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td>Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:

1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
• theoretical studies
• computer simulations
• hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;
5. Prepare a formal report describing the work undertaken and results obtained so far.
6. Present the work in a forum involving poster presentations and</td><td>48</td><td>Conduct a research project involving literature review, technical work, progress reporting, seminar delivery, and formal presentation under academic supervision.</td><td>3, 4, 5, 6</td></tr>
</table>

	demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM3106R	**Mini Project-III**	**CO 1**	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

SEMESTER – V									
Course Title	**Predictive Analysis**								
Course code	**24BCSM3107R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**

		Total hours: 45T+30P							
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the third year of the program**								
Course Objectives (Minimum 3)	1. To familiarize the students with various Statistical Data Analysis tools 2. Tools that can be used for effective decision making. 3. Provide students with the knowledge to apply predictive analytics techniques to solve practical business problems and improve decision-making processes.								
CO1	Understand the foundational concepts and significance of predictive analytics, including predictive models, their working mechanisms, and their role in transforming data into actionable insights for future decision-making.								
CO2	Apply statistical analysis techniques, such as measures of central tendency, dispersion, moments, skewness, and kurtosis, to analyze and interpret datasets effectively.								
CO3	Gain proficiency in data mining methodologies, including CRISP-DM framework, and identify the stages, tasks, and strategies involved in managing and executing a data mining project.								
CO4	Explore and apply the concept of unit of analysis, including data integration, CLEM expressions, and relationship modeling, to define and achieve specific modeling objectives.								
CO5	Develop and deploy predictive analytics solutions using IBM Watson Studio, leveraging its components like AutoAI, Machine Learning, Neural Network Modeler, and Data Refinery for practical use cases in data-driven decision-making.								
Unit-No.	**Content**		**Contact Hour**		**Learning Outcome**				**BL**
I	**Analytics Overview** What is Predictive Analysis, how predictive model works, why predictive modeling, what are the models in Predictive Analysis. How Predictive analytics: Transforming data into future insights, Analytics trends: Past, Present & Future, Towards a Predictive enterprise.		**7**		Understand basic analytics concepts and applications in business contexts.				1,2,3
II	**Statistical Analysis** Define Statistics and its types, Measures of Central Value; Mean, Median and Mode, Measures of Dispersion: Absolute and Relative measures of dispersion - Range, Quartile Deviation, Mean Deviation, Standard Deviation, Coefficient of Variation, Moments, Skewness, Kurtosis.		**8**		Understand fundamental statistical concepts, techniques, and their applications in data analysis to derive insights and support decision-making.				2,3,4
III	**Data Mining**		**10**		Understand data mining concepts, algorithms, and their applications to				2, 3, 4

	What is a Data Mining applications? Strategy for data mining: CRISP-DM, Steps of Data Mining, Stages and tasks in CRISP-DM, Life Cycle of a Data Mining Project, Skills Needed for Data Mining. Identify nodes and streams, The framework of a Data – mining project, Brief the unit of analysis, Explain the type of dialog box.		extract patterns and knowledge from large data sets.	
IV	**Unit of Analysis** Concepts of Unit of analysis (Distinct, Aggregate, SetToFlag), Integrate data, CLEM Expression, Role of Relationship between two fields, Identifying the modeling objective.	**12**	Understand the concept and importance of the unit of analysis in research and data analysis.	2,3, 4
V	**Predictive Analytics With Ibm Watson Studio** Understanding of IBM Cloud, IBM Watson Studio, Watson studio Components, Creating a Machine Learning Model, Data preparation, Watson Machine learning, Data Refinery, Watson Studio Neural Network Modeler, IBM Watson Studio jobs, Use case with AutoAI.	**8**	Understand predictive analytics concepts and IBM Watson Studio features for building predictive models. (2 - Understand). Apply IBM Watson Studio tools to develop and deploy predictive models effectively	2, 3, 4
Practical Component				
Practical 1	Introduction to Predictive Analysis	3	Explain the concept of predictive analysis and its importance in decision-making.	2
Practical 2	Predictive Modeling	**3**	Describe how predictive models work and their applications in various domains.	2
Practical 3	Analytics Trends Over Time	3	Analyze past, present, and future trends in analytics and their impact on business strategies.	4
Practical 4	Measures of Central Tendency	**3**	Compute and interpret mean, median, and mode for a dataset.	3
Practical 5	Measures of Dispersion	3	Calculate and compare measures of dispersion such as range, standard deviation, and coefficient of variation.	4
Practical 6	Skewness and Kurtosis Analysis	**3**	Evaluate skewness and kurtosis to understand the	3

			shape and distribution of data.	
Practical 7	CRISP-DM Framework Overview	3	Outline the stages and tasks in the CRISP-DM framework for data mining projects.	3
Practical 8	Experiment: Data Mining Project Life Cycle	3	Apply the stages of a data mining project life cycle to a practical case study.	4
Practical 9	Data Preparation Using Watson Studio	3	Utilize IBM Watson Studio's Data Refinery for data preparation tasks like cleaning and feature engineering.	3
Practical 10	Utilize IBM Watson Studio's Data Refinery for data preparation tasks like cleaning and feature engineering.	**3**	Differentiate between distinct, aggregate, and SetToFlag units of analysis in data mining contexts.	4

Textbooks

T1: "Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die" by Eric Siegel (2nd Edition, 2016)

T2: "Applied Predictive Modeling" by Max Kuhn and Kjell Johnson (2013).

Reference Books

R1: "Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei **(**3rd Edition, 2011)

R2: "Machine Learning and Data Science: An Introduction to Statistical Learning Methods with R" by Daniel D. Gutierrez**,** 2015

Other Learning Sources

O1: "Predictive Analytics and Data Mining" on Coursera by the University of Illinois

O2: "Advanced Predictive Modeling in R" on DataCamp

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the foundational concepts and significance of predictive analytics, including predictive models, their working mechanisms, and their role in transforming data into actionable insights for future decision-making.	PO1, PO2, PO3, PO4, PO10, PO12

2	Apply statistical analysis techniques, such as measures of central tendency, dispersion, moments, skewness, and kurtosis, to analyze and interpret datasets effectively.	PO1, PO2, PO3, PO4, PO10, PO12
3	Gain proficiency in data mining methodologies, including CRISP-DM framework, and identify the stages, tasks, and strategies involved in managing and executing a data mining project.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Explore and apply the concept of unit of analysis, including data integration, CLEM expressions, and relationship modeling, to define and achieve specific modeling objectives.	PO1, PO2, PO3, PO4, PO10, PO12
5	Develop and deploy predictive analytics solutions using IBM Watson Studio, leveraging its components like AutoAI, Machine Learning, Neural Network Modeler, and Data Refinery for practical use cases in data-driven decision-making.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM3107R	Predictive Analysis	CO 1	2	2	2	2	3					2		2
		CO 2	2	3	3	2	3					2		2
		CO 3	2	2	2	2	3					2		2
		CO 4	2	2	2	3	3					2		2
		CO 5	3	3	3	3	3					2		2

SEMESTER – VI									
Course Title	**Web Technologies**								
Course code	**24BCSM3201R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. To make a student familiar with client server architecture. 2. To teach developing a web application using various technologies. 3. To make the students learn the skills and project-based experience needed for entry into web application and development careers..								
CO1	Learn about Internet and Web Page Design.								
CO2	Understand detailed concepts on Web Browsers, Markup Language Basics and XML.								
CO3	Learn basic concepts of client side programming								
CO4	Learn detailed concepts on Web Server Side technologies.								
CO5	Understand few advanced Web Technologies and the Web Security.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**An Introduction to Web Technology** History of web Development, Time line, Motivation, Categories of Web Applications, Characteristics of Web Applications. Evolution and Need for Web Engineering, Web Engineering Models, Software Engineering v/s Web Engineering . World Wide Web: Introduction to TCP/IP and WAP, DNS, Email, TelNet, HTTP and FTP. Introduction to Browser and search engines, Search fundamentals, Search strategies, Directories search engines and Meta search engines, Working of the search engines , Miscellaneous Web Browser details, Introduction to Web Servers: Features of web servers, caching, case study-IIS, Apache, Configuring web servers	**8**	To understand web development basics, HTML, CSS, JavaScript, client-server architecture, and create simple web applications.	**1,2,3**
II	**Technologies for Web Applications** HTML and DHTML, HTML Basic Concepts, Static and dynamic HTML, Structure of HTML documents, HTML Elements, Linking in HTML, Anchor Attributes, Image Maps, Meta Information, Image Preliminaries, Layouts, Backgrounds, Colors and Text, Fonts, Tables, Frames and layers, Audio and Video Support with HTML Database integration, CSS, Positioning with Style sheets, Forms Control, Form. Elements. Introduction to CGI PERL, JAVA SCRIPT, ASP, Cookies Creating and Reading Cookies. Introduction of XML, Validation of XML documents, DTD, Ways to use XML, XML for data files, HTML Vs XML, Embedding XML into HTML documents, Converting XML to HTML for Display, Displaying XML using CSS and XSL, Rewriting	**10**	To master the advanced web technologies, frameworks, APIs, databases, and develop complex, scalable web applications with robust functionality.	**3,4**

	HTML as XML, Relationship between HTML, SGML and XML			
III	**Client Side Programming** Document Object Model (DOM),Overview of DOM, Jquery: ,Element Selector Document ready function, Events ,Ajax call ,plugins: Using JQuery UI ,Unobtrusive client validation ,JQuery templates ,Feature detection: Browser detection ,Feature detection ,Modernizer , polyfills. Introduction to AJAX, Overview of Traditional Web Communication Processes and Technologies, Interacting with the Web Server Using the XMLHttpRequest , Create an XMLHttpRequest Object Interact with the Web Server Working with PHP and AJAX: Introduction Process Client Requests Accessing Files Using PHP Applying Basic AJAX Techniques: Introduction Download Images Using AJAX Auto, Populate Select Boxes, Implementing Security and Accessibility in AJAX Applications: production, Secure AJAX Applications Accessible Rich Internet Applications	**10**	To learn JavaScript, DOM manipulation, event handling, form validation, AJAX, and create interactive, responsive web interfaces.	**3,4**
IV	**Server Side Programming** J2SE:Data Types, Arrays, Type Casting, Classes and Objects, Inheritance, Interfaces, Exception Handling, Multithreading,J2EE as a framework, Client Server Traditional model, Comparison amongst 2-tier, 3-tier and N-tier Architectures, Thin and Thick Clients. J2EE Servlet 2.x Specification, Writing small Servlet Programs, Deployment Descriptor, Inter Servlet Collaboration, Session: Definition, State on web, Different ways to track sessions, JSP Technology Introduction-JSP and Servlets-Running JSP Applications Basic JSP- JavaBeans Classes - Support for the Model- View- Controller Paradigm- Case Study- Related Technologies.	**9**	To understand server-side scripting, databases, authentication, RESTful APIs, server deployment, and build dynamic, data-driven web applications.	**4**
V	**Advanced Web Technologies and Web Security** Exposure to Advanced Web Technologies: Distributed Object based models- DCOM, CORBA, EJB; Web services and Related Technologies- ISAPI, SOAP, UDDI, WSDL; Other Advanced Web Technologies- AJAX, ISAPI, .NET. Web Security: Firewalls- definition and uses, network layer firewalls and application layer firewalls; Proxy servers, HTPS for secure web communication.	**8**	To explore advanced web frameworks, progressive web apps, web security principles, encryption, and secure web application development practices.	**4**

Practical Component

Practical Number	**Topic / Experiment**	**Contact Hours**	**Learning Outcome**	**BL**

Practical 1	Identify common HTML tags and their functions.	2	Identify and label common HTML tags and understand their basic functions in web development.	1,2,3
Practical 2	Match CSS properties to their effects.	2	To match CSS properties to their corresponding effects, enhancing their understanding of CSS styling.	3,4
Practical 3	Create a concept map of basic JavaScript concepts.	2	To create a concept map, demonstrating their understanding of basic JavaScript concepts and their interrelationships.	3,4
Practical 4	Explain the structure of a basic web page.	2	analyze and explain the structure and purpose of different sections in a basic web page.	4,5
Practical 5	Use HTML and CSS to create a simple web page.	2	Use HTML and CSS to create a functional web page with various elements and styles.	3,4
Practical 6	Implement basic form validation using JavaScript.	2	implement JavaScript to perform basic form validation, ensuring proper data entry before submission.	4,5
Practical 7	Evaluate the responsiveness of a web page.	2	evaluate a web page's responsiveness, identifying areas for improvement using developer tools.	4,5
Practical 8	Analyze the performance of a web page.	2	To analyze web page performance, using tools to identify and suggest solutions for performance bottlenecks..	3,4
Practical 9	Conduct a usability test on a web application..	2	Conduct and analyze usability tests, making recommendations to improve web application user experience.	5
Practical 10	Review and evaluate the quality of web code.	2	Review and evaluate web code quality for readability, efficiency, and best practices adherence.	5,6
Practical 11	Create a web application using HTML, CSS, and JavaScript.	2	Design and build a dynamic web application using HTML, CSS, and JavaScript.	4,5
Practical 12	Develop a custom CSS framework.	2	To develop a reusable custom CSS framework for use in various web development projects.	4,5

Practical 13	Develop a single page application using a JavaScript framework (e.g., React or Vue.js).	2	To create a SPA using a JavaScript framework, integrating dynamic content through API interactions.	3,4
Practical 14	Design and implement a RESTful API.	2	To design and implement a RESTful API using Node.js and Express, connecting it to a front-end..	5
Practical 15	Develop an interactive data dashboard.	2	develop interactive dashboards to visualize and explore data insights dynamically..	5,6

Text Books:

T1: Web Technologies: HTML, Javascript, PHP, Java, JSP, XML and AJAX, Black Book, Kogent Learning Solutions Inc., 1st Edition, 2009, Dreamtech Press.

T2: Web Technologies, Uttam K. Roy, 1st Edition, 2010, Oxford Publications.

Reference Books:

R1: N. P. Gopalan and J. Akilandeswari, Web Technology: A Developer's Perspective,1st Edition, 2014,Prentice-Hall of India Pvt. Ltd.

R2: Jeffrey C. Jackson, Web Technologies: A Computer Science Perspective, 1st Edition, 2008, Pearson Education India.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Learn about Internet and Web Page Design.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
2	Understand detailed concepts on Web Browsers, Markup Language Basics and XML.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Learn basic concepts of client side programming	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Learn detailed concepts on Web Server Side technologies.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
5	Understand few advanced Web Technologies and the Web Security.	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BCSM3201R	Web Technologies	CO1	2	2	2	2	3					2		2
		CO2	2	3	3	2	3					2		2
		CO3	2	2	2	2	3					2		2
		CO4	2	2	2	3	3					2		2
		CO5	3	3	3	3	3					2		2

SEMESTER – VI									
Course Title	**Compiler Design**								
Course code	**24BCSM3202R**	**Total credits: 4** **Total hours: 45T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**1**	**0**	**0**	**0**	**0**	**4**
Pre-requisite	Theory of Computation	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Discuss the process of translating a high-level language to machine code required for compiler construction. 2. Analyze the software tools and techniques used in compiler construction such as lexical analyser and parser generators 3. Discuss the data structures used in compiler construction such as abstract syntax trees, symbol tables, three address code and stack machines.								
CO1	Understand different phases of a compiler and explain lexical analysis principles.								
CO2	Apply parsing techniques to build a language-specific syntax analyser.								
CO3	Analyse syntax-directed translations, evaluation order, and type checking.								
CO4	Understand the association of runtime storage allocation with control flow and procedure calls.								
CO5	Apply code optimization on intermediate code for target code generation.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Compiler structure:** Analysis-synthesis model of compilation, Various phases of a compiler, Tool based approach to compiler construction. Lexical analysis: Interface with input, Parser and symbol table, Token, Lexeme and patterns, Regular definition, Transition diagrams, LEX.	**8**	Understand the analysis-synthesis model of compilation and the role of lexical analysis in converting source code into tokens using tools like LEX.	**1,2,3**
II	**Syntax analysis:** CFG's, Ambiguity, Associativity, Precedence, Top-down parsing, Recursive descend parsing, Transformation on the grammars, Predictive parsing, Bottom-up parsing, Operator precedence grammars, LR parsers (SLR, Canonical, LALR), YACC	**8**	Grasp the concepts of context-free grammars (CFGs) and develop skills in various parsing techniques including top-down and bottom-up parsing methods, utilizing tools like YACC.	**3,4**
III	**Syntax directed translation:** Inherited and synthesized attributes, Dependency graph, Evaluation order, Bottom up evaluation of S- attributed definitions L- attributed definitions and top down translation of attributes Type checking: Type system, Type expressions, Structural and name equivalence of types, Type conversion.	**10**	Learn to design and implement syntax-directed translation schemes, focusing on attribute grammars and type checking mechanisms, and understand the differences between structural and name equivalence of types.	**3,4**
IV	**Run time environments:** Storage organization, Storage-allocation strategies, Access to nonlocal names, Activation tree, Activation record, Parameter passing,	**10**	Comprehend the organization and management of run-time storage, including activation records and	**4**

	Symbol table and dynamic storage allocation Intermediate code generation: Intermediate representations, Translation of declarations, Assignments, Control flow, Boolean expressions and procedure calls.		parameter passing techniques, and learn the principles of intermediate code generation.	
V	**Code generation:** Issues in the design of a code generator, Basic blocks and flow graphs, Next use information, Register allocation, Code generation algorithm, Dag representation of programs, Code generation from dags, Peephole optimization and code generator generators.	**9**	Acquire knowledge on the design and implementation of a code generator, emphasizing optimization techniques, basic blocks, flow graphs, and register allocation strategies.	**4**

TEXT BOOKS:

T1: A.V. Aho, R. Sethi, J.D. Ullman, "Compilers: Principles, Techniques and Tools", Addison – Wesley.
T2: Steven S. Muchnick, "Advanced Compiler Design and Implementation", Elsevier

REFERENCE BOOKS:

R1: W. Appel, "Modern Compiler Implementation in C: Basic design", Cambridge Press
R2: Fraser and Hanson, "A Retargetable C Compiler: Design and Implementation", Addison-Wesley
R3: Dhamdhere, "Compiler Construction", McMillan

OTHER LEARNING RESOURCES: nil

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand different phases of a compiler and explain lexical analysis principles.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
2	Apply parsing techniques to build a language-specific syntax analyser.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
3	Analyse syntax-directed translations, evaluation order, and type checking.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
4	Understand the association of runtime storage allocation with control flow and procedure calls.	CO1, CO2, CO3, CO4, CO5, CO10, CO12
5	Apply code optimization on intermediate code for target code generation.	CO1, CO2, CO3, CO4, CO5, CO10, CO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**

24BCSM3202R	**Compiler Design**	**CO 1**	3	2	1	2	2					2		3
		CO 2	3	3	3	3	2					2		3
		CO 3	3	3	3	3	2					2		3
		CO 4	3	1	2	2	2					2		3
		CO 5	3	3	3	3	2					2		3

SEMESTER – VI									
Course Title	**Design and Analysis of Algorithms**								
Course code	**24BCSM3203R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Data Structures	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Analyse the asymptotic performance of algorithms. 2. Demonstrate a familiarity with major algorithms and data structures. 3. Apply important algorithmic design paradigms and methods of analysis.								
CO1	Understand analysis of algorithms using asymptotic notations for advanced problem-solving.								
CO2	Implement sorting, searching, matrix multiplication algorithms with varied strategies.								
CO3	Apply greedy approaches and dynamic programming to optimize problem-solving, for advanced algorithm design.								
CO4	Apply problem-solving strategies to complex scenarios, solving N-Queen, knapsack, and TSP, etc.								
CO5	Evaluate decision tree bounds, interpret complexity classes P, NP, NP-Complete proficiently.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction: Characteristics of algorithm.** Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.	**8**	Understand and analyze the best, average, and worst-case complexities of algorithms.	**2,4**
II	**Fundamental Algorithmic Strategies:** Brute-Force, Greedy, Dynamic Programming, Branch and Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics –characteristics and their application domains.	**8**	Apply different algorithmic strategies such as Brute-Force, Greedy, and Dynamic Programming to solve problems.	**3,4,5**
III	**Graph and Tree Algorithms:** Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.	**10**	Implement traversal algorithms and determine the shortest paths in graphs.	**3,4**
IV	**Tractable and Intractable Problems:**	**10**	Differentiate between P, NP, NP-complete, and NP-hard classes and	**2,4,5**

	Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques		understand the significance of Cook's theorem.	
V	**Advanced Topics:** Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE	**9**	Develop and analyze approximation and randomized algorithms for complex problems.	**3,4,6**

Practical Component

Practical Number	**Topic / Experiment**	**Contact Hours**	**Learning Outcome**	**BL**
Practical 1	Implement and analyze the time complexity of Bubble Sort, Selection Sort, and Insertion Sort.	**2**	Compare the time complexities of different sorting algorithms in best, average, and worst cases.	1,2,3
Practical 2	Use the substitution method to solve simple recurrence relations.	**2**	Apply the substitution method to determine the time complexity of recursive algorithms.	3,4
Practical 3	Implement a recursive algorithm and analyze its performance using the recursion tree method.	**2**	Visualize and analyze the time complexity of recursive algorithms using recursion trees.	3,4
Practical 4	Solve the 0/1 Knapsack problem using dynamic programming.	**2**	Develop and apply dynamic programming to solve optimization problems.	4,5
Practical 5	Implement a solution to the Traveling Salesman Problem using the branch and bound technique.	**2**	Apply branch and bound to solve combinatorial optimization problems.	3,4
Practical 6	Design and implement a greedy algorithm for the Fractional Knapsack problem.	**2**	Understand and apply the greedy strategy to solve optimization problems.	4,5
Practical 7	Implement backtracking to solve the N-Queens problem.	**2**	Apply backtracking techniques to solve constraint satisfaction problems.	4,5
Practical 8	Implement a brute-force solution for the Bin Packing problem and evaluate its efficiency.	**2**	Evaluate the efficiency of brute-force algorithms for combinatorial problems.	3,4
Practical 9	Implement Depth First Search (DFS) and Breadth First Search (BFS) for graph traversal.	**2**	Implement and compare DFS and BFS traversal techniques.	5
Practical 10	Implement Dijkstra's algorithm to find the shortest path in a weighted graph.	**2**	Apply shortest path algorithms to find the minimum distance between nodes in a graph.	5,6
Practical 11	Implement Kruskal's and Prim's algorithms to find the Minimum Spanning Tree (MST).	**2**	Compare and apply different algorithms to find MST in a graph.	4,5
Practical 12	Perform topological sorting on a directed acyclic graph (DAG).	**2**	Understand and implement topological sorting on DAGs.	4,5

Practical 13	Implement the reduction technique to show that a problem is NP-complete.	**2**	Demonstrate the reduction technique to classify problems within NP-complete.	3,4
Practical 14	Solve an NP-hard problem using approximation algorithms.	**2**	Apply approximation techniques to find near-optimal solutions to NP-hard problems.	5
Practical 15	Implement a randomized algorithm for the Minimum Cut problem.	**2**	Understand and apply randomized algorithms to solve complex problems.	5,6

TEXT BOOKS:

T3: Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.

T4: Fundamentals of Algorithms – E. Horowitz et al.

REFERENCE BOOKS:

R1: Algorithm Design, 1ST Edition, Jon Kleinberg and ÉvaTardos, Pearson.

R2: Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition,

R3: Michael T Goodrich and Roberto Tamassia, Wiley.

R4: Algorithms -- A Creative Approach, 3RD Edition, UdiManber, Addison-Wesley, Reading, MA.

OTHER LEARNING RESOURCES:

1. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/
2. https://www.vssut.ac.in/lecture_notes/lecture1428551222.pdf

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand analysis of algorithms using asymptotic notations for advanced problem-solving.	CO1, CO2, CO3, CO4, CO11, CO12
2	Implement sorting, searching, matrix multiplication algorithms with varied strategies.	CO1, CO2, CO3, CO4, CO11, CO12
3	Apply greedy approaches and dynamic programming to optimize problem-solving, for advanced algorithm design.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12
4	Apply problem-solving strategies to complex scenarios, solving N-Queen, knapsack, and TSP, etc.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12
5	Evaluate decision tree bounds, interpret complexity classes P, NP, NP-Complete proficiently.	CO1, CO2, CO3, CO4, CO9, CO10, CO11, CO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM3203R	Design & Analysis of Algorithms	CO1	2	2	2	2							3	2
		CO2	2	2	2	2							2	2
		CO3	2	2	3	3					1	1	2	2
		CO4	2	3	2	2					1	1	2	2
		CO5	1	2	1	2	2				1	1	1	2

SEMESTER VI									
Course Title	**Statistical Methods and Modelling**								
Course code	**24BCSM3204R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T + 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Mathematics	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives	1. To provide a comprehensive understanding of statistical methods and their application in data analysis and decision-making. 2. To equip students with the ability to perform hypothesis testing and statistical inference to draw meaningful conclusions from data. 3. To develop skills in regression, correlation, and advanced statistical modeling for real-world problem-solving. 4. To introduce time series analysis and predictive modeling techniques for forecasting and trend analysis. 5. To enable students to apply statistical tools and software to analyze and visualize data effectively.								
CO1	Understand fundamental statistical concepts and probability distributions to analyze data.								
CO2	Apply statistical inference techniques for hypothesis testing and decision-making.								
CO3	Develop regression and correlation models to establish relationships between variables.								
CO4	Implement advanced statistical modeling methods, including ANOVA and time series analysis, for complex datasets.								
CO5	Utilize statistical tools and programming languages to analyze and visualize data effectively.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Statistics and Probability:** - Descriptive Statistics: Measures of Central Tendency (Mean, Median, Mode), Measures of Dispersion (Variance, Standard Deviation, Range). - Probability Basics: Definitions, Rules of Probability, Conditional Probability, Bayes' Theorem. - Probability Distributions: Binomial, Poisson, and Normal Distributions, Introduction to Sampling Distributions	**8**	Understand basic statistical concepts and probability theory for analyzing and modeling data.	**1,2, 3**
II	**Statistical Inference and Hypothesis Testing :** - Sampling Techniques: Simple Random Sampling, Stratified Sampling, and Systematic Sampling. - Hypothesis Testing: Null and Alternative Hypotheses, Type I and Type II Errors, p-values, and Confidence Intervals.	**12**	Apply statistical inference methods to draw conclusions and test hypotheses based on sample data.	**3,4**

	- Parametric Tests: Z-Test, t-Test (One-sample, Two-sample), F-Test. - Non-Parametric Tests: Chi-Square Test, Mann-Whitney U Test, Kruskal-Wallis Test.			
III	**Regression and Correlation:** - Simple Linear Regression: Fitting a Line, Interpretation of Coefficients, and Assumptions. - Multiple Linear Regression: Interpretation of Results, Multicollinearity, and Model Evaluation Metrics. - Correlation: Pearson, Spearman, and Kendall Tau Correlation Coefficients. - Logistic Regression: Binary Classification Problems, Odds Ratio, and Model Evaluation Metrics (ROC, AUC).	**9**	Model relationships between variables using regression and correlation techniques.	**3,4**
IV	**Advanced Statistical Modelling** : - Analysis of Variance (ANOVA): One-Way and Two-Way ANOVA, Interpretation of Results. - Time Series Analysis: Components (Trend, Seasonality, Randomness), Moving Averages, Exponential Smoothing. - Introduction to Generalized Linear Models (GLM): Poisson Regression, Multinomial Logistic Regression. - Model Diagnostics and Residual Analysis.	**9**	Develop advanced models for handling complex datasets and patterns.	**4**
V	**Applications of Statistical Methods in Real-World Problems :** - Case Study 1: Statistical Analysis for Market Research and Consumer Behavior. - Case Study 2: Predictive Modelling in Healthcare (e.g., Survival Analysis). - Case Study 3: Time Series Analysis for Stock Market Predictions. - Data Visualization Techniques: Static and Interactive Plots using tools like Python (Matplotlib, Seaborn) or R (ggplot2).	**9**	Apply statistical methods to solve real-world problems across domains like business, healthcare, and social sciences.	**4**
Practical Component				
Practical 1	Calculate measures of central tendency (mean, median, mode) and dispersion (variance, standard deviation). Create visualizations such as histograms, box plots, and scatter plots.	**2**	Summarize and visualize data to understand its structure and distribution.	1,2

Practical 2	Generate and analyze binomial, Poisson, and normal distributions. Fit data to these distributions and interpret the results.	**2**	Understand and apply probability distributions to model real-world scenarios.	1,3
Practical 3	Perform a one-sample t-test, two-sample t-test, and paired t-test on given datasets. Interpret the p-values and draw conclusions.	**2**	Conduct hypothesis tests and interpret results for decision-making.	2,3
Practical 4	Perform Chi-Square tests for independence and goodness-of-fit, and use Mann-Whitney U test for comparing two independent groups.	**2**	Apply non-parametric tests to datasets that do not meet parametric assumptions.	3
Practical 5	Fit a simple linear regression model to a dataset, interpret coefficients, and evaluate model performance using R^2 and residual analysis.	**2**	Model and interpret relationships between two variables using linear regression.	3,4
Practical 6	Build a multiple linear regression model to analyze relationships between a dependent variable and multiple independent variables. Check for multicollinearity and assess model performance.	**2**	Develop and evaluate regression models with multiple predictors.	1,2
Practical 7	Compute Pearson, Spearman, and Kendall Tau correlation coefficients for a dataset. Visualize correlations using a heatmap.	**2**	Analyze and interpret the strength and direction of relationships between variables.	1,3
Practical 8	Decompose a time series into trend, seasonality, and residual components. Perform moving average smoothing and exponential smoothing.	**2**	Analyze time series data and apply smoothing techniques for trend detection.	2,3
Practical 9	Perform one-way and two-way ANOVA to test differences between group means. Analyze the results and visualize group comparisons.	**2**	Compare means across multiple groups and interpret statistical significance.	3
Practical 10	Apply PCA on a high-dimensional dataset, reduce dimensionality, and visualize the results in a lower-dimensional space. Analyze the variance explained by each principal component.	**2**	Reduce dimensionality and extract meaningful patterns from high-dimensional datasets	3,4

TEXT BOOKS:

T1: Gupta, S. C., & Kapoor, V. K. - *"Fundamentals of Mathematical Statistics"*, Sultan Chand & Sons, 11th Edition, 2014.

T2: Montgomery, D. C., & Runger, G. C. - *"Applied Statistics and Probability for Engineers"*, Wiley, 7th Edition, 2020.

REFERENCE BOOKS:

R1: Freund, J. E., & Perles, B. M. - "Statistics: A First Course", Pearson, 9th Edition, 2017.

R2: Casella, G., & Berger, R. L. - "Statistical Inference", Cengage Learning, 2nd Edition, 2001.

OTHER LEARNING RESOURCES:

O1: Online Courses:

"Statistics for Data Science and Business Analysis" by Udemy.

"Statistical Thinking for Data Science and Analytics" by EdX (Columbia University).

O2: Web Resources:

Khan Academy: Modules on probability, statistics, and regression analysis.

DataCamp: Interactive tutorials on statistical modeling and data visualization using Python.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enabling solving skills of ordinary differential equations of various order.	PO1, PO2, PO5, PO12
2	Enabling solving skills of higher order ODE with constants coefficients.	PO1, PO2, PO5, PO12
3	Enable to formulate and solve problems involving random variables.	PO1, PO2, PO4, PO6, PO12
4	Enable to solve problems of various probability distribution.	PO1, PO4, PO5, PO10, PO12
5	Analysing experimental data by applying statistical methods.	PO2, PO4, PO5, PO11, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM3204R	**Statistical Methods and Modelling**	**CO 1**	3	3			2							1
		CO 2	3	3			2							2
		CO 3	2	2		2		1						1
		CO 4	2			2	2					1		1
		CO 5		3		3	2						1	2

<table>
<tr><th colspan="9">SEMESTER – VI</th></tr>
<tr><td>Course Title</td><td colspan="8">Mini Project IV</td></tr>
<tr><td rowspan="2">Course code</td><td rowspan="2">24BCSM3205R</td><td rowspan="2">Total credits: 1
Total hours: 48</td><td>L</td><td>T</td><td>P</td><td>S</td><td>R</td><td>O/F</td><td>C</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>4</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Pre-requisite</td><td>NIL</td><td>Co-requisite</td><td colspan="7">NIL</td></tr>
<tr><td>Programme</td><td colspan="8">Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)</td></tr>
<tr><td>Semester</td><td colspan="8">Winter/ II semester of the third year of the programme</td></tr>
<tr><td>Course Objectives (Minimum 3)</td><td colspan="8">1. Develop advanced technical skills through individual or group project work in engineering domains.
2. Enhance project management abilities, including workload management and meeting project deadlines effectively.
3. Improve presentation skills through seminar presentations and public demonstrations of project outcomes.</td></tr>
<tr><td>CO1</td><td colspan="8">Apply IT principles to solve real-world problems.</td></tr>
<tr><td>CO2</td><td colspan="8">Contribute to complex projects as a team member.</td></tr>
<tr><td>CO3</td><td colspan="8">Apply programming languages and platforms to develop software and hardware solutions.</td></tr>
<tr><td>CO4</td><td colspan="8">Demonstrate personal and team management skills in software development.</td></tr>
<tr><td>CO5</td><td colspan="8">Cultivate employability skills and professionalism commitment.</td></tr>
<tr><td>Unit-No.</td><td colspan="3">Content</td><td>Contact Hour</td><td colspan="2">Learning Outcome</td><td>BL</td></tr>
<tr><td>I</td><td colspan="3">This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to:
1. Perform a literature search to review current knowledge and developments in the chosen technical area;
2. Undertake detailed technical work in the chosen area using one or more of:
a. theoretical studies
b. computer simulations
c. hardware construction;
3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;
4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;</td><td>48</td><td colspan="2">Demonstrate proficiency in conducting literature reviews, performing technical work, maintaining progress reports, delivering seminars, preparing formal reports, and presenting project outcomes effectively.</td><td>2,3,4,5,6</td></tr>
</table>

	5. Prepare a formal report describing the work undertaken and results obtained so far; and 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

REFERENCE BOOKS:

R1. "The Craft of Research" by Wayne C. Booth, Gregory G. Colomb, and Joseph M. Williams

R2. "Doing Your Research Project: A Guide for First-Time Researchers" by Judith Bell and Stephen Waters

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM3205R	Mini Project-IV	CO 1	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

SEMESTER – VI									
Course Title	**Foundations of Data Science**								
Course code	**24BCSM3206R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+ 30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Mathematics and Statistics, Programming	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the third year of the Programme**								
Course Objectives (Minimum 3)	1. Develop Proficiency in Key Data Science Tools and Programming Languages. 2. Understand and Apply Fundamental Statistical and Machine Learning Methods. 3. Cultivate Critical Thinking and Problem-Solving Skills in a Data-Driven Context.								
CO1	Acquire foundational knowledge in data science.								
CO2	Understand and apply various data analytics techniques.								
CO3	Apply advanced tools to work on dimensionality reduction and mathematical operations.								
CO4	Manage diverse data types and employ programming for knowledge representation through visualization.								
CO5	Demonstrate numerous open source data science tools to solve real-world problems through industrial case studies.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Foundations of Data Science:** Overview of data science, its significance, and application across various sectors. Introduction to the data science workflow: data collection, cleaning, analysis, and visualization.	**8**	Grasp foundational concepts in data science, acquire programming skills, analyze data, and communicate findings effectively using statistical methods.	**1, 2**
II	**Statistical Methods and Data Manipulation:** Essential statistics for data science, probability theory, data summarization, and visualization techniques. Introduction to programming with Python and R for data manipulation using Pandas and NumPy.	**8**	Understand statistical techniques, manipulate data using programming languages, analyze datasets, interpret results, and communicate findings accurately.	**2,3**
III	**Machine Learning Techniques:** Supervised learning models (regression, decision trees, SVMs), unsupervised learning models (clustering, PCA), and model evaluation metrics. Introduction to machine learning algorithms implementation using scikit-learn.	**10**	Understand machine learning algorithms, apply techniques to analyze data, develop models, evaluate performance, and make predictions effectively.	**2,4**
IV	**Advanced Data Science Concepts:** Deep learning with TensorFlow, natural language processing basics, and introduction to big data technologies like Hadoop and Spark. Real-world applications and case studies.	**10**	Explore advanced topics in data science such as deep learning, natural language processing, and big data analytics, applying them effectively.	**2,3**

V	**Capstone Project in Data Science:** Application of data science processes to a real-world problem. Project phases include problem identification, data acquisition, preprocessing, modelling, analysis, and presentation of results.	**9**	Demonstrate proficiency in applying data science skills to solve real-world problems, presenting comprehensive solutions and insights effectively.	**2,5**
Practical Component				
Practical 1	Match basic data science terminology with their definitions.	2	Identify and understand key data science terms and definitions.	1,2,3
Practical 2	Identify different types of data (categorical, numerical, etc.).	2	Recognize and categorize different types of data in various data sets.	3,4
Practical 3	Interpret basic data visualizations (bar charts, histograms, scatter plots).	2	Interpret and explain information presented in basic data visualizations.	3,4,5
Practical 4	Explain the concept and importance of descriptive statistics.	2	Understand and explain the importance and application of descriptive statistics.	3,4,5
Practical 5	Perform basic data cleaning tasks.	2	Apply data cleaning techniques to prepare data sets for analysis.	3,4
Practical 6	Create data visualizations using software (e.g., Excel, Tableau, Python libraries).	2	Create accurate and effective data visualizations to represent data insights.	3,4
Practical 7	Analyze the correlation between variables.	2	Analyze and interpret the correlation between variables in data sets..	3,4
Practical 8	Perform and interpret hypothesis tests.	2	Perform hypothesis tests and interpret their significance in data analysis..	3,4
Practical 9	Evaluate the performance of machine learning models..	2	evaluate and interpret the performance of machine learning models using appropriate metrics.	3,4,5
Practical 10	Critically analyze the quality and reliability of data sources.	2	critically assess the quality and reliability of various data sources.	3,4,5
Practical 11	Build and test predictive models.	2	Build and test predictive models, demonstrating their understanding of model development.	3,4,5
Practical 12	Design a survey for data collection.	2	design effective surveys to collect relevant and unbiased data.	2,3,5
Practical 13	Implement a data processing pipeline.	2	Implement an ETL pipeline to process data efficiently.	2,3,5

Practical 14	Create a data-driven story.	2	Create and present a data-driven story, effectively communicating insights.	2,3,5
Practical 15	Develop an interactive data dashboard.	2	develop interactive dashboards to visualize and explore data insights dynamically..	2,3,5

Text Books:

T1: Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking, Foster Provost and Tom Fawcett, 1st Edition, 2013.

T2: Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython", Wes McKinney, 2nd Edition, 2017.

Reference Books:

R1: An Introduction to Statistical Learning: with Applications in R, Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, 2nd Edition, 2021.

R2: Machine Learning: A Probabilistic Perspective, Kevin P. Murphy, 1st Edition, 2012.

R3: Data Mining: Practical Machine Learning Tools and Techniques, Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal, 4th Edition, 2016.

Additional Resources:

- SKaggle - https://www.kaggle.com/
- DataCamp - https://www.datacamp.com/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Acquire foundational knowledge in data science.	CO1, CO2, CO3, CO4, CO10, CO12.
2	Understand and apply various data analytics techniques.	CO1, CO2, CO3, CO4, CO10, CO12.
3	Apply advanced tools to work on dimensionality reduction and mathematical operations.	CO1, CO2, CO3, CO4, CO10, CO12.
4	Manage diverse data types and employ programming for knowledge representation through visualization.	CO1, CO2, CO3, CO4, CO10, CO12.
5	Demonstrate numerous open source data science tools to solve real-world problems through industrial case studies.	CO1, CO2, CO3, CO4, CO10, CO12.

MAPPING TABLE

Course code	**Course Name**		**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**
24BCSM3206R	**Foundations of Data Science**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VII									
Course Title	**PE-I : Neural Networks**								
Course code	**24BCSM4101R**	**Total credits:4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**

		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Mathematics Foundations and Programming Skills	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. To introduce the fundamental concepts and architectures of neural networks. 2. To develop an understanding of training algorithms, including backpropagation and optimization techniques. 3. To explore advanced neural network models such as CNNs and RNNs. 4. To apply neural networks for solving real-world problems in areas like image processing and time series analysis. 5. To understand and implement techniques for regularization and improving model generalization.								
CO1	Understand the structure and functionality of basic and advanced neural network architectures.								
CO2	Implement training algorithms and optimization techniques for neural networks.								
CO3	Analyze and solve classification, regression, and sequential data problems using neural networks.								
CO4	Develop convolutional and recurrent neural networks for domain-specific applications.								
CO5	Apply techniques like regularization and dropout to enhance the performance and generalization of models.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Neural Networks:** • Basics of Neural Networks: Biological vs. Artificial Neural Networks • Perceptron Model: Single Layer Perceptron, Limitations of Perceptron • Activation Functions: Sigmoid, ReLU, Tanh, Softmax • Applications of Neural Networks in various domains	8	Understand the foundational concepts of neural networks and their applications.	**1,2,3**
II	**Multi-Layer Perceptron (MLP)** • Feedforward Neural Networks • Forward and Backpropagation Algorithm • Gradient Descent and Optimization Techniques • Role of Learning Rate and Hyperparameter Tuning	8	Analyze the working of multi-layer perceptrons and implement backpropagation for training neural networks.	**3,4**
III	**Regularization and Generalization:** • Overfitting and Underfitting in Neural Networks	8	Understand and apply techniques to improve the generalization of neural networks.	**3,4**

	• Regularization Techniques: L1, L2 Regularization, Dropout • Batch Normalization • Early Stopping and Cross-Validation			
IV	**Convolutional Neural Networks (CNNs):** • Basics of Convolutional Neural Networks • Convolution and Pooling Operations • Architectures of CNNs: LeNet, AlexNet, VGG, ResNet • Applications of CNNs in Image Processing	8	Develop convolutional neural network models for image-based tasks.	**4,5**
V	**Recurrent Neural Networks (RNNs) and Advanced Topics:** • Basics of Recurrent Neural Networks (RNNs) • Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) • Introduction to Generative Adversarial Networks (GANs) • Applications in Natural Language Processing and Time Series Prediction	8	Understand and implement RNNs for sequential data and explore advanced topics like GANs.	**3,4,5**

Practical Component				
Practical 1	Create a single-layer perceptron using Python or a deep learning framework (e.g., TensorFlow/PyTorch) to classify linearly separable data.	3	Understand the working of a perceptron and its limitations.	1,2 ,3
Practical 2	Implement and visualize the effects of different activation functions (Sigmoid, ReLU, Tanh, Softmax) on a neural network's performance.	3	Understand how activation functions influence the output of neural networks.	3,4
Practical 3	Manually implement forward and backpropagation for a simple neural network and verify the weight updates.	3	Analyze how neural networks learn using forward and backpropagation.	4,5
Practical 4	Build and train a multi-layer perceptron to classify the MNIST handwritten digits dataset.	3	Learn to train a multi-layer perceptron for real-world classification tasks.	3,5
Practical 5	Implement L1, L2 regularization, and dropout on a neural network and compare their effects on overfitting.	3	Understand and apply regularization techniques to improve generalization.	4,5
Practical 6	Implement batch normalization and experiment with different learning rates to observe their impact on convergence speed.	3	Analyze how batch normalization and learning rates affect training stability and performance.	4,5
Practical 7	Design and train a CNN on a dataset like CIFAR-10 or Fashion-MNIST for image classification.	3	Understand the working of CNN layers and their application in image processing.	3,5

Practical 8	Visualize the feature maps learned by a CNN for different layers to understand hierarchical feature extraction.	3	Explore how CNNs extract features at different levels of abstraction.using IBM Watson.	4,5
Practical 9	Build an RNN to perform sentiment analysis on text data or time series prediction on a simple dataset.	3	Learn how RNNs handle sequential data and temporal dependencies.	3,4
Practical 10	Train an LSTM model for text generation or stock price prediction using a sequential dataset.	3	Understand the capabilities of LSTMs in handling long-term dependencies in sequential data.	3,5

Textbooks

T1: Simon Haykin - "Neural Networks and Learning Machines", Pearson.
T2: Ian Goodfellow, Yoshua Bengio, and Aaron Courville - "Deep Learning", MIT Press, 2016.

Reference Books

R1: Charu C. Aggarwal - "Neural Networks and Deep Learning: A Textbook", Springer, 2018.
R2: Christopher Bishop - "Pattern Recognition and Machine Learning", Springer, 2006.

Other Learning Resources

O1: "Deep Learning Specialization" by Andrew Ng on Coursera.
O2: "Neural Networks and Deep Learning" by Geoffrey Hinton on Coursera.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the structure and functionality of basic and advanced neural network architectures.	PO1, PO2, PO3, PO4, PO10, PO12
2	Implement training algorithms and optimization techniques for neural networks.	PO1, PO2, PO3, PO4, PO10, PO12
3	Analyze and solve classification, regression, and sequential data problems using neural networks.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Develop convolutional and recurrent neural networks for domain-specific applications.	PO1, PO2, PO3, PO4, PO10, PO12
5	Apply techniques like regularization and dropout to enhance the performance and generalization of models.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4101R	**Neural Networks**	**CO 1**	2	2	1	3						1		2
		CO 2	2	2	1	3						1		2
		CO 3	3	3	3	3	3					1		3
		CO 4	3	3	3	3						1		3

		CO 5	3	3	3	3	3					1		3

SEMESTER – VII									
Course Title	**PE-I : Artificial Intelligence**								
Course code	**24BCSM4101R**	**Total credits:4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Basic Knowledge of Machine Learning	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. Describe the field of AI and its subfields machine learning, NLP and computer vision 2. Learn and implement key NLP algorithms and models, such as virtual agents, chatbots, sentiment classifiers, and language translators, to solve a variety of natural language understanding tasks. 3. Explore Practical Applications of Computer Vision**,** Investigate real-world applications of computer vision in various fields, including autonomous vehicles, surveillance systems, medical imaging, and augmented reality.								
CO1	Remember and understand relevance of AI in today's world.								
CO2	Gain an idea about Machine Learning application in real world problem.								
CO3	Develop an understanding deep learning concept in AI.								
CO4	Apply and Analyze data by applying various machine learning algorithms.								
CO5	Enhance problem-solving abilities with IBM Cloud service.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Artificial Intelligence:** Overview of Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL), Natural Language Processing(NLP), Computer Vision. Describe the types of AI, Explain what the main focus of AI is, List of practical applications of AI, Describe the eras of computing, Explain the difference between deterministic and probabilistic systems, Road Map of AI.	**8**	Gain a foundational understanding of Artificial Intelligence, its types, key focus areas, practical applications, historical development, and the differences between deterministic and probabilistic systems, along with an overview of the AI roadmap.	**1,2**
II	**Introduction to Machine Learning:** Introduction to Machine Learning, Supervised Learning & Unsupervised Learning. Supervised Learning Algorithms: Linear Regression, Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, KNN. Unsupervised Learning Algorithms: K-Means Clustering, Hierarchical Clustering.	**8**	Develop a fundamental understanding of Machine Learning, including supervised and unsupervised learning techniques, and gain practical experience with algorithms such as Linear Regression, Logistic Regression, SVM, Decision Trees, Random Forest, KNN, K-Means Clustering, and Hierarchical Clustering.	**1,2,4**
III	**Introduction to Deep Learning:** Introduction to Deep Learning:	**8**	Students will gain a comprehensive	**2,3,4,5**

	Neural Networks, Layers, and Activation Functions, Back Propagation, Optimizers, Deep Learning Algorithm: ANN, CNN, RNN, LSTM with Tensorflow and Keras, Introduction to Autoencoders, Generative Adversarial Networks (GANs), Restricted Boltzmann Machines (RBMs) and Applications.		introduction to Deep Learning, covering neural networks, activation functions, backpropagation, optimizers, and algorithms such as ANN, CNN, RNN, LSTM using TensorFlow and Keras, along with an understanding of Autoencoders, GANs, RBMs, and their applications in various domains.	
IV	**Introduction to Natural Language Processing and Computer Vision:** Introduction to Natural Language Processing, NLP application in real world, components of NLP, Virtual Agents overview, Chatbot and its parts, Tokenization, Stemming & Lemmatization in NLP, BagofWords,TF-IDF,Word2Vec, Word Embedding in NLP. Introduction to Computer Vision, Image Classification and Object Recognition, Feature Detection and Description algorithms with OpenCV.	**8**	Acquire foundational knowledge in Natural Language Processing (NLP) including its applications, components, and techniques like tokenization, stemming, TF-IDF, and Word Embedding, as well as in Computer Vision covering image classification, object recognition, and feature detection using OpenCV.	**2,3**
V	**Evolution from DeepQA to Watson services on IBM Cloud:** Explain the DeepQA architecture was; Explain why IBM decided to commercialize Watson and Describe the evolution of Watson services from the original DeepQA architecture to the present. Introduction to IBM Watson: explore Watson Studio, IBM's integrated environment for data scientists, developers, and domain experts, to build and deploy ML models. Creating Chatbots using Watson Assistant: learn the basics of Watson Assistant, including intents, entities, dialog flows, and integration capabilities. Natural Language Understanding (NLU), Language Translator, Text to Speech, Speech to Text on IBM Cloud.	**8**	Understand the evolution of IBM Watson from DeepQA to its current services on IBM Cloud, including Watson Studio for ML model development, Watson Assistant for creating chatbots, and capabilities like Natural Language Understanding, Language Translator, Text to Speech, and Speech to Text on the IBM Cloud platform.	**1,2,3**

Practical Component				
Practical 1	**Exploring AI Types and Applications :** Create a presentation or simulation to classify types of AI (Reactive, Limited Memory, Theory of Mind, Self-Aware) and demonstrate real-world applications in NLP, Computer Vision, and robotics.	**3**	Understand the classification and scope of AI. Explore real-world use cases across industries.	1,2 ,3
Practical 2	**Deterministic vs. Probabilistic Systems** Simulate a deterministic system (e.g., rule-based chatbot) and a probabilistic system (e.g., predictive text) using Python. Compare their behaviour.	**3**	Differentiate between deterministic and probabilistic systems. Analyze the strengths and limitations of each system.	4,5
Practical 3	**Implementing Supervised Learning Algorithms** Implement and compare Linear Regression, Logistic Regression, and Support Vector Machines on a dataset (e.g., Iris or Titanic dataset).	**3**	Understand supervised learning algorithms and apply them to real-world datasets. Analyze model performance using evaluation metrics.	2,3,4
Practical 4	**Clustering Algorithms for Unsupervised Learning:** Implement K-Means and Hierarchical Clustering algorithms on a dataset to group data points and visualize clusters.	**3**	Understand and apply unsupervised learning algorithms. Evaluate the quality of clustering using metrics like Silhouette Score.	3,5
Practical 5	**Building a Neural Network with TensorFlow** Build an Artificial Neural Network (ANN) for binary classification using TensorFlow and evaluate its performance using accuracy, precision, and recall.	**3**	Understand the structure and functionality of ANNs. Implement and optimize neural networks for specific tasks.	3,5
Practical 6	**Introduction to Natural Language Processing** Implement tokenization, stemming, lemmatization, and a Bag-of-Words model on a text dataset to preprocess and analyze textual data.	**3**	Understand fundamental NLP preprocessing techniques. Apply these techniques to clean and analyze textual data.	1,2,3
Practical 7	**Image Classification Using CNN** Build a Convolutional Neural Network (CNN) in TensorFlow to classify images from the CIFAR-10 dataset.	**3**	Understand the structure and functionality of CNNs. Train and evaluate CNNs for image classification tasks.	3,6
Practical 8	**Creating a Chatbot with Watson Assistant** Use IBM Watson Assistant to build a chatbot with intents, entities, and	**3**	Understand the components and functionality of chatbots.	3,6

	dialog flows. Test it with real-world queries.		Design and deploy a simple chatbot using IBM Watson.	
Practical 9	**Generative Adversarial Networks (GANs)** Objective: Build a simple GAN in TensorFlow to generate synthetic images from noise, using a dataset like MNIST.	**3**	Understand the architecture of GANs and their training process. Generate and evaluate synthetic data using GANs.	3,5
Practical 10	**Exploring Watson Services** Objective: Use IBM Watson Studio to deploy a machine learning model and integrate services like Speech-to-Text, Text-to-Speech, and Natural Language Understanding.	**3**	Explore IBM Watson's capabilities for AI solutions. Apply Watson services to build integrated AI applications.	3,6

TEXT BOOKS:

T1: Introduction to Data Science: B.Uma Maheswari& R. Sujata.

T2: Deep Learning with TensorFlow and Keras by Amita Kapoor, Antonio Gulli & Sujit pal

REFERENCE BOOKS:

R1: Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence" by Jon Krohn, Grant Beyleveld, and Aglaé Bassens.

R2: Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth.

OTHER LEARNING RESOURCES:

O1: Python & Machine Learning -https://www.w3schools.com/python/python_ml_getting_started.asp

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO-PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Remember and understand relevance of AI in today's world.	PO1, PO2, PO3, PO4, PO10, PO12
2	Gain an idea about Machine Learning application in real world problem.	PO1, PO2, PO3, PO4, PO10, PO12
3	Develop an understanding deep learning concept in AI.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Apply and Analyze data by applying various machine learning algorithms.	PO1, PO2, PO3, PO4, PO10, PO12
5	Enhance problem-solving abilities with IBM Cloud service.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4101R	**Artificial Intelligence**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VII									
Course Title	**Project-I**								
Course code	**24BCSM4102R**	**Total credits: 6**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 144	**0**	**0**	**0**	**16**	**12**	**0**	**6**
Pre-requisite	NIL	**Co-requisite**	NIL						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer / I semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. To enable students to perform comprehensive literature searches, critically review current knowledge, and synthesize information on advanced technical topics. 2. To provide students with hands-on experience in conducting detailed technical work, including theoretical studies, computer simulations, or hardware construction. 3. To improve students' ability to communicate their research findings and technical work effectively through progress reports, seminars, formal reports, and presentations.								
CO1	Apply IT principles for real-world problem-solving using programming languages and network solutions.								
CO2	Contribute to complex projects as a team member, demonstrating effective personal and team management.								
CO3	Develop programming skills to create and assess software, hardware, and network solutions.								
CO4	Apply personal and team management skills as a professional software developer.								
CO5	Cultivate employability skills and uphold professionalism in software development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a) theoretical studies b) computer simulations c) hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project; 4. Deliver a seminar on the general area of work being undertaken and specific contributions to that field;	**144**	Develop and Demonstrate comprehensive research, technical, and communication skills through literature review, technical implementation, progress documentation, and presentation.	**3,4,5,6**

	5. Prepare a formal report describing the work undertaken and results obtained so far; and Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4102R	**Project-I**	**CO 1**	2	2	1	2	2				2		2	
		CO 2	2	2	1	2	3				2		2	
		CO 3	1	2	1	2	2				3		3	
		CO 4	2	2	1	2	2				2		2	
		CO 5	1	1	1	2	1				1		3	

SEMESTER – VII									
Course Title	**PE-II: Data Mining and Analytics**								
Course code	**24BCSM4103R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**

		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/I semester of the fourth year of the programme**								
Course Objectives (Minimum 3)	This course is an introduction to data mining techniques, algorithms, and applications. Covers data pre-processing, pattern discovery, clustering, classification, and association rule mining. Emphasizes practical implementation and real-world data analysis.								
CO1	Understand diverse real-life data, mastering statistical foundations, and identifying challenges in data mining.								
CO2	Utilize Association Rule Mining to address practical problems, demonstrating understanding and proficiency in application.								
CO3	Apply Cluster analysis in unsupervised learning to real-world datasets, showcasing competence in data clustering techniques.								
CO4	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.								
CO5	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Data Mining:** What is data mining, applications of data mining, issues and challenges faced in data mining, types of data attributes – nominal, binary, ordinal, numeric, discrete and continuous, basic statistical descriptions of data – measures of central tendency and dispersion, major tasks in data pre-processing	**8**	Learning Outcome: Understand data mining concepts, applications, challenges, data attributes, statistical descriptions, and data pre-processing tasks for effective data analysis and decision-making.	**1, 2**
II	**Association Rule Mining:** Definitions of frequent itemset, maximal frequent itemset, closed frequent itemset, association rules, Frequent itemset generation – Apriori algorithm, DIC algorithm and FP-growth algorithm, Generating association rules from frequent item sets	**9**	Learning Outcome: Master association rule mining concepts including frequent itemsets, algorithms (Apriori, DIC, FP-growth), and generating association rules for data pattern discovery.	**3,4**
III	**DataClustering:** what is cluster analysis, need for cluster analysis, data matrix, dissimilarity matrix, proximity and dissimilarity measures for different data attribute types, partitioning based clustering methods, density-based clustering methods and hierarchical clustering methods, measuring cluster quality	**10**	Understand cluster analysis principles, dissimilarity measures, and various clustering methods for effective data segmentation and cluster quality evaluation.	**1, 2**
IV	**Prediction**: What is prediction, application areas of prediction, Simple linear regression, Multiple linear regression, Predictor Error	**8**	Master prediction concepts, including regression techniques, predictor error measures, and accuracy assessment	**2, 3**

	measures, Common techniques for assessing accuracy – holdout, random sampling, cross-validation, bootstrap		methods for effective predictive modeling and decision-making.	
V	**Classification**: what is classification, differences between classification and prediction, applications of classification, Some classification algorithms - Decision Trees, Naïve Bayes Classification, Logistic Regression and K-nearest Neighbour Classification, Confusion matrix and metrics for evaluating classifier performance, Overview of ensemble methods – Bagging and Boosting, Ensemble-based classifier - Random Forest	**10**	Understand classification principles, algorithms (Decision Trees, Naïve Bayes, Logistic Regression, K-nearest Neighbor), evaluation metrics, and ensemble methods for effective data classification.	**2, 3, 4**
Practical Component				
Practical 1	Explore what data mining is and its applications.	**3**	Understand data mining and its applications, foundational for data analysis.	**2**
Practical 2	Identify and classify types of data attributes (nominal, binary, ordinal, numeric, discrete, continuous).	**3**	Recognize various data attributes, crucial for data preprocessing.	**2**
Practical 3	Calculate measures of central tendency and dispersion for a dataset.	**3**	Apply statistical measures to summarize data, essential for data analysis.	**3**
Practical 4	Perform major tasks in data preprocessing, including handling missing data and normalization.	**3**	Execute data preprocessing techniques, key for data quality improvement.	**3**
Practical 5	Implement the Apriori algorithm to generate frequent itemsets and association rules.	**3**	Apply Apriori algorithm for association rule mining, critical for pattern discovery.	**3**
Practical 6	Use FP-Growth algorithm to generate frequent itemsets.	**3**	Develop frequent itemsets using FP-Growth, important for efficient pattern mining.	**3**
Practical 7	Perform K-means clustering on a dataset and analyze clusters.	**3**	Apply K-means clustering, essential for grouping similar data points.	**3**
Practical 8	Implement hierarchical clustering and analyze the resulting dendrogram.	**3**	Utilize hierarchical clustering to understand data structure, crucial for unsupervised learning.	**3**
Practical 9	Implement simple and multiple linear regression models for prediction.	**3**	Develop predictive models using linear regression, fundamental for forecasting.	**3**
Practical 10	Build a decision tree classifier and evaluate its performance.	**3**	Create classification models with decision trees,	**3**

			key for categorical data analysis.	

TEXT BOOKS:

T1: Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, India

T2: AK Pujari, Data Mining Techniques, University Press, India

REFERENCE BOOKS:

R1: Han, Manilla and Smyth, Principles of Data Mining, PHI, India

R2: Tan, Steinbach,Vipin Kumar, Introduction to Data Mining, Pearson Education Inc.

OTHER LEARNING RESOURCES:

O1: Coursera course like "Data Mining Specialization" covers fundamental data mining concepts, techniques, and applications.

O2: edX course like "Data Mining for Business Analytics" covers data mining techniques for business analytics, including data preprocessing, classification, prediction, clustering, association rules, and text mining.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand diverse real-life data, mastering statistical foundations, and identifying challenges in data mining.	PO1, PO2, PO3, PO4, PO10, PO12
2	Utilize Association Rule Mining to address practical problems, demonstrating understanding and proficiency in application.	PO1, PO2, PO3, PO4, PO10, PO12
3	Apply Cluster analysis in unsupervised learning to real-world datasets, showcasing competence in data clustering techniques.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.	PO1, PO2, PO3, PO4, PO10, PO12
5	Implement popular methodologies for prediction using Data Mining techniques, demonstrating practical application skills.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4103R	**Data Mining and Analytics**	**CO 1**	2	2	1	3						1		2
		CO 2	2	2	1	3						1		2
		CO 3	3	3	3	3	3					1		3
		CO 4	3	3	3	3						1		3
		CO 5	3	3	3	3	3					1		3

SEMESTER – V									
Course Title	**PE-II: Information Retrieval Systems**								
Course code	**24BCSM4103R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer/ I semester of the fourth year of the programme**								
Course Objectives (Minimum 3)	This course provides an in-depth introduction to the principles and practices of Information Retrieval (IR). Students will learn about the theory behind IR systems, the algorithms and techniques used to implement them, and the evaluation methods to measure their effectiveness. Topics covered include text processing, indexing, querying, ranking, and various IR models.								
CO1	Understand the fundamental concepts of Information Retrieval.								
CO2	Learn about various indexing and search techniques.								
CO3	Explore different IR models and their applications.								
CO4	Study the methods of evaluating IR systems.								
CO5	Gain hands-on experience with IR system implementation.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction to Information Retrieval** Course overview and objectives History of IR systems Applications of IR	8	Understand the fundamentals of information retrieval systems, including their history, applications, and objectives for effective information management and retrieval.	1, 2, 3
II	**Text Processing and Indexing** Text preprocessing: tokenization, stemming, and lemmatization Stopword removal and handling special characters Inverted index and its construction Index compression techniques	8	Master text processing techniques including tokenization, stemming, lemmatization, stopwords removal, inverted index construction, and index compression for efficient information retrieval systems.	2, 3, 4
III	**Retrieval Models** Boolean retrieval model Vector space model Probabilistic retrieval models	10	Understand retrieval models including Boolean, vector space, and probabilistic models for efficient information retrieval and relevance ranking in various contexts.	2, 3
IV	**Relevance Feedback and Query Expansion** Relevance feedback mechanisms Pseudo-relevance feedback Query expansion techniques	10	Master relevance feedback mechanisms, pseudo-relevance feedback, and query expansion techniques to enhance information retrieval precision and relevance.	3, 4, 5

V	**Evaluation of IR Systems** Evaluation metrics: precision, recall, F-measure, MAP, NDCG Test collections: TREC, Cranfield paradigm User studies and A/B testing	9	Evaluate information retrieval systems using metrics like precision, recall, F-measure, MAP, NDCG, test collections, and user studies for system improvement.	4, 5
Practical Component				
Practical 1	Review course objectives and explore the history of Information Retrieval (IR) systems.	3	Understand the fundamentals and evolution of IR systems.	2
Practical 2	Investigate various applications of IR in different fields.	3	Recognize the diverse applications of IR systems.	2
Practical 3	Implement tokenization, stemming, and lemmatization on a text dataset.	3	Apply text preprocessing techniques to prepare data for indexing.	3
Practical 4	Remove stopwords and handle special characters in a text dataset.	3	Clean text data by removing irrelevant components.	6
Practical 5	Construct an inverted index for a given text corpus.	3	Develop an inverted index to facilitate efficient search and retrieval.	3
Practical 6	Implement basic index compression techniques on an inverted index.	3	Compress index data to optimize storage and retrieval efficiency.	3
Practical 7	Implement Boolean, Vector Space, and Probabilistic retrieval models.	3	Apply various retrieval models to query and retrieve relevant documents.	3
Practical 8	Implement relevance feedback mechanisms and query expansion techniques.	3	Enhance search results using relevance feedback and query expansion.	3
Practical 9	Calculate precision, recall, F-measure, MAP, and NDCG for an IR system.	3	Evaluate the performance of IR systems using standard metrics.	5
Practical 10	Conduct user studies and perform A/B testing for an IR system.	3	Assess IR system effectiveness through user studies and A/B testing.	5

TEXT BOOKS:

T1: "Introduction to Information Retrieval" by Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze.

REFERENCE BOOKS:

R1: "Modern Information Retrieval: The Concepts and Technology behind Search" by Ricardo Baeza-Yates and Berthier Ribeiro-Neto.

OTHER LEARNING RESOURCES:

O1: TREC (Text REtrieval Conference) datasets and proceedings

O2: Online tutorials and courses (e.g., Coursera, edX)

O3: Software tools and libraries (e.g., Apache Lucene, Solr, Elasticsearch)

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental concepts of Information Retrieval.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
2	Learn about various indexing and search techniques.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
3	Explore different IR models and their applications.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Study the methods of evaluating IR systems.	PO1, PO2, PO3, PO4, PO5, PO10, PO12
5	Gain hands-on experience with IR system implementation.	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
24BCSM4103R	Information Retrieval Systems	CO1	2	2	1	1	1					1		2
		CO2	3	3	3	3	2					2		3
		CO3	3	2	3	3	3					2		3
		CO4	2	2	2	2	2					2		2
		CO5	2	2	3	2	2					2		3

SEMESTER – VII									
Course Title	**Summer Internship**								
Course code	**24BCSM4104R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 120	**0**	**0**	**0**	**16**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Summer / I semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. Equip students with practical experience in their field of study, ensuring they are prepared to transition smoothly into industry roles by mastering modern tools, technologies, and industry-standard practices.								

	2. Enhance students' soft skills, including presentation, interpersonal communication, and documentation, as well as office etiquette, to ensure they can effectively navigate and contribute to professional environments. 3. Foster a deep understanding of professional and ethical responsibilities in engineering, encouraging students to adhere to industry best practices, processes, and regulations. 4. Promote the ability to work effectively in teams, especially within multidisciplinary contexts, emphasizing collaboration, communication, and the integration of diverse perspectives and expertise. 5. Cultivate a professional work ethic characterized by productivity, consistency, and punctuality, preparing students to meet the demands and expectations of the industry.			
CO1	Apply modern tools and technologies in real-world scenarios to solve complex engineering problems.			
CO2	Analyse industry-standard reporting methods to produce technical documents adhering to professional guidelines.			
CO3	Develop solutions using design, development, and testing practices aligned with current industry standards.			
CO4	Demonstrate effective communication, teamwork, and interpersonal skills in a multi-disciplinary team environment.			
CO5	Evaluate professional and ethical responsibilities to make informed decisions in engineering practices.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	Method and Pedagogy: ● Summer internship shall be at least 90 hours during the summer vacation only. ● Department/Institute will help students to find an appropriate company/industry/organization for the summer internship. ● The student must fill up and get approved a Summer Internship Acceptance form by the company and provide it to the Coordinator of the department within the specified deadline. ● Students shall commence the internship after the approval of the department Coordinator. Summer internships in research centers is also allowed. ● During the entire period of internship, the student shall obey the rules and regulations of the company/industry/organization and also those of the University. ● Due to inevitable reasons, if the student will not able to attend the internship for few days with the permission of the supervisor, the department Coordinator should be informed via e-mail and these days should be compensated later. ● The student shall submit two documents to the Coordinator for the evaluation of the summer internship: a) Summer Internship Report b) Summer Internship Assessment Form ● Upon the completion of summer internship, a hard copy of “Summer Internship Report”	**120**	Complete a 90-hour summer internship, submit an internship report, and demonstrate practical industry experience through documentation and evaluation.	**3, 4, 5**

	must be submitted to the Coordinator by the first day of the new term. ● The report must outline the experience and observations gained through practical internship, in accordance with the required content and the format described in this guideline. Each report will be evaluated by a faculty member of the department on a satisfactory/unsatisfactory basis at the beginning of the semester. ● If the evaluation of the report is unsatisfactory, it shall be returned to the student for revision and/or rewriting. If the revised report is still unsatisfactory the student shall be requested to repeat the summer internship.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply modern tools and technologies in real-world scenarios to solve complex engineering problems.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
2	Analyse industry-standard reporting methods to produce technical documents adhering to professional guidelines.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
3	Develop solutions using design, development, and testing practices aligned with current industry standards.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
4	Demonstrate effective communication, teamwork, and interpersonal skills in a multi-disciplinary team environment.	PO1, PO2, PO3, PO4, PO5, PO6, PO9, PO10, PO12
5	Evaluate professional and ethical responsibilities to make informed decisions in engineering practices.	PO1, PO2, PO3, PO4, PO5, PO6, PO8, PO9, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM4104R	Summer Internship	CO 1	3	3	3	3	3	3			2	1		2
		CO 2	2	2	3	3	2	2			2	1		2
		CO 3	2	2	3	2	2	2			2	1		2
		CO 4	3	3	3	3	3	3			2	3		2

		CO 5	2	2	2	2	2	2		3	2	1		2

SEMESTER – VIII									
Course Title	**Cryptography & Network Security**								
Course code	**24BCSM4201R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 36	**3**	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	4. To provide deeper understanding into cryptography, its application to network security, threats/vulnerabilities to networks and countermeasures. 5. To explain various approaches to Encryption techniques, strengths of Traffic Confidentiality, Message Authentication Codes. 6. To familiarize Digital Signature Standard and provide solutions for their issues.								
CO1	Identify basic security attacks and services.								
CO2	Use symmetric and asymmetric key algorithms for cryptography.								
CO3	Design a security solution for a given application.								
CO4	Analyze Key Management techniques and the importance of number Theory.								
CO5	Understanding of Authentication functions how Message Authentication Codes and Hash Functions work.								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**INTRODUCTION**: Security trends, The OSI Security Architecture, Security Attacks, Security Services, and Security Mechanisms, A model for Network security. CLASSICAL ENCRYPTION TECHNIQUES: Symmetric Cipher Modes, Substitute Techniques, Transposition Techniques, Rotor Machines, Stenography.	7	To understand different security services and security mechanisms and different classical encryption techniques	1,2,3
II	**BLOCK CIPHER AND DATA ENCRYPTION STANDARDS:** Block Cipher Principles, Data Encryption Standards, the Strength of DES, Differential and Linear Crypt Analysis, Block Cipher Design Principles. ADVANCED ENCRYPTION STANDARDS: Evaluation Criteria for AES, the AES Cipher. MORE ON SYMMETRIC CIPHERS: Multiple Encryption, Triple DES, Block Cipher Modes of Operation, Stream Cipher and RC4.	7	Understanding block cipher and data encryption standards	3,4,5

III	**PUBLIC KEY CRYPTOGRAPHY AND RSA:** Principles Public key crypto Systems, Diffie Hellman Key Exchange, the RSA algorithm, Key Management, Elliptic Curve Arithmetic, Elliptic Curve Cryptography. MESSAGE AUTHENTICATION AND HASH FUNCTIONS: Authentication Requirement, Authentication Function, Message Authentication Code, Hash Function, Security of Hash Function and MACs. HASH AND MAC ALGORITHM: Secure Hash Algorithm, Whirlpool, HMAC, CMAC. DIGITAL SIGNATURE: Digital Signature, Authentication Protocol, Digital Signature Standard.	7	Analyzing the concepts of Public key cryptography and RSA.	3,4,5
IV	**AUTHENTICATION APPLICATION:** Kerberos, X.509Authentication Service, Public Key Infrastructure. EMAILSECURITY: Pretty Good Privacy (PGP) and S/MIME. IPSECURITY: Overview, IP Security Architecture, Authentication Header, Encapsulating Security Pay load, Combining Security Associations and Key Management.	7	Analyzation of the concepts of Authentication Application, Email Security and IP Security.	3,5
V	**WEB SECURITY:** Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET), Intruders, Viruses and related threats. FIREWALL: Firewall Design principles, Trusted Systems	7	Analyzation of Web Security and firewall.	3,4,5

TEXT BOOKS:

T1: William Stallings (2006), Cryptography and Network Security: Principles and Practice,4th edition, Pearson Education, India.

T2: William Stallings (2000), Network Security Essentials (Applications and Standards), Pearson Education, India.

REFERENCEBOOKS:

R1: Charlie Kaufman (2002), Network Security: Private Communication in a Public World, 2nd edition, Prentice Hall of India, New Delhi.

R2: Atul Kahate (2008), Cryptography andNetworkSecurity,2nd edition, Tata McGraw hill, India.

R3: Robert Bragg, Mark Rhodes (2004), Network Security: The complete reference, Tata McGraw-Hill, India.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Identify basic security attacks and services.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
2	Use symmetric and asymmetric key algorithms for cryptography.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
3	Design a security solution for a given application.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
4	Analyze Key Management techniques and the importance of number Theory.	CO1,CO2,CO3,CO4,CO5,CO10,CO12
5	Understanding of Authentication functions the manner in which Message Authentication Codes and Hash Functions work.	CO1,CO2,CO3,CO4,CO5,CO10,CO12

MAPPING TABLE

Course code	Course Name		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
24BCSM4201R	Cryptography & Network Security	CO1	2	2	1	1	1					1		2
		CO2	3	3	3	3	2					2		3
		CO3	3	2	3	3	3					2		3
		CO4	2	2	2	2	2					2		2
		CO5	2	2	3	2	2					2		3

SEMESTER – VIII									
Course Title	**Machine Learning**								
Course code	**24BCSM4202R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. To provide fundamental theoretical foundations knowledge of various learning algorithms 2. Apply suitable machine learning techniques for data handling and knowledge extraction 3. To gain the fundamental knowledge and understand the context of supervised and unsupervised learning through real-life examples 4. Evaluate the performance of algorithms based on corresponding metrics identified and to provide solutions for various real world applications								
CO1	Recognize the characteristics of machine learning strategies								
CO2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations								
CO3	Analyse and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns								
CO4	Understand the concept of ensemble learning and its applications								
CO5	Evaluate the performance of various algorithms								

Unit-No.	Content	Contact Hour	Learning Outcome	BL
I	**Introduction to Machine Learning:** Machine Learning introduction, Types of Machine Learning: Supervised, Unsupervised, and Reinforcement Learning, Applications of Machine Learning in Various Fields	7	Gain knowledge of Machine Learning fundamentals, including types (supervised, unsupervised, reinforcement), and applications across diverse fields for practical implementation.	**1, 2**
II	**Data Preprocessing and Exploratory Data Analysis** Data Collection and Cleaning, Handling Missing Values, Feature Scaling and Normalization, Data Visualization Techniques, Exploratory Data Analysis (EDA), statistical measures, Basic tools (plots, graphs and summary statistics) of EDA	8	Master data pre-processing techniques including collection, cleaning, handling missing values, feature scaling, normalization, visualization, and exploratory data analysis tools.	**3, 4**
III	**Supervised Learning and Unsupervised Learning** Linear and Non-Linear examples – Multi– Class & Multi-Label classification, Linear Regression, Multiple Linear Regression – Naïve Bayes Classifier – Decision	12	Demonstrate proficiency in supervised learning (regression, classification) and unsupervised learning (clustering, dimensionality reduction) algorithms for data	**3, 4**

	Trees – ID3 – CART, K-NN classifier – Logistic regression – Perceptron – Single layer & Multi-layer – Support, Clustering basics (Partitioned, Hierarchical and Density based) - K- Means clustering – K-Mode clustering – Self organizing maps, Expectation maximization – Principal Component Analysis		analysis and modelling tasks.	
IV	**Ensemble Learning** Bagging-Committee Machines and Stacking-Boosting-Ranking based aggregation-based aggregation	8	Apply ensemble learning techniques such as bagging, boosting, stacking, and ranking for improving predictive model accuracy and robustness.	**3, 4, 5**
V	**Model Evaluation and Optimization** Model Validation Techniques: Train/Test Split, Cross-Validation, Performance Metrics, Confusion Matrix, Accuracy, Precision, Recall, F1-Score	10	Effectively evaluate machine learning models using validation techniques like train/test split, cross-validation, and performance metrics (accuracy, precision, recall, F1-score).	**3, 5**
Practical Component				
Practical 1	Collect and clean a real-world dataset, handle missing values, and apply feature scaling and normalization techniques.	3	Understand the importance of data preprocessing and apply cleaning techniques to prepare data for machine learning models.	1,2,3
Practical 2	Perform EDA on a dataset using statistical measures and visualization techniques such as histograms, scatter plots, box plots, and correlation matrices.	3	Analyze and visualize data to uncover patterns, trends, and relationships.	3,4
Practical 3	Implement linear regression for a continuous target variable and logistic regression for binary classification tasks.	3	Understand and implement regression models for predictive analysis.	3,4
Practical 4	Build and evaluate classification models using Decision Trees (e.g., ID3, CART) and K-Nearest Neighbors (K-NN).	3	Develop classification models and compare their performance on given datasets.	4,5
Practical 5	Perform clustering using K-Means, K-Mode, and hierarchical clustering methods. Evaluate the clusters using metrics such as Silhouette Score.	3	Group data points into clusters and analyze their structure and patterns.	3,4
Practical 6	Apply PCA to reduce the dimensionality of a high-dimensional dataset and visualize the transformed data.	3	Understand and implement dimensionality reduction techniques to simplify data while retaining essential features.	4,5

Practical 7	Implement ensemble models such as Bagging, Boosting (e.g., AdaBoost, Gradient Boosting), and Stacking on a classification dataset.	3	Combine multiple models to improve performance and analyze the advantages of ensemble learning.	4,5
Practical 8	Implement a single-layer perceptron and extend it to a multi-layer perceptron (MLP) using a library like TensorFlow or PyTorch.	3	Understand the structure and working of perceptrons and neural networks for classification tasks.	3,4
Practical 9	Evaluate classification models using metrics such as confusion matrix, accuracy, precision, recall, and F1-score. Perform cross-validation.	3	Assess the performance of machine learning models using evaluation metrics.	5
Practical 10	Perform hyperparameter tuning using techniques such as Grid Search and Random Search to optimize machine learning models.	3	Optimize machine learning models to achieve better performance.	5,6

TEXT BOOKS:

T1: S. Shalev-Shwartz, S.Ben-David, "Understanding Machine Learning: From Theory to Algorithms", Cambridge University Press, 2014.

T2: Ethem Alpaydin,"Introduction to Machine Learning", MIT Press, Prentice Hall of India,

REFERENCE BOOKS:

R1: Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning", MIT Press, 2nd Edition, 2018.

R2: Tom Mitchell, "Machine Learning", McGraw Hill, 3rd Edition, 1997.

R3: Data Mining: Concepts and Techniques", Third Edition, 2 Jiawei Han, Micheline Kamber and Jian Pei, ISBN 0123814790,(2011).

R4: Duda, Richard, Peter Hart, and David Stork, "Pattern Classification," 2nd Edition, John Wiley & Sons, Hoboken, 2000.

OTHER LEARNING RESOURCES:

O1: Coursera provides course like "Machine Learning" that covers a broad introduction to machine learning, data mining, and statistical pattern recognition.

O2: edX provides course like "Machine Learning" that covers the essentials of machine learning, focusing on supervised and unsupervised learning, mathematical and heuristic aspects, and hands-on implementation of algorithms.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Recognize the characteristics of machine learning strategies	PO1, PO2, PO3, PO4, PO10, PO12

2	Demonstrate proficiency in data analytics and apply advanced tools to work on dimensionality reduction and mathematical operations	PO1, PO2, PO3, PO4, PO10, PO12
3	Analyze and Apply the suitable supervised learning and unsupervised learning methods for real-world problems and for handling unknown patterns	PO1, PO2, PO3, PO4, PO5, PO10, PO12
4	Understand the concept of ensemble learning and its applications	PO1, PO2, PO3, PO4, PO10, PO12
5	Evaluate the performance of various algorithms	PO1, PO2, PO3, PO4, PO5, PO10, PO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4202R	**Machine Learning**	**CO 1**	2	2	1	3						1		2
		CO 2	2	2	1	3						1		2
		CO 3	3	3	3	3	3					1		3
		CO 4	3	3	3	3						1		3
		CO 5	3	3	3	3	3					1		3

SEMESTER – VIII									
Course Title	**Image Processing and Pattern Recognition**								
Course code	**24BCSM4203R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	Nil	**Co-requisite**	Nil						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives	1. To provide an in-depth understanding of the fundamentals of digital image processing, including image representation, enhancement, and transformation								

(Minimum 3)	techniques. 2. To equip students with the knowledge of image segmentation and compression methods for efficient image analysis and storage. 3. To introduce the principles of feature extraction and pattern recognition, focusing on classifiers and dimensionality reduction techniques. 4. To explore the applications of machine learning and deep learning models in image classification and object detection tasks. 5. To enable students to design and implement real-world projects involving image processing and pattern recognition, such as face recognition or medical image analysis.			
CO1	Understand spatial filtering techniques to enhance digital images in the spatial domain effectively.			
CO2	Evaluate the impact of Fourier Transform properties on image enhancement in the frequency domain.			
CO3	Compare and contrast error-free and lossy image compression models for efficient data storage.			
CO4	Implement adaptive thresholding techniques for image segmentation and edge detection in practice.			
CO5	Understand the core concepts and principles of pattern recognition.			
Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	**Introduction** Digital image representation, Fundamentals steps in Image Processing, Elements of DIP systems. **Digital Image Fundamentals** Elements of Visual Perception, Sampling and Quantization, Relationships between pixels, Linear and Nonlinear operations.	8	To understand digital image fundamentals.	1,2
II	**Image Enhancement in Spatial domain** Enhancement by Point Processing, Histogram Processing, Spatial Filtering.. **Image Enhancement in Frequency Domain** Introduction to the Fourier Transform, The discrete Fourier Transform, Properties of the two-dimensional Fourier Transform, Smoothing Frequency-domain filters, Sharpening Frequency domain filters.	8	Understanding Image Enhancement in Spatial domain and Frequency Domain.	3,4
III	Image Compression Fundamentals, Image Compression Models, Error Free Compression, Lossy Compression.	8	Analyzing the concepts of Image Compression Fundamentals.	3,4,5

IV	Image Segmentation Threshold Techniques: Global, Adaptive and Optimum thresholding, Edge detection, Region Growing.	8	Analyzation of the concepts of Image Segmentation Threshold Techniques.	3,5
V	**Pattern Recognition** Definition and scope of pattern recognition History and applications of pattern recognition Basic concepts: patterns, features, and classifiers Overview of different approaches: statistical, syntactic, and neural Pattern recognition system design cycle	8	Analyzation of Pattern Recognition.	4,5,6

Practical Component				
Practical 1	**Implementing Basic Image Enhancement Techniques:** Perform contrast stretching and histogram equalization on grayscale and color images using Python/OpenCV.	**3**	Understand and apply basic image enhancement techniques to improve image quality.	1,2,3
Practical 2	**Spatial and Frequency Domain Filtering:** Apply spatial filters (e.g., smoothing, sharpening) and frequency domain filters (e.g., low-pass, high-pass) to enhance image features.	**3**	Analyze and enhance images using spatial and frequency domain filters.	3,4
Practical 3	**Edge Detection and Morphological Operations:** Implement edge detection techniques (Sobel, Canny) and morphological operations (dilation, erosion, opening, and closing) on binary and grayscale images.	**3**	Identify edges and refine image structure using morphological techniques.	3,4
Practical 4	**Image Segmentation Techniques:** Perform image segmentation using thresholding (global, adaptive, and Otsu's method) and region-based methods (region growing).	**3**	Segment images into meaningful regions for further analysis.	4,5
Practical 5	**Feature Extraction (Shape, Texture, and Color):** Extract and visualize shape, texture (e.g., GLCM), and color features from images using Python/OpenCV.	**3**	Understand and implement feature extraction techniques for image representation.	3,4
Practical 6	**Dimensionality Reduction with PCA and LDA:** Implement Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) to	**3**	Reduce the dimensionality of datasets for computational efficiency in pattern recognition tasks.	4,5

	reduce feature dimensions of image datasets.			
Practical 7	**Classifier Development for Pattern Recognition:** Develop and evaluate classifiers (e.g., SVM, k-NN) for pattern recognition tasks such as digit or character recognition.	**3**	Train and evaluate classifiers for recognizing patterns in image datasets.	4,5
Practical 8	**Convolutional Neural Networks (CNNs) for Image Classification:** Implement CNNs for image classification tasks using TensorFlow/Keras, and visualize feature maps.	**3**	Develop CNN models to classify images and understand their underlying mechanics.	3,5
Practical 9	**Transfer Learning Using Pre-trained Models:** Apply transfer learning using pre-trained models like VGG16 or ResNet to classify images in custom datasets.	**3**	Utilize pre-trained models for efficient and accurate image classification.	3,5
Practical 10	**Object Detection and Real-Time Processing:** Develop an object detection system using YOLO or Faster R-CNN, and test it on real-time video feeds.	**3**	Design and implement an object detection system for practical applications.	5,6

TEXT BOOKS:

T1: Rafael C Gonzalez, Richard E Woods,―Digital Image Processing, Pearson Education Publications.

T2: Rajjan Shinghal - Pattern Recognition, Oxford Publications.

T3: Chanda and Majumder,―Digital Image Processing and Analysis, Prentice Hall Publications.

REFERENCE BOOKS:

R1: Rafael C Gonzalez, Richard E Woods,―Digital Image Processing with Matlab, Pearson Education Publications.

R2: S. Sridhar,―Digital Image Processing, Oxford University Press.

R3: Jayaraman,―Digital Image Processing, McGraw Hill.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand spatial filtering techniques to enhance digital images in the spatial domain effectively.	CO1,CO2,CO3, CO5,CO10,CO12
2	Evaluate the impact of Fourier Transform properties on image enhancement in the frequency domain.	CO1,CO2,CO3, CO5,CO10,CO12

3	Compare and contrast error-free and lossy image compression models for efficient data storage.	CO1,CO2,CO3, CO5,CO10,CO12
4	Implement adaptive thresholding techniques for image segmentation and edge detection in practice.	CO1,CO2,CO3, CO4, CO5,CO10,CO12
5	Understand the core concepts and principles of pattern recognition.	CO1,CO2,CO3, CO4, CO5,CO10,CO12

MAPPING TABLE

Course code	**Course Name**		**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
24BCSM4203R	**Image Processing and Pattern Recognition**	**CO1**	2	2	1	3						1		2
		CO2	2	2	1	3						1		2
		CO3	3	3	3	3	3					1		3
		CO4	3	3	3	3						1		3
		CO5	3	3	3	3	3					1		3

SEMESTER – VIII									
Course Title	**Project-II**								
Course code	**24BCSM4204R**	**Total credits: 6**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 144	**0**	**0**	**0**	**12**	**16**	**0**	**6**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Bachelor of Technology in Computer Science and Engineering (Data Science & Artificial Intelligence)**								
Semester	**Winter/ II semester of the fourth year of the Programme**								
Course Objectives (Minimum 3)	1. To equip students with the IT principles and programming skills. 2. To equip students with team management abilities necessary to develop and implement software, hardware, and network solutions effectively. 3. To cultivate employability skills and professionalism in software development.								
CO1	Apply IT principles for real-world problem-solving using programming languages and network solutions.								
CO2	Manage complex projects as a team member, demonstrating effective personal and team management skills.								
CO3	Develop programming skills to create and assess software, hardware, and network solutions.								
CO4	Apply personal and team management skills as a professional software developer.								
CO5	Cultivate employability skills and uphold professionalism in software development.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**BL**
I	This course will be conducted largely as an individual or small group project under the direct supervision of a member of academic staff. The specific project topic undertaken will reflect the common interests and expertise of the student(s) and supervisor. Students will be required to: 1. Perform a literature search to review current knowledge and developments in the chosen technical area; 2. Undertake detailed technical work in the chosen area using one or more of: a) theoretical studies b) computer simulations c) hardware construction; 3. Produce progress reports or maintain a professional journal to establish work completed, and to schedule additional work within the time frame specified for the project;	144	To understand, analyze, manage, develop and cultivate programming skills to create and assess software, hardware, and network solutions.	1,2,3,4,5

	4. Deliver a presentation on the general area of work being undertaken and specific contributions to that field; 5. Prepare a formal report describing the work undertaken and results obtained so far. 6. Present the work in a forum involving poster presentations and demonstrations of operational hardware and software.			

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply IT principles for real-world problem-solving using programming languages and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
2	Contribute to complex projects as a team member, demonstrating effective personal and team management.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
3	Develop programming skills to create and assess software, hardware, and network solutions.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
4	Apply personal and team management skills as a professional software developer.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12
5	Cultivate employability skills and uphold professionalism in software development.	PO1, PO2, PO3, PO4, PO5, PO6, PO10, PO12

MAPPING TABLE

Course code	Course Name		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
24BCSM4204R	Project-II	CO 1	2	2	2	1	2	2				2		2
		CO 2	2	2	2	1	2	3				2		2
		CO 3	1	1	2	1	2	2				3		3
		CO 4	2	2	2	1	2	2				2		2
		CO 5	1	1	1	1	2	1				1		3

ASSAM DOWN TOWN UNIVERSITY

Curriculum and Syllabus

Master of Computer Application
on
Artificial Intelligence and Deep Learning

OUTCOME BASED EDUCATION FRAMEWORK
CHOICE BASED CREDIT SYSTEM
Version: 1.0

FACULTY OF COMPUTER TECHNOLGY

July, 2024

PREAMBLE

Assam down town University is a premier higher educational institution which offers Bachelor, Master, and Ph.D. degree programmes across various faculties. These programmes, collectively embodies the vision and mission of the university. In keeping with the vision of evolutionary changes taking place in the educational landscape of the country, the university has restructured the course curriculum as per the guidelines of National Education Policy 2020. This document contains outline of teaching and learning framework and complete detailing of the courses. This document is a guidebook for the students to choose desired courses for completing the programme and to be eligible for the degree. This volume also includes the prescribed literature, study materials, texts, and reference books under different courses as guidance for the students to follow.

Recommended by the Board of Studies (BOS) meeting of the Faculty of Computer Technology held on dated 15/07/2024 and approved by the 51st Academic Council (AC) meeting held on dated 26/07/2024.

Chairperson, Board of Studies

Member Secretary, Academic Council

Vision

To become a Globally Recognized University from North Eastern Region of India, dedicated to the Holistic Development of Students and Making Society Better

Missions

1. Creation of curricula that address the local, regional, national, and international needs of graduates, providing them with diverse and well-rounded education.
2. Build a diverse student body from various socio-economic backgrounds, provide exceptional value-based education, and foster holistic personal development, strong academic careers, and confidence.
3. Achieve high placement success by offering students skill-based, innovative education and strong industry connections.
4. Become the premier destination of young people, desirous of becoming future professional leaders through multidisciplinary learning and serving society better.
5. Create a highly inspiring intellectual environment for exceptional learners, empowering them to aspire to join internationally acclaimed institutions and contribute to global efforts in addressing critical issues, such as sustainable development, Climate mitigation and fostering a conflict-free global society.
6. To be renowned for creating new knowledge through high quality interdisciplinary research for betterment of society.
7. Become a key hub for the growth and excellence of AdtU's stakeholders including educators, researchers and innovators
8. Adapt to the evolving needs and changing realities of our students and community by incorporating national and global perspectives, while ensuring our actions are in harmony with our foundational values and objectives of serving the community.

Programme Details

Programme Overview

Master of Computer Application is a Post Graduate Degree awarded for the programme in Computer Technology. Being one of the top universities, we want to establish a unified educational environment that incorporates the newest technical advancements and meets market demands. To give students the greatest technical education possible, the FoCT has partnerships with leading academic and research institutions across the world and focuses on the emerging areas of computer technology such as Artificial Intelligence, Machine Learning, Deep Learning, Cyber Security, Data Science, Augmented Reality, Virtual Reality, etc. Deserving students can apply for scholarships.

I. **Specific Features of the Curriculum** Master of Computer Application curriculum has been designed as per the latest requirements of Industry Employability.

II. Eligibility Criteria:

To pursue the MCA (Master of Computer Application) course, the candidates will have to fulfil the given below eligibility criteria:

- The candidate must have passed BCA/B.Sc. (Computer Science)/B.Tech (IT)/ B.E. (CSE)/B.Tech (CSE)/B.E. (IT)/ B.Tech. (IT) or equivalent Degree;

or

- Passed any graduation degree B.Sc/B.Com. / B.A./ B.Voc preferably with Mathematics at 10+2 level or at Graduation level.

Note: Candidate having graduation without mathematics must undertake and clear additional bridge courses.

III. Program Educational Objectives(PEOs):

PEO-1: To produce post graduates with a two-year professional education in Computer Applications with technical, professional, and communications skills.

PEO-2: To encourage effective software development and heighten technological awareness among the students, so that they become the future specialists in the industry.

PEO-3: To encourage students to learn, engage and update themselves to carryout independent or collaborative research so that they can address technological challenges in their corresponding field of expertise.

IV. Program Specific Outcomes(PSOs):

PSO1: Ability to demonstrate a degree of mastery in computer applications.

PSO2: Ability to independently carry out research/investigation and developmental work to solve practical problems.

PSO3: Develop sound knowledge and skill sets in fields related to human-computer interaction and management of industrial processes for the design and implementation of intelligent systems.

V. Program Outcomes(POs): (8-12)

PO1: Computational knowledge: Apply the knowledge of computing fundamentals, computing specialisation, and mathematics to solve real world problems.

PO2: **Problem analysis:** Identify, formulate, review literature, and analyse to solve complex problems using knowledge of mathematics and computer science.

PO3: Design/development of solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet with the needs of appropriate public health and safety, cultural, societal, and environmental scenarios.

PO4: **Conduct investigations of complex computing problems**: Use research-based knowledge and methods to design experiments, analyse and interpret data and perform information synthesis so as to provide valid conclusions.

PO5: Modern tool usage: Create, select, adapt, and apply appropriate techniques, resources, and modern tools for solving complex computing problems, along with an understanding of their limitations.

PO6: Individual and team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

PO7: Professional ethics: Commit to professional ethics, responsibilities, and norms of computing practices.

PO8: Environment and sustainability: Understand and demonstrate the impact of the professional solutions in societal and environmental contexts and need for sustainable development.

PO9: Communication efficacy: Communicate effectively on complex technical activities with the computing community, and with society at large.

PO10. Innovation and entrepreneurship: Using innovation to identify opportunity and to create value and wealth for the betterment of the individual and society as a whole.

PO11. Project management and finance: Demonstrate knowledge and understanding of software engineering and project management principles and apply these to one's work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning for continual development as a computing professional.

VI. Total Credits to be Earned:

Summary of Credits	
Semester	**Credit**
1	24
2	26
3	28
4	18
TOTAL	**96**

VII. Career Prospects:

The career prospects for MCA graduates are generally good, as there is a high demand for IT professionals in India and around the world. The MCA degree provides you with a strong foundation in computer science and software development, which can open doors to a variety of career opportunities.

The following are the career prospects of the MCA courses:

Software Developer: This is one of the most common career paths for MCA graduates. Software developers design, develop, and test software applications. They may work for a variety of companies, including software development firms, IT consulting firms, and large corporations.

Data Scientist: Data scientists are responsible for collecting, analysing, and interpreting data to solve business problems. They are in high demand across a variety of industries, including finance, healthcare, and retail.

Cloud Architect: Cloud architects design and build cloud-based solutions for businesses. They need to have a strong understanding of cloud computing technologies and how to use them to meet the needs of their clients.

Web Developer: Web developers design and develop websites and web applications. They may work for a variety of companies, including web development firms, marketing agencies, and e-commerce businesses.

IT Project Manager: IT project managers are responsible for planning, executing, and monitoring IT projects. They need to have strong technical skills as well as strong project management skills.

Cybersecurity Specialist: Cybersecurity specialists are responsible for protecting computer systems and networks from cyberattacks. They need to have a strong understanding of cybersecurity threats and how to prevent them.

The salary for MCA graduates can vary depending on their experience, skills, and location. However, MCA graduates typically earn more than graduates with other degrees.In addition to the career options listed above, MCA graduates can also work as system analysts, database administrators, network engineers, and quality assurance testers.

EVALUATION METHODS

The student performance shall be evaluated through In-semester (Sessional) and semester-end examinations. A weightage of 40% or as prescribed by the programme shall be added to the score of the end semester examination.

A. INTERNAL ASSESSMENT:

The teacher who offers the course shall be responsible for internal assessment by conducting in-semester (sessional) examination and evaluating the performance of the students pursuing that course. The components for internal assessment are illustrated in the table given below.

SN	Components/ Examinations	Marks Allotted
1.	In-Sem Exam – I (ISE-I) (Written Examination)*	30
2.	In-Sem Exam – II (ISE-II) (Written Examination)*	30
3.	Assignment	10
4.	Presentation (SP)	10
5.	Quiz	5
6.	Class Performance based score*	5

are compulsory

Note: *Total Internal assessment should be out of 40*

INSTRUCTION

1. If a student fails to appear in the any of the component without any valid reason he/she shall be marked zero in that component. However, the course teacher at his discretion may arrange for the missed test on an alternate date for the absentee students after determining ground with genuine/valid reasons for the absent.

2. The report of evaluation of an activity towards the in-semester (sessional) component of a course shall be duly notified by the concerned course teacher within a week of completion.

3. The program coordinators should upload the in-semester marks to the ERP and forward acknowledgement of all the courses of the program to the Controller of Examinations before the start of the End-semester examination.

B. SEMESTER END EXAMINATION:

Time table for end semester examination is published at least 25 days prior to the start of Examination.

I. Pre-Examination:

Eligibility Criteria for a student to appear in University Examinations:

The student shall only be allowed to appear in a University Examination, if:

i) He/ She is a registered student of the University;

ii) He/ She is of good conduct and character;

iii) He/ She has completed the prescribed Programme of study with minimum percentage of attendance as laid down in the Regulations of the Programme concerned.

Under special cases, a student may be allowed to appear for an examination without being registered in the University but the result of the said student will be kept on hold till the registration of the concerned student is completed.

II. Admit Card:

Admit card for the examination may be downloaded through ERP where the system will generate a Unique ID Cards through online.

The University shall have the right to cancel admission for examination of any candidate on valid grounds.

III. Pattern of Question Papers:

The question paper shall follow the principles of Bloom's Taxonomy.

Table

S. N.	Level	Questions /verbs for test
1	Remember	List, Define, tell, describe, recite, recall, identify, show who, when, where, etc.
2	Understand	Describe, explain, contrast, summarize, differentiate, discuss etc.
3	Apply	Predict, apply, solve, illustrate, determine, examine, modify
4	Analyze	Classify, outline, categorize, analyze, diagrams, illustrate, infer, etc.
5	Evaluate	Assess, summarize, choose, evaluate, recommend, justify, compare etc.
6	Create	Design, Formulate, Modify, Develop, integrate, etc.

Note: No course is to be evaluated on basis of **all 6 knowledge levels.**

The format of the question paper across all the programmes follow a unique pattern and the total marks is 60

Table 1: Question paper pattern for End semester examination

Sl no	Question pattern	Total marks
1	MCQs (10 Questions)	10
2	2 Marks questions (10 Questions)	20
3	4 Marks questions (5 Questions)	20
4	10 Marks questions (1 Question)	10

IV. Examination Duration:

Each paper of 60 marks shall ordinarily be of two hours duration.

V. Practical Examinations, Viva-Voce etc.:

i) Practical examination shall be conducted in the presence of one external expert and one or more internal examiners.

ii) Viva-Voce, Oral examinations of the Project report, Dissertation etc. shall be undertaken by a Board of Examiners constituted by the respective Dean of Program with the advice of Supervisor(s).

VI. Procedure of Expulsion:

If any candidate is found to be using any unfair-means during the examination, the invigilator may cease his/her answer sheet and report it directly to the Officer-in-Charge. The Office-in-Charge of the center may take appropriate decisions as per the rules and procedure of the examination. The Officer-in-Charge may allow the students to write the exam with new answer sheet or may expel the student from appearing the paper depending on the nature of unfair-means. In case of Computer based test, the students may be directed to write an apology letter and sign in the prescribe expulsion form. The student may not be allowed to write that examination.

VII. Instruction to the Students:

(i) The students shall not bring to the Examination Hall, any electronic gadget used as a means of communication or record except electronic calculator, if required.

(ii) The students shall not receive any book or printed or hand written or photo copy (Xerox) or blank-paper from any other person while he/she is in the examination-

room or in laboratory or in any other place to which he/she is allowed to have access during course of examination.

(iii) The students shall not communicate with any other candidate in the examination room or with any other person in and outside the examination-room.

(iv) The students shall not see, read or copy anything written by any other candidate, nor shall he/she knowingly or negligently permit any other candidate to see, read or copy anything written by him/her or conveyed by him/her.

(v) The students shall not write anything on the Question Paper or in other paper or materials during the examination, or pass any kind of paper to any other candidate in the examination-room, or to any person outside the room.

(vi) The students shall not disclose his/her identity to the examiner by writing his/her name or putting any sign / symbol in any part of his answer-script.

(vii) The students shall not use any abusive language or write any objectionable remark or make any appeal to examiner by writing in any part of his answer-script.

(viii) The students shall not detach any page from the answer-script or insert any authorized or unauthorized loose sheet into it. He /she shall also not insert any other answer-script / loose sheet by removing the pins of the origin answer-scripts and re-fixing it.

(ix) The students shall not resort to any disorderly conduct inside the examination-room or misbehave with the invigilator or any other examination official.

VIII. Provision for an Amanuensis (writer):

(i) A candidate may be provided with an Amanuensis (writer) to write down on dictation on his / her behalf on ground of his / her physical disability to write down by himself / herself due to accident or any other reason. The amanuensis may be provided till he / she recovers from the physical disability. The physical disability to write down by himself / herself must be supported by Medical Certificate from a competent Medical Officer.

(ii) The qualifications of the amanuensis so provided must not be equal or higher than that of the candidate. This is also to be supported by Certificate from the Faculty of Study where the Amanuensis is provided.

(iii) Such candidates are to be accommodated in a separate room under the supervision of an invigilator so that the fellow candidates are not disturbed in the process.

C. Credit Point:

It is the product of grade point and number of credits for a course, thus, CP = GP x CR

i. Credit:

A unit by which the course work is measured. It determines the number of hours of instructions required per week. 'Credit' refers to the weightage given to a course, usually in terms of the number of instructional hours per week assigned to it. Credits assigned for a single course always pay attention to how many hours it would take for an average learner to complete a single course successfully.

ii. Grade Point:

Grade Point is a numerical weight allotted to each Grade Letter on a 10-point scale.

iii. Letter Grade:

Letter Grade is an index of the performance of students in a said paper of a particular course. Grades are denoted by letters O, A+, A, B+, B, C, P, F and Abs. Student obtaining Grade F / Grade Abs shall be considered failed/ absent and, will be required to appear in the subsequent ESE. The UGC recommends a 10-point grading system with the following (Table: 1) Letter Grades:

(i) A Letter Grade shall signify the level of qualitative/quantitative academic achievement of a student in a Course, while the Grade Point shall indicate the numerical weight of the Letter Grade on a 10-point scale.

(ii) There shall be 08 (eight) Letter Grades bearing specific Grade Points as listed in Table 1, where the Letter Grades 'O' to 'P' shall indicate successful completion of a course.

(iii) Apart from the 08 (eight) regular Letter Grades listed in Table 1, there shall be 03 (three) additional Letter Grades, which shall be awarded if a Course is withdrawn or spanned over the next Semester or remains incomplete as stated in Table 2.

Table 2: Letter Grades and Grade Points

Letter Grade	Grade Points	Description
O	10	Outstanding
A+	9	Excellent
A	8	Very Good
B+	7	Good
B	6	Above Average
C	5	Average
P	4	Pass
F	0	Fail
Abs	0	Absent
UFM	0	Unfair Means

iv. Grade Point Average:

a. SGPA (Semester Grade Point Average)

The SGPA of a student in a Semester shall be the weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered in that Semester, irrespective of whether he/she could or could not complete the Courses. More specifically, the calculation of SGPA shall take into account the Courses graded with Letter Grades 'O' to 'F' as given in Table 1.

$$\text{SGPA} = \frac{\sum_{i=1}^{n} C_i G_i}{\sum_{i=1}^{n} C_i} \qquad (1.1)$$

The SGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.1) up to two decimal places, where n is the total number of Credit Courses registered by the student in that Semester, Gi is the Grade Point secured in the i^{th} registered Course and Ci is the Credit (weight) of that Course.

b. CGPA (Cumulative Grade Point Average)

(i) The CGPA of a student in a Semester of a Programme shall be the accumulated weighted average of the Grade Points secured by the student in all the Credit Courses (both Core and Elective Courses) he/she registered and successfully completed so far starting from the enrollment in the Programme. In other words,taking into account all the Courses graded with 'O' to 'P' as given in Table 1.1, generally the CGPA of a student shall be calculated starting from the first Semester of his/her enrolled Programme, while the CGPA of a lateral-entry student shall be calculated starting from the Semester of his/her enrollment.

(ii) The CGPA of a student in a Semester shall be calculated on a 10-point scale using Equation (1.2) up to two decimal places, where N is the total number of Credit Courses registered and successfully completed so far by the student, Gi is the Grade Point secured in the ith completed Course and Ci is the Credit (weight)of that Course.

$$\text{CGPA} = \frac{\sum_{i=1}^{N} C_i G_i}{\sum_{i=1}^{N} C_i} \qquad (1.2)$$

(iii) The CGPA shall be convertible into equivalent percentage of marks using Equation Conversion of CGPA to percentage marks: = CGPA*10

D. Post-Examination

i. Transcript or Grade Card or Certificate:

A marking certificate shall be issued to all the registered students after every Semester. The Semester mark sheet will display the course details (code, title, number of credits, grade secured) along with total credit earned in that Semester.

ii. Grievance Readdress Mechanism:

Students with any dissatisfaction or grievance regarding the marks awarded in any of the Papers / Courses may appeal to the Controller of Examinations for remedial action such as Re-evaluation within 10 days of the declaration of result.

(i) A student has options to appeal for re-evaluation of his /her answer script to the Controller of Examination.

(ii) Application for re-evaluation / re-scrutiny of answer scripts shall be made in the definite proforma available with the Examination Office through the head of the respective departments within 10 days of declaration of the results of the respective examinations.

(iii) The Controller of Examination may appoint an examiner for re-evaluation and will consider and recognize the evaluation done by a University appointed examiner.

(iv) There shall be no provision for re-evaluation of the Practical Papers, Project Work, and Dissertation etc. However, the students fail in practical examination or viva voce and wish to appear again may apply to be evaluated can do so with the next schedule.

(v) After screening the application for re-evaluation, the CoE may send the answer scripts of the student to the examiners appointed by the CoE with the approval of Vice Chancellor.

(vi) The marks/grades achieved by the students after the re-evaluation shall be final and binding.

(vii) Fresh Marks – sheets / Grade Card shall be issued only if the candidate secures pass marks / passing grade in the re-evaluated paper.

(viii) Revaluation of answer scripts shall be deemed to be an additional facility provided to the students with a view to improving upon their results at the preceding examination result for any reason whatsoever shall not confer any right upon them for admission to next higher class which matters always be regulated in accordance with the relevant rules or regulations framed by the University.

(ix) If as a result of revaluation of the candidate attracts the provision of condonation of deficiency, the same may be applied to his/her only for fresh attempt.

INSTRUCTION TO TEACHERS AND STUDENTS

(Teaching and Learning Methods)

In all the courses the teacher has to select topics for teacher-method which should not be less than 20 percent. The approach will be direct class room teaching through series of lectures delivering concepts using ITC facilities, white or black board. Notes may also be circulated to the students however; the students are to be involved in preparation of the notes. The teacher will be responsible in selecting the best note for circulation. The teacher- centric methodology has recently fallen out of favour because this strategy for teaching is seen to favour passive students.

1. Student- centric / Constructivist Approach:

The topics of the courses may be selected at the start of the class and assigned one topic to each of the student for studying by themselves, prepare presentations, notes etc., and present at respective class time after consultation and discussion with the course teachers. The teacher facilitates the learning of the students by guiding and providing input and explaining concepts. 60 percent of the course contents may be selected for this purpose. To avoid behaviour problems, teachers must lay a lot of groundwork in student- centric classrooms. Typically, it involves instilling a sense of responsibility in students. In addition, students must learn internal motivation.

a. Project-Based Learning: The teacher may select 5 percent of topics for the purpose and may conduct visit to the laboratory for experiments or field and survey. The selection of the topic may be done considering the available facility for the purpose. However, in the final semester of each of the programme the student has to undergo a project-Based learning at least 4 months duration. This approach will help the student to think critically, evaluate, analyse, make decisions, collaborate, and more.

b. Inquiry-Based Learning: The teacher/ students are supposed to list at least five questions in each contact hour and student solve these question or search for answer which becomes the home work for the students “question-driven” learning approach. The teacher may look for the correctness of the solution or the best possible answer and discuss in the successive class. This will help in the preparation for various competitive examination and develop a habit for search for solutions.

c. Flipped Classroom: About 10 percent of the course content has to be completed by this method. In this approach the students are asked to watchvideo or lecture prepared by the

teacher or any video available (relevant to the course). A set of questions may be given to the students for searching answers by the students. The idea is that students should have more time in-classroom focusing on achieving these higher levels of thinking and learning. The Flipped classroom is also an acronym. The letters FLIP represent the four pillars included in this type of learning: Flexible environment, Learning culture shift, Intentional content, and Professional educator. As you can see, the second pillar refers to a culture shift from the traditional approach where students are more passive to an approach where students are active participants. As a result, this approach is also a student- centric teaching method.

d. Cooperative Learning: The remaining five percent has to be completed by cooperative learning approach. In this approach the students are allotted with problems. During the library hours the student along with the teacher visits library search probable solution for the assigned problem. The same has to be done in group so that the students discuss among themselves for the appropriate answers. Essentially, cooperative learning believes that social interactions can improve learning. In addition, the approach recreates real-world work situations in which collaboration and cooperation are required.

The percentage categorization for the completion of a theory course

Teacher- centric or Direct Classroom Teaching: Delivery by series of lectures	20%
Student- centric Approach, Student present and deliver lectures in presence of teacher and supervised by teacher	60%
Student visit fields or perform experiments or teacher perform demonstration	05%
Flipped Classroom approach	10%
Cooperative learning approach	05%

Inquiry based approach has to be followed in all of the classes

Teacher has to distribute the topics to be considered for teaching by the above-mentioned approaches and prepare lesson plan for execution and maintain a file.

Breakdown of Credits according to NEP2020: Credits (for 2023-24 Syllabus)

Sl. No	Category		Total number of Credits
1	University Core(UC)	Skill Enhancement Course (SEC)	0
		Ability Enhancement Course (AEC)	6
		Field Training	0
		Discipline Specific Elective (DSE)	0
		Value Added Course (VAC)	1
2	University Elective (UE)	Multidisciplinary Course (MDC)	3
		Value Added Course (VAC)	1
3	Program Core(PC)	Discipline Specific Core (DSC)	43
		Field Training	1
		Research /Industry Internship	14
		Summer Internship	0
4	Program Elective (PE)	Discipline Specific Elective (DSE)	6
		Value Added Course (VAC)	0
5	Faculty Core(FC)	Skill Enhancement Course (SEC)	7
		Ability Enhancement Course (AEC)	0
6		Bridge Course	14
Total			**96**

Breakdown by categories of courses

Sl no	Category	Credits	%
1	Science	3	3.125
2	Engineering	92	95.83
3	Commerce and Management	1	1.04
Total			96

PCI, INC, AICTE regulated programs shall have to follow the regulating body

SEMESTER WISE COURSE DISTRIBUTION

	S. N.	Course Code	Course Title	Course Category	Engagement							Maximum Marks for			
					L	T	P	S	R	O	C	IA*	SEE*	PE*	Total
Semester I	1.	24MCAM1101R	Data Visualisation	DSC (Major)	3	0	2	0	0	0	4	40	60	100	200
	2	24MCAM1102R	Web Applications Development	DSC (Minor)	2	0	2	0	0	0	3	40	60	100	200
	3	24MCAM1103R	Programming And Data Structures	DSC (Minor)	2	0	2	0	0	0	3	40	60	100	200
	4	24MCAM1104R	PG Project I	DSC (Minor)	0	0	4	0	0	0	2	0	100	0	100
	5	24MCAM1105R	Mathematics For Computer Applications	MDC	2	0	0	0	0	0	2	40	60	0	100
	6	24UMPD1102R	Effective English	AEC	0	0	4	0	0	0	2	0	0	100	100
	7	24MCAM1106R	Fundamentals of Management	VAC	2	0	0	0	0	0	2	40	60	0	100
	8	24MOSY1101R/ 24MOSY1102R/ 24MOSY1103R	Creative Thinking: Techniques and Tools for Success Imperial College London/ Foundations of Digital Marketing and E-commerce Google/ Marketing Digital Para E-commerce	VAC	1	0	0	0	0	0	1	0	100	0	100
	9	24UBCC1101	Co-Curricular Activities	Co-Curricular	0	0	0	4	0	0	1	0	0	100	100
	Total without Bridge Course				**12**	**0**	**14**	**4**	**0**	**0**	**20**	**200**	**500**	**500**	**1200**
	10	24MCAM1107R	Fundamentals of Graph Theory	Bridge Course	1	0	2	0	0	0	2	40	60	100	200
	11	24MCAM1108R	Algorithms for Problem Solving	Bridge Course	1	0	2	0	0	0	2	40	60	100	200
	Total				**14**	**0**	**18**	**4**	**0**	**0**	**24**	**280**	**620**	**700**	**1600**

***IA: Internal Assessment, SEE: Semester End Examination,**
PE: Practical Examination

	S. N.	Course Code	Course Title	Course Category	Engagement							Maximum Marks for			
					L	T	P	S	R	O	C	IA*	SEE*	PE*	Total
Semester II	1.	24MCAM1201R	Cloud Fundamentals	DSC (Major)	3	0	2	0	0	0	4	40	60	100	200
	2	24MCAM1202R	Advanced Database Management Systems	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	3	24MCAM1203R	Operating Systems Concepts and Design	DSC (Minor)	2	0	2	0	0	0	3	40	60	100	200
	4	24MCAM1204R	PG Project II	DSC (Minor)	0	0	0	4	6	0	2	0	100	0	100
	5	24UMPD1202R	Communication Mastery	AEC	0	0	4	0	0	0	2	0	0	100	100
	6	24MCAM1205R	Teaching Assistantship	SEC	0	0	4	0	0	0	2	0	0	100	100
	7	24MCAM1206R	Logical Reasoning and Quantitative Aptitude	SEC	2	0	0	0	0	0	2	40	60	0	100
	8	24MOSY1201R/ 24MOSY1202R/ 24MOSY1203R	1. Leading Diverse Teams & Organizations 2. Generative AI with Large Language Models 3. Google - Digital Transformation Using AI/ML with Google Cloud Specialization	VAC	1	0	0	0	0	0	1	0	0	100	100
	9	24UCDT1201R	Ideation and Design Thinking		1	0	0	0	0	0	1	0	0	100	100
	10	24UBEC1201R	Extra-Curricular Activities	Extra-Curricular	0	0	0	4	0	0	1	0	0	100	100
	11	24MCAM1208R	Field Visit	Field Training	0	0	0	0	0	8	1	0	0	100	100
	Total without Bridge Course				**11**	**0**	**14**	**8**	**6**	**8**	**22**	**160**	**340**	**900**	**1400**
	12	24MCAM1209R	Fundamentals of Discrete Mathematics	Bridge Course	2	0	0	0	0	0	2	40	60	0	100
	13	24MCAM1210R	Mathematics for Basic Programming	Bridge Course	2	0	0	0	0	0	2	40	60	0	100
	Total				**15**	**0**	**14**	**8**	**6**	**8**	**26**	**240**	**460**	**900**	**1600**

***IA: Internal Assessment, SEE: Semester End Examination,**
PE: Practical Examination

	S. N.	Course Code	Course Title	Course Category	Engagement							Maximum Marks for			
					L	T	P	S	R	O	C	IA*	SEE*	PE*	Total
Semester III	1.	23MCAO2101R	Predictive Analysis	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	2	23MCAO2102R	Computer Network and Information Security	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	3	23MCAO2103R	Usability Design of Software Application	DSC (Major)	2	0	0	0	0	0	2	40	60	0	100
	4	23MCAO2104R	Object Oriented Paradigm Using Java	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	5	23MCAO2105R/ 23MCAO2109R	PE I - Fundamentals of AI/ML / Business Intelligence	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	6	24MOSY2101R	PDP -Corporate Proficiency	VAC	0	0	2	0	0	0	1	0	0	100	100
	7	23MCAO2106R	Summer Internship	Summer Internship	0	0	0	16	0	0	4	0	0	100	100
	8	23MCAO2107R	PG Research Project I	Research / project	0	0	4	8	24	0	8	0	0	100	100
	9	23MCAO2108R	Field Visit	Field Training	0	0	0	0	0	8	1	0	0	100	100
	Total				**10**	**0**	**14**	**24**	**24**	**8**	**28**	**200**	**300**	**800**	**1300**

***IA: Internal Assessment, SEE: Semester End Examination, PE: Practical Examination**

	S. N.	Course Code	Course Title	Course Category	Engagement							Maximum Marks for			
					L	T	P	S	R	O	C	IA*	SEE*	PE*	Total
Semester IV	1.	24MCAM2201R	Artificial Intelligence and Deep Learning	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	2	24MCAM2202R	PE II – Data Engineering / Pattern Recognition	DSC (Major)	2	0	2	0	0	0	3	40	60	100	200
	3	24MCAM2203R	PG Research Project II	Research / project	0	0	4	8	56	0	12	0	0	100	100
	Total				**4**	**0**	**8**	**8**	**56**	**0**	**18**	**80**	**120**	**300**	**500**

***IA: Internal Assessment, SEE: Semester End Examination,**
PE: Practical Examination

SEMESTER – I									
Course Title	**DATA VISUALIZATION**								
Course code	**24MCAM1101R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 45T+30P	**3**	**0**	**2**	**0**	**0**	**0**	**4**
Pre-requisite	**BasicProgramming Knowledge**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives	1. Gain insights into the significance of data visualization in IT applications and its role in decision-making. 2. Master R programming, Python, and their visualization libraries (e.g., ggplot2, Matplotlib) while exploring interactive tools like Tableau or Cognos. 3. Learn to analyse and visualize data effectively to address and solve practical IT challenges.								
CO1	Understand the importance of data visualization in IT applications and its role in decisionmaking processes.								
CO2	Demonstrate proficiency in using R programming and Python for data analysis andvisualization tasks.								
CO3	Utilize data visualization libraries like ggplot2 (R) and Matplotlib (Python) to createcustomized visualizations.								
CO4	Design clear and persuasive visualizations using principles of effective data presentationand storytelling.								
CO5	Utilize interactive data visualization tools like Cognos to create dynamic visualizations andapply data analysis techniques and visualization tools to solve complex IT problems.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to DataVisualization and RProgramming:**Understanding the importance of data visualization in ITapplications; Overview of R programmingfor data analysis andvisualization; Introduction to datavisualization libraries: ggplot2(R) and Matplotlib (Python); Basic plotting techniques anddata representation in R andPython; Hands-on exercises withsimple plots and datavisualization examples.	**9**	Explain the significance ofdata visualization in ITapplications and utilize Rprogramming along withggplot2 and Matplotliblibraries for data analysisand visualization.	1,2
II	**Advanced Data Visualization Techniques:** Advanced Features of ggplot2for Customized Plots in R; Customizing Plot Aestheticsand Themes in ggplot2; Introduction to Interactive DataVisualization Tools: Cognos; Creating InteractiveVisualizations using Cognos; Hands-on Exercises: AdvancedPlotting Techniques andInteractive Visualizations.	**9**	Apply advanced features ofggplot2 for creatingcustomized plots, utilizeCognos for interactive datavisualizations, anddemonstrate proficiencythrough hands-on exercisesin advanced plottingtechniques and interactivevisualizations.	2,3
III	**Maps, Geospatial Data Visualization, Export Features:**Introduction to Folium; Maps with Markers, Choropleth Maps; Export Feature – Data Visualization. Generating a PNG picture; Generating PDF documents	**9**	Create geospatial visualizations using Folium, export data visualizations in various formats (PNG, PDF), plot multiple graphs with subfigures, and implement	1,2

	Multiple graphs plotting and export; Inserting subfigure Hypothesis and Gradient Descent; Understanding Hypothesis; Implementation of hypothesis in Python; Gradient Descent Implementation		hypothesis testing and gradient descent algorithms in Python	
IV	**Visualization by using Seaborn Library:** Relational plot: Dist Plot, Line Plot, Lmplot ; Categorical plot: Stripplot, Swarmplot,Barplot, Countplot, Boxplot, Violinplot, Stripplot Distribution plot: Joinplot, Distpot, Pairplot, Rugplot; Regression plot: Simple Linear plot with additional parameters (hue and markers), Setting size and color of the plot, Displaying multiple plots, Size and aspect ratio of plots; Matrix plot: Heatmaps, Cluster Maps.Style and Color: Set the background to be white, Set the background to be ticks, Set the background to be darkgrid, Set the background to be whitegrid.	**9**	Utilize the Seaborn library to create a variety of plots, customize the style and color of visualizations, and adjust plot settings such as size, aspect ratio, and axes for effective data representation.	1,2
V	**Application of Data Visualization in Engineering:**Case studies demonstrating data visualization; Visualizing datasets & Creating dashboards for analysis; Real-world applications and projects.	**9**	Apply data visualization techniques to analyze engineering datasets, create dashboards, and interpret insights from real-world case studies and projects.	1,2

TEXT BOOKS:

T1:Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell. 4th edition. New York: Garland Science; 2002.

REFERENCE BOOKS:

R1: Cooper GM. The Cell: A Molecular Approach. 2nd edition. Sunderland (MA): Sinauer Associates; 2000.

R2: Ambrose and Dorothy. Cell Biology. 2nd Edition. MEasty, ELBS Publications; 1970.

R3: Sharp, Lester W. Fundamentals of Cytology. 1st edition. Mc Graw Hill Company; 1943.

OTHER LEARNING RESOURCES:

https://www.ncbi.nlm.nih.gov/books/NBK9839/?term=cell%20Biolpgy

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Able to explain the fundamentals and advances of cytology including structure and functions of cell and cell organelles.	**1,3 & 4**
2	Able to explain the cell cycle and cell division.	**1,2**
3	Learn and develop skills for operating microscope, preparing slides by various staining techniques	**7.9,10**
4	Apply knowledge of cellular processes to explain how cells operate and interact within living organisms.	**5,7**
5	Demonstrate a comprehensive understanding of cell structure and function.	**5,8**

SEMESTER – I									
Course Title	**WEB APPLICATIONS DEVELOPMENT**								
Course code	**24MCAM1102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**BasicProgramming Knowledge**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives	1. To understand the Fundamental principle of Web design, hosting and maintenance. 2. To understand tools required for Web design 3. To develop robust server-side applications using Node.js, other back-end frameworks.								
CO1	Understand the fundamentals of web application development, including client-side and server-side technologies.								
CO2	Create web pages using HTML, Cascading Style Sheets, JavaScript and XML								
CO3	Develop dynamic web applications using server-side programming languages and frameworks such as PHP, Python								
CO4	Implement data storage and retrieval in web applications using databases like MySQL								
CO5	Understand basics of Internet								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Basics of Internet:** Client/Server Computing: What is C/S Computing, Middleware, Fat client VS Fat Servers, N-tiered Software Architecture. WWW, Browser, Server, Webpage, Web development, HTML, HTTP, IP address, DNS & ports; secure HTTP, Secure Socket layer, WWW Proxies	**8**	Describe, illustrate, and explain Internet, Internet history and browsers, Servers.	1, 2
II	**Searching and web casting Technique:** Popular web servers, basic feature; bookmarks, cookies, progress indicators, customization of browsers, browsing tricks, search engines, architecture of search engines, search tools, web crawlerSed : Scripts, Operation, Addresses, commands, Applications, grep and sed.	**8**	Demonstrate the use of bookmarks, cookies, and progress indicators in web browsers. Integrate web technologies and search techniques into real-world applications.	1, 2
III	**Web page design:** Designing web pages with HTML- use of tags, hyperlinks, URLs, tables, text formatting, graphics & multimedia, imagemap, frames and forms in web pages. Use of Cascading Style Sheet in web pages. Introduction to DHTML.	**10**	Design and develop interactive, well-structured, and visually appealing web pages using HTML, CSS, and DHTML.	2, 3
IV	**Creating interactive and dynamic web pages with JavaScript:** JavaScript overview; constants, variables, operators, expressions & statements; user-defined & built-in functions; client-side form validation; using properties	**10**	Describe key concepts of JavaScript, including constants, variables, operators, expressions, statements, and XML basics such as user-defined tags, XML DTDs, and XSL AND Implement	2, 3, 4

	and methods of built-in objects. **Extensible Markup Language (XML):** Introduction- using user-defined tags in web pages; displaying XML contents; XML DTDs; use of XSL		JavaScript Design and develop dynamic and interactive web pages that leverage JavaScript for interactivity and XML for structured data	
V	**Web Server:** Web services and web server functionality; web server composition; registration; conceptual architecture of some typical web servers. Server-side scripting: overview of server-side scripts like CGI, ASP, and JSP. Server-side scripting using PHP; Web database connectivity- introduction to ODBC; PHP with database connectivity.	**8**	Explain the concepts and purposes of server-side scripting languages, develop and deploy web applications that incorporate server-side scripting, demonstrating a comprehensive understanding of web server functionality and web services..	3, 4, 5
Practical				
Practical 1	Write a HTML program for the demonstration of Unordered Lists.	2	Learn to createUnordered Lists	2, 3
Practical 2	Write a HTML program for the demonstration of ordered Lists	2	Learn to create Ordered Lists	2, 3
Practical 3	Write a HTML program for the demonstration of Definition Lists	2	Learn to create Definition Lists	2
Practical 4	Write a HTML program for the demonstration of Nested Lists	2	Learn to create Nested Lists Lists	2, 3
Practical 5	Write a HTML program for demonstrating Hyperlinks. Navigation from one page to another and Navigation within the page.	2	Understand hyperlink and implement hyperlink navigation from one page to another and within the page	2, 3
Practical 6	Write a HTML program for time-table using tables.	2	Understand and implement queue using linked list	2, 3
Practical 7	Write a HTML program to develop a static Home Page using frames.	2	Understand and apply polynomial addition and subtraction	4
Practical 8	Write a HTML program to develop a static Registration Form	2	Understand and convert infix to postfix using C program.	3
Practical 9	Write a HTML program to develop a static Login Page.	2	Evaluate expression using C program.	3
Practical 10	Write a HTML program to develop a static Web Page for Catalog.	2	Understand AVL tree and implement using array.	3
Practical 11	Write a HTML program to develop a static Web Page for Shopping Cart.	2	Develop heap using priority queue.	3
Practical 12	Write HTML for demonstration of cascading Embedded stylesheets.	2	Understand graph and its representation.	3
Practical 13	Write HTML for demonstration of cascading External stylesheets	2	Understand the concepts of traversal breadth first traversal.	3
Practical 14	Write HTML for demonstration of cascading Inline styles.	2	Understand and apply the concepts of traversal depth first traversal.	3
Practical 15	Write a JavaScript program to validate USER LOGIN page.	2	Develop a user login page	3

Practical 16	Write a program for implementing XML documents for CUSTOMER DETAILS.	2	Develop skills in creating well-formed XML documents.	3
Practical 17	Write an internal Document Type Definition to validate XML for CUSTOMER DETAILS	2	Learn how to create an internal DTD to define the structure and constraints of an XML document.	3
Practical 18	Write an external Document Type Definition to validate XML for CUSTOMER DETAILS	2	Develop skills in managing and maintaining DTD files for multiple XML documents..	3
Practical 19	Write a simple servlet that displays a message	2	Develop skills in handling HTTP requests and generating HTTP responses.	3
Practical 20	Write a servlet that reads parameters from employee login page	2	Understand how to read and process parameters from HTTP requests in a servlet.	3

TEXT BOOKS:

T1: C. Xavier, "Web Technology & Design", New Age Publication, 2003
T2: Austin and Pawlan, "Advanced Programming for JAVA2 Platform", Pearson, 2000

REFERENCE BOOKS:

R1: Oliver, Dick; SAMS Teach Yourself Html 4 in 24 Hours; Techmedia.
R2: Ashbacher, Charles; SAMS Teach Yourself XML in 24 Hours; Techmedia.

OTHER LEARNING RESOURCES:

https://onlinecourses.swayam2.ac.in/nou24_cs09/preview

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamentals of web application development, including client-side and server-side technologies.	**1, 5**
2	Create web pages using HTML, Cascading Style Sheets, JavaScript and XML	**3, 5**
3	Develop dynamic web applications using server-side programming languages and frameworks such as PHP, Python	**1, 3**
4	Implement data storage and retrieval in web applications using databases like MySQL	**1, 4**
5	Understanding the basics of the Internet.	**1, 8**

SEMESTER – I									
Course Title	**PROGRAMMING AND DATA STRUCTURES**								
Course code	**24MCAM1103R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives	1. To understand the basic programming in C 2. To acquire the basic concepts of Data Structures and their applications. 3. To acquire knowledge on Searching and Sorting and their applications.								
CO1	Understand the fundamental concepts and principles of data structures								
CO2	Use and implement appropriate data structure for the required problems using a programming language such as C.								
CO3	Implement various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.								
CO4	Understand various searching & sorting techniques.								
CO5	Apply algorithms for various sorting techniques and compare their performance in terms of Space and Time complexity.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to Data Structures:** Algorithms and Flowcharts, Basics Analysis on Algorithm, Complexity of Algorithm, Introduction and Definition of Data Structure, Classification of Data, Arrays, Various types of Data Structure, Static and Dynamic Memory Allocation, Function, Recursion.	**4**	Interpret and create flowcharts to represent algorithms systematically, understand the concepts of algorithmic complexity and its significance in evaluating the efficiency of algorithms.	2, 3
II	**Arrays, Pointers and Strings:** Introduction to Arrays, Definition, One Dimensional Array and Multidimensional Arrays, Pointer, Pointer to Structure, various Programs for Array and Pointer. Strings. Introduction to Strings, Definition, Library Functions of Strings.	**6**	Understand the concept of arrays as a sequential collection of elements of the same data type. Understand pointers as variables that store memory addresses. Define strings as arrays of characters	2
III	**Stacks and Queue:** Introduction to Stack, Definition, Stack Implementation, Operations of Stack, Applications of Stack and Multiple Stacks. Implementation of Multiple Stack Queues, Introduction to Queue, Definition, Queue Implementation, Operations of Queue, Circular Queue, De-queue and Priority Queue.	**6**	Understand the concept of a stack as a Last In, First Out (LIFO) data structure. Identify scenarios where stacks are applicable, Understand the concept of a queue as a First In, First Out (FIFO) data structure.	2
IV	**Linked Lists and Trees:** Introduction, Representation and Operations of Linked Lists, Singly Linked List, Doubly Linked List, Circular Linked List, And Circular Doubly Linked List **Trees:** Introduction to Tree, Tree Terminology Binary Tree, Binary Search Tree, Strictly	**8**	Understand the concept of a linked list as a linear data structure, understand tree terminologies such as parent, child, sibling, leaf, and height.	2, 3

	Binary Tree, Complete Binary Tree, Tree Traversal, Threaded Binary Tree, AVL Tree B Tree, B+ Tree.			
V	**Graphs, Searching, Sorting and Hashing Graphs:** Introduction, Representation to Graphs, Graph Traversals Shortest Path Algorithms. **Searching and Sorting:** Searching, Types of Searching, Sorting, Types of sorting like quick sort, bubble sort, merge sort, selection sort **Hashing:** Hash Function, Types of Hash Functions, Collision, Collision Resolution Technique (CRT), Perfect Hashing	**6**	Explain the operational principles and theoretical foundations of finite state machines, evaluate the efficiency and effectiveness of using different types of mathematical machines for specific computational problems	3, 4, 5
Practical				
Practical 1	Write a C program to calculate the sum of first n natural numbers using for loop.	**2**	Understand the basic syntax, data types, variables, operators, and control structures in C.	2, 3
Practical 2	Write a C program to find the factorial of a number.	**2**	Analyse and Develop skills in designing and writing structured and modular programs using procedural programming concepts.	3, 4
Practical 3	Write a C program to print Fibonacci series using for loop.	**2**	Develop skills in designing and writing structured and modular programs using procedural programming concepts.	4
Practical 4	Write a C Program to check whether a number is a prime number or not.	**2**	Develop the ability to implement an algorithm to determine whether a number is prime using efficient techniques like trial division.	4
Practical 5	Write a C program to print Prime Numbers From 1 to N.	**2**	Gain a clear understanding of prime numbers and their characteristics	2
Practical 6	Write a C program to insert elements in an array and display the same.	**2**	Understand the basics of arrays in C programming, including declaration and initialization.	2, 3
Practical 7	Write a C program to search an element in array using C (linear search)	**2**	Understand and implement the linear search algorithm for finding an element in an array.	2, 4
Practical 8	Write a C program to implement stack operations (Push, Pop and Display).	**2**	Understand the stack data structure and its operations (Push, Pop).	2

Practical 9	Write a C Program to Implement a Queue using an Array(insert, delete and display).	**2**	Understand the queue data structure and its operations (Insert, Delete). Implement a queue using an array and manage elements using queue operations.	2, 3
Practical 10	Write a C program to sort N numbers in ascending order using Bubble sort and print both the given and the sorted array	**2**	Understand the bubble sort algorithm and its basic principles. Learn to manipulate arrays for sorting purposes.	2, 3
Practical 11	Write a C Program to sort an array using Insertion Sort.	**2**	Understand the insertion sort algorithm and its key features. Implement insertion sort to rearrange elements in an array.	2, 3, 4
Practical 12	Write a C Program to sort an array using Selection Sort.	**2**	Understand the selection sort algorithm and its characteristics. Apply selection sort to sort arrays of varying sizes efficiently.	2, 3, 4
Practical 13	Write a C program to perform Link List operations (Insert in the beginning, last, specified position, Delete from Beginning, last, specified location, Show)	**2**	Implement linked list operations (Insert at beginning, end, specified position; Delete from beginning, end, specified position).	4
Practical 14	Write a C program to convert Infix to post-fix	**2**	Understand the process of converting infix expressions to postfix (Reverse Polish Notation),	2, 3
Practical 15	Write a C program to convert Post-fix to infix	**2**	Implement a stack to assist in the conversion of postfix expressions to infix	3,4, 5

TEXT BOOKS:

T1: Fundamentals of Data structures in C, 2nd Edition,E. Horowitz, S. Sahni and Susan,Anderson-Freed, University Press.

REFERENCE BOOKS:

R1: Data structures A Programming Approach with C, D. S. Kushwaha and A. K. Misra, PHI.
R2: Data structures and Program Design in C,R. Kruse, C. L. Tondo and B. LeungPearson.

ONLINE LEARNING RESOURCES:

https://www.programiz.com/dsa

https://onlinecourses.swayam2.ac.in/cec24_cs17/preview

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the fundamental concepts and principles of data structures	**1, 5**
2	Use and implement appropriate data structure for the required problems using a programming language such as C.	**1, 3**
3	Implement various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.	**1, 3**
4	Understand various searching & sorting techniques.	**2, 5**
5	Apply algorithms for various sorting techniques and compare their performance in terms of Space and Time complexity.	**1, 4**

SEMESTER – I									
Course Title	**PG PROJECT I**								
Course code	**24MCAM1104R**	**Total credits: 2** **Total hours: 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives	1. To develop the practical skill 2. To enhance Critical Thinking and Problem-Solving Abilities 3. To Improve Communication and Collaboration Skills								
CO1	Demonstrate a sound technical knowledge of the selected project topic.								
CO2	Illustrate problem identification, formulation, and solution.								
CO3	Plan solutions to complex problems utilizing a systems approach.								
CO4	Design the prototype of the solution and analyse the prototype using testing methods.								
CO5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Project Introduction and Planning:** Overview of project goals and objectives. Importance of project planning and management. Project Proposal Development: Identifying a project topic or problem Conducting preliminary research Writing a project proposal Setting timelines and milestones Resource allocation (materials, tools, etc.) Risk assessment and management strategies	**6**	Describe, illustrate, and explain problem	2, 3
II	**Literature Review and Research Methodology:** Identifying and reviewing relevant literature Summarizing and synthesizing existing research Highlighting gaps and formulating research questions Choosing appropriate research methods (qualitative, quantitative, or mixed methods) Designing research instruments (surveys, interviews, experiments, etc.) Ethical considerations in research	**6**	Describe, illustrate and Literature Review	2, 3, 4
III	**Data Collection and Analysis:** Gathering primary data (surveys, experiments, interviews, observations) Ensuring data accuracy and reliability, Storing and organizing data Using statistical tools and software for data analysis, Interpreting data and drawing	**6**	Describe, illustrate and explain data collection and analysis	2, 3, 4

	conclusions, Visualizing data (charts, graphs, tables)			
IV	**Project Implementation and Monitoring:** Project Execution, Implementing the project plan Monitoring progress and making adjustments as needed Documentation and Record Keeping: Keeping detailed records of project activities and outcomes Troubleshooting and problem-solving during implementation, Interim Reporting: Preparing and presenting progress reports Receiving and incorporating feedback	**6**	Describe, illustrate and explain project implementation and execution	6
V	**Project Presentation and Evaluation Final Reporting:** Writing the final project report (structure, content, formatting) Ensuring clarity, coherence, and completeness Project Presentation: Preparing visual aids (slides, posters, etc.) Practicing presentation skills (public speaking, answering questions) Evaluation and Reflection: Assessing project outcomes against objectives Reflecting on the project process and personal learning Identifying areas for future improvement and potential follow-up projects	**6**	Describe, illustrate and explain testing and documentation	2,3, 5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate a sound technical knowledge of selected project topic.	**1, 2, 4**
2	Illustrate problem identification, formulation and solution.	**1, 2**
3	Plan solutions to complex problems utilizing a systems approach.	**1, 2, 3, 5, 7**
4	Design the prototype of the solution and analyse the prototype using testing methods.	**1, 2, 3, 5, 6, 7, 8**
5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.	**1, 2, 3, 6, 8**

SEMESTER – I									
Course Title	**MATHEMATICS FOR COMPUTER APPLICATIONS**								
Course code	**24MCAM1105R**	**Total credits: 2** **Total hours: 30T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives	1. Use discrete mathematics to model and solve problems in computer science 2. Understand and apply basic algebraic concepts and techniques 3. Use mathematical knowledge to enhance their understanding of other computer science topics								
CO1	Understand and use the principles of mathematical reasoning to solve real-world problems.								
CO2	Interpret the results of mathematical computations and make contributions to their field.								
CO3	Use appropriate mathematics skills and techniques in the analysis of data								
CO4	Demonstrate proficiency in using matrix techniques to develop solutions								
CO5	Apply mathematical machines to solve practical problems in areas such as automata theory, formal languages, and computational complexity.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Set Relations and Functions:** Definition of sets and subsets, Intersection, Union and complements, DeMorgan's law, Cardinality Relations, Equivalence Relations etc, Mappings One-one, Onto etc, Counting principles, pigeon hole principle, Lattice, Boolean Algebra, Grammar and Language	**6**	Explain the properties of relations, examine and distinguish between different types of relations, mappings, and algebraic structures	1, 2
II	**Algebraic Structures:** Group, Semigroup, Monoid, Ring, Field, Vector Space	**6**	Explain the axioms and properties, use the principles and operations of groups, semigroups, monoids, rings, fields, and vector spaces to solve relevant mathematical problems.	2, 3
III	**Logic**: Logic operators like AND OR etc, Truth Tabkes, Theory of inference and deduction, Mathematical Induction, Predicate Calculus	**6**	Explain the construction and interpretation of truth tables, Examine logical statements and arguments, identifying valid and invalid inferences, and analyzing the structure of logical proofs.	2, 3
IV	**Linear Equation and Matrices**: Row / column operations, Gaussian Eliminations, Decomposition, Inverse System of linear equations, Linear dependence, Eigen values, Eigen vectors	**6**	Explain the principles and procedures, analyze the structure and properties of matrices, interpret the significance of eigenvalues and eigenvectors in practical applications.	2, 3, 4
V	**Mathematical Machines**: Finite state machine, Push down automata and Turing	**6**	Explain the operational principles and theoretical foundations of	3, 4, 5

	machine (Introduction and preliminary concepts only)		finite state machines, evaluate the efficiency and effectiveness of using different types of mathematical machines for specific computational problems	

TEXT BOOKS:

T1: Elements of Discrete Mathematics – A Computer Oriented Approach, Liu C.L., 4th Ed.

REFERENCE BOOKS:

R1: Discrete Mathematical Structures with Applications to Computer Science, Tremblay, Manohar, 1st. Ed.

OTHER LEARNING RESOURCES:

https://onlinecourses.nptel.ac.in/noc24_ma68/preview

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and use the principles of mathematical reasoning to solve real-world problems.	**1, 2**
2	Interpret the results of mathematical computations and make contributions to their field.	**2, 4**
3	Use appropriate mathematics skills and techniques in the analysis of data	**1, 4**
4	Demonstrate proficiency in using matrix techniques to develop solutions	**1, 3**
5	Apply mathematical machines to solve practical problems in areas such as automata theory, formal languages, and computational complexity.	**1, 3**

SEMESTER – I									
Course Title	**PDP I - EFFECTIVEENGLISH**								
Course code	**24UMPD1102R**	**Total credits: 4** **Total hours: 30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To introduce the types of sentences and their significance. 2. To strengthen the students' vocabulary to enhance their speaking and writing skills. 3. To familiarize the students with the importance of dress codes in various organizations. 4. To introduce the 3 P's (Planning, prioritizing & performing) of Time Management. 5. To give insight into English pronunciation and into central concepts in phonetics.								
CO1	Enable students to analyze and identify the different types of sentences.								
CO2	Integrate the skills of reading and speaking in professional communication.								
CO3	Learn Dress code Etiquettes that will boost their confidence and morals.								
CO4	Learn about the effective and efficient utilization of time.								
CO5	Develop learners' pronunciation with the knowledge of Phonetics.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Grammar:** Interchange of Interrogative and Assertive Sentences, Exclamatory and Assertive Sentences Types of Tenses,Common Errors, Synonyms, Antonyms, Homonyms	**12**	Understand the rules for converting interrogative sentences into assertive sentences and vice versa. Develop the ability to express the same idea in different sentence forms, enhancing language flexibility and communication skills.	2, 3, 4
II	**Reading Skills:** Techniques of Effective Reading, Gathering ideas and information from a text The SQ3R Technique Interpret the text	**12**	Develop and apply various reading strategies such as skimming, scanning, and in-depth reading for different purposes.	3
III	**Listening Skills:** What is listening? The Process of Listening, Factors that adversely affect Listening, Difference between Listening and Hearing, Purpose and Importance of Effective Listening, How to Improve Listening Process,	**12**	Learn the stages of the listening process: receiving, understanding, evaluating, remembering, and responding. Develop awareness of each stage to improve overall listening effectiveness. Acquire and practice techniques to improve the listening process, such as maintaining eye contact, providing feedback, and avoiding interruptions.	2, 3, 4
IV	**Conflict Management** Definition, Type of Conflict Management,	**12**	Define conflict management and understand its significance in	3, 4

	Effects of Conflict Management, Methods to deal with Conflicts (Negative)		personal, professional, and social contexts. Develop techniques to address and manage negative conflicts effectively, such as active listening, empathy, and open communication.	
V	**Time-Management Skills** Introduction To Time Management, Purpose And Importance of Time Management, Basic Tips to Maintain Time. Activity: Problem solving activity: A situation will be given to the students and they will have to tell us how to handle the situation or solve the problem.	**12**	Recognize the role of effective time management in achieving goals and maintaining work-life balance. Understand the purpose of time management, including improving efficiency, productivity, and reducing stress. Develop skills to track and analyze how time is spent on various activities.	3, 4, 5

TEXT BOOKS:

T1: Wren,P.C and Martin,H. 1995. High School English Grammar and Composition, S Chand Publishing.
T2: English Grammar in Use, Raymond Murphy 4th edition,CUP.
T3: Barrett, Grant. 2016. Perfect English Grammar: The Indispensible Guide to Excellent Writing and Speaking, Zephyros Press.

REFERENCE BOOKS:

R1: English Vocabulary in Use (Advanced), Michael McCarthy and Felicity, CUP.
R2: Effective Communication and Soft Skills, Nitin Bhatnagar, Pearsons.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enable students to analyze and identify the different types of sentences.	**2, 9**
2	Integrate the skills of reading and speaking in professional communication.	**6, 9**
3	Learn Dress code Etiquettes that will boost their confidence and morals.	**7, 8**
4	Learn about the effective and efficient utilization of time.	**11, 12**
5	Develop learners' pronunciation with the knowledge of Phonetics.	**6, 9**

SEMESTER – I									
Course Title	**FUNDAMENTALS OF MANAGEMENT**								
Course code	**24MCAM1106R**	**Total credits: 4** **Total hours: 30T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To familiarize the students with the concept of management and its evolution. 2. To provide the knowledge and applicability of the various functions of management. 3. To gain knowledge on the contemporary issues and trends in managing an organization. 4. To apply managerial knowledge to the real-world management challenges through case studies. 5. To equip the students with the applicability of management practices in their day to day lives.								
CO1	Describe the primary aspects of management and the theories related to the evolution of management.								
CO2	Understand the fundamentals of Planning function in management and its applicability in managerial decisions.								
CO3	Illustrate the different approaches in organizing and managing human resource in an organization.								
CO4	Exhibit the aspects of directing and controlling function of management and their applicability.								
CO5	Identify the emerging trends and the key issues in managing organizations in the era of globalization.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Development of Management Thought**: Basics of Management -Meaning and Definition, Need, Scope, Characteristics and levels of Management. Functions of Management. P-O-L-C Framework of Management. Roles and Qualities of a Manager. Management Vs Administration. Classical and Contemporary theories of Management - Robert Owen, F.W.Taylor, Henry Fayol, Hawthorne Studies by Elton Mayo, Peter F. Drucker, C.K. Prahlad and Peter Senge	**6**	Understand the basic terms and need of management. The evolution of management thoughts by different theorists both classical and contemporary.	1,2
II	**Fundamentals of Planning:** Planning- Meaning, importance, limitations and steps in planning. Planning and Forecasting. Types of Plans. Concept of MBO, MBE and Planning Premises. Strategic Planning and Analysis of Business Environment- SWOT Analysis and BCG Matrix. Concept and types of decision making.	**4**	Identify the nature, need and process of planning in effective management. Analyse the different tools for analysis of business environment.	1,2, 4

III	**Organizing and Staffing Function:** Organizing – Meaning and importance. Concept of Organogram and Span of Management. Formal and Informal Organization. Authority Vs Responsibility. Concept of Departmentation. Difference between delegation, centralization and decentralization Staffing- Meaning, nature, importance and process. Job analysis and job description. Personnel Management and HRM. HRM- Meaning and objectives, Recruitment and Selection, Training and Development, Compensation and Performance Appraisal.	**8**	Understand the key elements of organizing function. Evaluate the different approaches of staffing and its applicability.	1,2, 5
IV	**Directing and Controlling Function:** Directing - Meaning of Directing and supervising. Motivation – Meaning, importance, types and techniques. Theories – Maslow's Need Hierarchy Theory, Mc Gregor's Theory X and Y, Herzberg's -Two Factor Theory. Leadership – Meaning, importance and qualities. Leadership styles. Communication – Meaning, importance, channels and types. Control – Meaning, importance and techniques and effective Control system.	**9**	Apply the different concepts and theories of motivation and leadership in managing a firm. Assess the role of effective communication skills.	2,3, 4
V	**Contemporary Concepts and Issues in Management**: Concept of Knowledge Management, Corporate Social Responsibility (CSR), Corporate Governance, Business Ethics, ESG Framework and Sustainable Development Goals (SDGs). Emerging Trends in management.Challenges faced in managing Organizations in 21st Century.	**3**	Identifying the emerging trends and management practices. Assess the challenges encountered by managers in global context.	1,2, 4

Text Books:

T1: Principles of Management, Neeru Vasishth and Vibhuti Vasishth, Taxmann's,2022, New Delhi.
T2: Principles & Practice of Management, L.M. Prasad, Sultan Chand & Sons,2019, New Delhi.
T3: Principles of Management, P.C. Tripathi & P.N. Reddy, Tata McGraw, 2017, New Delhi.

Reference Books:

R1: Essentials of Management, B.P.Singh & A.K.Singh ,Excel Books, New Delhi.
R2: Management Concepts and Practice, T N Chhabra, Dhanpat Rai & Co. (Pvt.Ltd.),2018, New Delhi.

OTHER LEARNING RESOURCES:

- https://youtube.com/watch?v=G4vr5jz3Z9w
- https://www.youtube.com/watch?v=q6LMjurECZM
- https://archive.nptel.ac.in/courses/110/107/110107150/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Describe the primary aspects of management and the theories related to the evolution of management.	**1**
2	Understand the fundamentals of Planning function in management and its applicability in managerial decisions.	**3**
3	Illustrate the different approaches in organizing and managing human resource in an organization.	**6**
4	Exhibit the aspects of directing and controlling function of management and their applicability.	**4**
5	Identify the emerging trends and the key issues in managing organizations in the era of globalization.	**8, 10**

SEMESTER – I									
Course Title	**MOOCS I - FOUNDATIONS OF DIGITAL MARKETING AND E-COMMERCE**								
Course code	**24MOSY1102R**	**Total credits: 4** **Total hours: 15T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. Develop a foundational understanding of digital marketing and e-commerce to prepare learners for entry-level roles in these fields. 2. Equip learners with practical skills and knowledge to create effective digital marketing strategies and manage e-commerce platforms. 3. Provide hands-on experience with industry-standard tools like Google Ads, Shopify, and MailChimp, enabling learners to apply what they learn in real-world scenarios.								
CO1	Understand the key concepts of digital marketing and e-commerce and how they create value for businesses.								
CO2	Create and execute digital marketing strategies that effectively attract and engage customers.								
CO3	Analyze customer data and marketing performance to make informed decisions that improve outcomes.								
CO4	Build and manage e-commerce stores that enhance the customer experience and drive sales.								
CO5	Utilize digital tools and platforms to implement and optimize marketing campaigns across various channels.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to Digital Marketing and E-commerce:**Overview of Digital Marketing and E-commerce Evolution of Digital Marketing in the Modern Business World Key Concepts and Terminology in Digital Marketing and E-commerce Understanding the Difference Between Digital Marketing and Traditional Marketing Introduction to Online Consumer Behavior	**3**	**Understand the fundamental concepts** of digital marketing and e-commerce, including key terminology and the distinction between digital and traditional marketing methods.	1
II	**The Customer Journey and Marketing Funnel:**Defining the Customer Journey: Stages and Touchpoints The Marketing Funnel: Awareness, Consideration, Conversion, and Loyalty Mapping the Customer Journey Importance of Customer Experience in the Digital Age Metrics for Measuring Customer Journey Success	**3**	Describe the stages of the customer journey and how to apply the marketing funnel to plan and execute effective marketing strategies.	2

III	**Digital Marketing Strategy and Planning:**Developing a Digital Marketing Strategy Setting SMART Goals for Digital Marketing Campaigns Channel Selection: SEO, SEM, Social Media, Email Marketing Content Planning and Development Aligning Digital Marketing Strategies with Business Objectives	**3**	Develop a comprehensive digital marketing strategy, including setting goals, selecting appropriate channels, and planning content that aligns with business objectives.	3
IV	**Implementing Digital Marketing Campaigns:**Executing SEO and SEM Campaigns Social Media Marketing: Platforms, Tools, and Best Practices Email Marketing: Campaign Types, Tools, and Techniques Display Advertising: Formats, Targeting, and Optimization Integrating Multiple Channels for a Cohesive Strategy	**3**	Execute and manage digital marketing campaigns across various platforms, utilizing tools and best practices to reach and engage target audiences effectively.	4
V	**Measuring and Optimizing Marketing Performance:**Key Performance Indicators (KPIs) for Digital Marketing Tools and Techniques for Measuring Campaign Success Data-Driven Decision Making: Using Analytics to Improve Campaigns Attribution Models in Digital Marketing Continuous Improvement: A/B Testing, Campaign Optimization, and Reporting	**3**	Evaluate and optimize digital marketing performance by analyzing key metrics, using data-driven insights to enhance strategies, and ensuring continuous improvement through testing and reporting.	5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the key concepts of digital marketing and e-commerce and how they create value for businesses.	**1, 2**
2	Create and execute digital marketing strategies that effectively attract and engage customers.	**3, 5**
3	Analyze customer data and marketing performance to make informed decisions that improve outcomes.	**4, 9**
4	Build and manage e-commerce stores that enhance the customer experience and drive sales.	**3, 6**
5	Utilize digital tools and platforms to implement and optimize marketing campaigns across various channels.	**5, 12**

SEMESTER – I									
Course Title	**BRIDGE COURSE I - FUNDAMENTALS OF GRAPH THEORY**								
Course code	**24MCAM1107R**	**Total credits: 2** **Total hours: 15T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To introduce students to the foundational concepts of graph theory and their applications in various domains. 2. To enable students to model and solve real-life problems using graph theory, focusing on graph properties and algorithms. 3. To develop the ability to analyze and apply different types of graphs to practical scenarios.								
CO1	Model problems using different types of basic graphs like trees, bipartite graphs, and planar graphs.								
CO2	Understand and identify special graphs such as Eulerian and Hamiltonian graphs.								
CO3	Analyze various forms of connectedness in a graph and their implications in solving problems.								
CO4	Apply graph coloring techniques to solve problems and understand their theoretical aspects.								
CO5	Model and analyze real-life problems as graph problems using the concepts learned.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Fundamental Concepts:** Introduction to Graph Theory: Basic Concepts, Definitions, Graph Types Adjacency and incidence matrices,Isomorphisms, paths, walks, cycles, components Cut-edges, cut-vertices, bipartite graphs, Eulerian and Hamiltonian graphs, vertex degrees, Reconstruction conjecture, extremal problems, degree sequences, Directed graphs, Orientations, and tournaments.	**7**	Remembering and understanding basic graph concepts and applying them to different problems.	1, 2, 3
II	**Trees:** Trees and forests, characterizations of trees, Spanning trees, radius and diameter, Enumeration of trees, Counting spanning trees, Deletion-contraction, The matrix tree theorem, Minimum spanning trees Prim's and Kruskal's algorithm Single source shortest paths Bellman Ford, Single source shortest paths Dijkstra's algorithm	**7**	Applying and analyzing tree structures in graph theory.	1,2, 3,4
III	**Matching and Covers:** Matchings, maximal and maximum matchings M-augmenting paths, Hall's theorem and consequences, Min-max theorems, Maximum matchings and vertex covers, Independent sets and edge covers, Connectivity, vertex cuts Edge-connectivity	**8**	Applying and analyzing different types of matchings and covers in graphs.	3,4
IV	**Connectivity and Paths:**Cuts and Edges connectivity, Blocks, k-connected graphs, Menger's theorem, Network flow problems, Flows and source/sink cuts, Ford-Fulkerson	**7**	Describing and analyzing connectivity and path problems in graphs.	2,4

	algorithm, Maximum Network Flow			
V	**Graph Coloring:** Vertex colorings, Upper Bounds, Brooks Theorem, Bounds on chromatic numbers, Chromatic numbers of graphs constructed from smaller graphs, Chromatic polynomials, Properties of the chromatic polynomial, Chordal Graphs and A Hint of perfect Graphs **Planar Graphs:** Planar graphs, Euler's formula, Kuratowski's theorem, Five and four-color theorems, Line Graph and Edge coloring, Hamiltonian Cycle, Planarity, coloring and Cycles.	**8**	Understanding and analyzing graph coloring techniques and their applications.	1,2, 4

TEXT BOOKS:

T1: Fundamentals of Internet and www, Greenlaw R and Hepp E,Tata McGrawHill, 2007

REFERENCE BOOKS:

R1: The Internet Book, D Comer, Pearson Education, 2019.
R2: The Complete reference to Internet, M. L. Young, Tata McGraw Hill, 2007

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Review the current topics in Web & Internet technologies.	**1, 12**
2	Describe the basic concepts for network implementation.	**1, 5**
3	Learn the basic working scheme of the Internet and World Wide Web.	**1, 4**
4	Understand fundamental tools and technologies for web design.	**3, 5**
5	Implement principles of UX design to create intuitive and user-friendly web interfaces.	**3, 9**

SEMESTER – I									
Course Title	**BRIDGE COURSE II - ALGORITHM FOR PROBLEM SOLVING**								
Course code	**24MCAM1108R**	**Total credits: 2** **Total hours: 15T+15P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			1	**0**	**2**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To understand the basics of algorithm and computational knowledge. 2. To understand computational thinking. 3. To apply algorithmic problem-solving techniques to real world problems. 4. To write efficient code for solving computational challenges.								
CO1	Understand computational thinking and its four pillars.								
CO2	Understand and apply algorithms for various problems.								
CO3	Understand and analyze the principle of divide and conquer.								
CO4	Understanding graph theory and representation.								
CO5	Understanding linked list and trees in data structure.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to Computational Thinking:** About computational thinking, four pillars of computational thinking.	**8**	Understanding and applying computational thinking for different real-world computer-based problems.	1, 3
II	Introduction to algorithm: About algorithm, basic algorithmlikesearchandsort, pseudocode	**10**	Understanding and applying different algorithm and analyzing the algorithm	1, 3, 4
III	Divide and Conquer: Principles of divide and conquer,QuickSort,Merge Sort	**8**	Understanding, applying divide and conquer method, and analyzingquicksortandmerge sort.	1,3, 4
IV	Graph Algorithms: Basics of graph theory, Graph representation and traversal, Shortestpathalgorithm.	**10**	Understanding, applying graph structure, and analyzing shortest path algorithm.	1, 3, 4
V	DataStructures:Basicdata structures,linkedlist,trees, minimum spanning trees	**10**	Gain a clear understanding of what an algorithm is and why algorithms are fundamental in computer science and problem-solving.	2, 3, 4

TEXT BOOKS:

T1:"IntroductiontoAlgorithms"byThomasH.Cormen,CharlesE.Leiserson,RonaldL.Rivest,and Clifford Stein.

T2:“ComputationalThinking:ABeginner'sGuidetoProblem-SolvingandProgramming”byKarl Beecher.

T3:"CProgrammingAbsoluteBeginner'sGuide"byPerryandMiller

REFERENCE BOOKS:

R1:"Code:TheHiddenLanguageofComputerHardwareandSoftware"byCharlesPetzold.

R2: "Algorithms to Live By: The Computer Science of Human Decisions" by Brian Christian and Tom Griffiths.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand computational thinking and its four pillars.	**1, 3**
2	Understand and apply algorithms for various problems.	**2, 3, 4**
3	Understand and analyze the principle of divide and conquer.	**1, 2, 7**
4	Understanding graph theory and representation.	**1, 2, 4, 7**
5	Understanding linked list and trees in data structure.	**1, 2, 5, 7**

SEMESTER – I									
Course Title	**CO-CURRICULAR ACTIVITIES**								
Course code	**24UBCC1101R**	**Total credits: 1** **Total hours: 15**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. Enhance holistic development by integrating essential life skills such as teamwork, leadership, and time management through co-curricular activities. 2. Foster social and emotional growth by promoting inclusivity, cultural awareness, and empathetic interactions within the community. 3. Encourage exploration and innovation by providing opportunities for students to discover new interests and nurture a spirit of curiosity and lifelong learning.								
CO1	Demonstrate enhanced teamwork and leadership abilities by actively participating in group projects and collaborative activities.								
CO2	Exhibit improved social and emotional intelligence through participation in activities that promote empathy, cultural understanding, and inclusivity.								
CO3	Apply time management and organizational skills in planning and executing co-curricular events and projects.								
CO4	Showcase creativity and innovation by developing new ideas and solutions in various co-curricular initiatives.								
CO5	Reflect on personal growth and learning experiences gained through engagement in diverse co-curricular activities, leading to lifelong learning and self-improvement.								
Content									
AdtU encourages activities aimed at developing students' social and soft skills, promoting holistic development. Keeping in mind the 360-degree learning methodology, students are engaged in various activities beyond regular classes. Experts are invited to conduct workshops that benefit students, while technical quizzes are conducted to enhance their knowledge. Technical seminars help develop their public speaking abilities.									

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Able to explain the fundamentals and advances of cytology including structure and functions of cell and cell organelles.	**1,3 & 4**
2	Able to explain the cell cycle and cell division.	**1,2**
3	Learn and develop skills for operating microscope, preparing slides by various staining techniques	**7.9,10**
4	Apply knowledge of cellular processes to explain how cells operate and interact within living organisms.	**5,7**
5	Demonstrate a comprehensive understanding of cell structure and function.	**5,8**

SEMESTER – II									
Course Title	**CLOUD FUNDAMENTALS**								
Course code	**24MCAM1201R**	**Total credits: 5** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**4**	**0**	**0**	**0**	**5**
Pre-requisite	**Basic Programming Knowledge**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To provide an overview of an exciting field of Cloud Computing 2. To introduce tools requires building, deploying, running and managing applications on a cloud platform. 3. To develop the cloud application development skills, such as Python, REST architecture, JSON, Cloud Foundry and DevOps services 4. To enable students to have skills that will help them to solve complex real-world problems in decision support.								
CO1	Understand fundamental cloud computing concepts and architectures.								
CO2	Analyze and implement RESTful APIs and data services on cloud platforms.								
CO3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.								
CO4	Develop and deploy applications using Python and related frameworks.								
CO5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Introduction to Cloud Computing and IBM Cloud:** Defining cloud computing, Factors leading to cloud adoption, Types of cloud services (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid) Overview of IBM Cloud Creating an IBM Cloud account, Navigating the IBM Cloud dashboard.	**9**	Understand the fundamental concepts of cloud computing, including service models (IaaS, PaaS, SaaS), deployment models, and the core functionalities of IBM Cloud, while gaining hands-on experience in navigating the IBM Cloud platform.	1,2
II	**IBM Cloud Core Services ordata services on IBM Cloud:** Watson AI services on IBM Cloud, Introduction to DevOps, Implementing DevOps practices on IBM Cloud, Using IBM Cloud toolchains for DevOps, REST API, Types and capabilities of databases, Apis interaction with Cloudant database.	**9**	Implement IBM Cloud core services, including Watson AI and DevOps practices, to develop and manage cloud applications efficiently, leveraging REST APIs and Cloudant database interactions for seamless data handling.	2,3
III	**Application Development on IBM Cloud:** Understanding business problems and goals, Functional and non-functional requirements, IBM Cloud App ID, Container orchestration (Kubernetes), Pods, Deployment, and Service, Cloud Foundry.	**9**	Design and deploy cloud-based applications using IBM Cloud technologies, incorporating container orchestration with Kubernetes, Cloud Foundry, and IBM Cloud App ID for authentication and application management.	1,2

IV	**Developing Python Applications for the Cloud:** Building Python applications for cloud deployment Integrating popular Python frameworks (Django, Flask) with cloud services Leveraging cloud storage, databases, and AI services with Python Implementing scalability and high availability for Python applications on the cloud	**9**	Develop scalable and high-performing Python applications for cloud deployment, integrating frameworks like Django and Flask with cloud storage, databases, and AI services while ensuring reliability and performance.	1,2
V	**Security and Compliance on IBM Cloud:** Overview of IBM Cloud security features, Implementing security best practices on IBM Cloud, Ensuring compliance on IBM Cloud, OAuth protocol.	**9**	Ensure security and compliance on IBM Cloud by implementing best practices, understanding IBM Cloud's security features, adhering to regulatory requirements, and utilizing authentication protocols such as OAuth for secure access management.	1,2

Text Books:

1. Cloud Computing: Concepts, Technology & Architecture by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood.
2. Cloud Computing: A Hands-On Approach by Arshdeep Bahga and Vijay Madisetti.
3. Cloud Native Architectures: Design High-availability and Cost-effective Applications for the Cloud by Tom Laszewski, Kamal Arora, and Erik Farr.

Reference Books:

1. Cloud Computing Principles and Paradigms by Rajkumar Buyya, James Broberg, and Andrzej Goscinski.
2. Cloud Computing: Principles, Systems and Applications by Nick Antonopoulos and Lee Gillam.
3. Cloud Computing: From Beginning to End by Ray J. Rafaels.

Additional Resources:

1. Cloud Essentials: CompTIA Authorized Courseware for Exam CLO-001 by Kalani Kirk Hausman and Susan L. Cook.
2. Cloud Computing: A Practical Approach by Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter.
3. IBM Cloud Essentials: V3 Certification Study Guide by William Rothwell.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand fundamental cloud computing concepts and architectures.	**1,3 & 4**
2	Analyze and implement RESTful APIs and data services on cloud platforms.	**1,2**
3	Design and deploy cloud applications using IBM Cloud services, including Kubernetes.	**3, 5, & 7**
4	Develop and deploy applications using Python and related frameworks.	**5,7, & 9**
5	Apply advanced cloud concepts and architectures to deploy applications on Kubernetes clusters.	**5,8, & 10**

SEMESTER – II									
Course Title	**ADVANCED DATABASE MANAGEMENT SYSTEM**								
Course code	**24MCAM1102R**	**Total credits: 4** **Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**4**	**0**	**0**	**0**	**4**
Pre-requisite	**Database Management System**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To learn the fundamentals of data models and to represent a database system using ER diagrams. 2. To understand the internal storage structures using different file and index techniques which will help in physical database design. 3. To understand the concepts of database administration, Homogeneous and Heterogeneous Systems and knowledge based and database systems.								
CO1	Understand core database concepts, including data, information, metadata, and components of a Database Management System								
CO2	Understand and apply various data modelling concepts								
CO3	Understand and apply relational database concepts, relational algebra and SQL								
CO4	Implement various normalization techniques ensuring efficient data organization								
CO5	Understand the concepts of deadlocks, database security and distributed database systems								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	Database concept: Data, information metadata, Terminology of File, Association Between Fields, Entities and their attributes, Relationship Record and Files, Components of Database Management System, Mapping Between View, Data Independence.	**6**	Understanding of key database concepts including data modelling, entity-relationship modelling, and schema design.	1, 2, 3
II	Data Models: Entities, Attributes and Associations, Relationship Among Entities, Data Model Classification- Relational Model, Hierarchical Model & Network Model, Entity–Relationship Model, Concept of File Organization – Sequential Files, Index-Sequential Files, Direct Files, Secondary Key Retrieval.	**6**	Develop a foundational understanding of data modelling concepts, including entities, attributes, relationships, and the classification of data models such as relational, hierarchical, and network models.	3, 4
III	Relational Database: Attributes and Domains, Tuples, Relation and their Schemas, Relational Algebra: Basic Operations, Relational Algebra queries, Relational Calculus: Tuple Calculus, Domain Calculus, SQL, Data Manipulations in SQL, Data Manipulations in QBE	**6**	Design, query, and manipulate data within relational database systems using both SQL and QBE approaches.	3, 4
IV	Functional Dependency and Normalization: First, Second, Third Normal Forms, Good and Bad Decomposition, Multivalued Dependency, Fourth Normal Form, Fifth Normal Form, Network Data Model	**6**	Understand and apply database design principles, normalization techniques, and an understanding of alternative data models like the network data model.	2, 3, 4

V	Database operations and maintenance: Database Administrator (DBA, Database Security, Integrity and Control, Distributed Database: Data Distribution, Deadlock in Distributed Systems, Security and Protection, Homogeneous and Heterogeneous Systems, Knowledge Base and Database Systems, Expert Database Systems, Object Database System.	**6**	Understanding of essential database management principles and practices, including the role of Database Administrators (DBAs), database security measures, integrity constraints, and control mechanisms.	2, 3, 4
Practical				
Practical 1	You are given the following table structure for a database that stores information about students in a university. Write an SQL statements to create this table. Table: Students • StudentID (Primary Key): Integer, Auto Increment • FirstName: Varchar(50), Cannot be NULL • LastName: Varchar(50), Cannot be NULL • DateOfBirth: Date • Email: Varchar(100), Unique, Cannot be NULL EnrollmentDate: Date, cannot be NULL	**2**	Create a database table with specified constraints and properties using SQL.	1, 5
Practical 2	Write the SQL statement to create the Books table with the given structure. Table: Books • BookID (Primary Key): Integer, Auto Increment • Title: Varchar(100), Cannot be NULL • Author: Varchar(100), Cannot be NULL • PublishedYear: Year • Genre: Varchar(50) • ISBN: Varchar(13), Unique, Cannot be NULL Write SQL queries to: 1. Insert a new book into the Books table. 2. Retrieve all books written by a specific author. 3. Update the genre of a book based on its BookID.	**2**	Create a database table and perform basic SQL operations such as inserting data, retrieving data, and updating records.	1, 5
Practical 3	On the Books table, write SQL queries to: 1. Delete a book from the table by BookID 2. Retrieve all books published after the year 2000 3. Retrieve the count of books by each author	**2**	Create a database table and perform basic SQL operations	2

Practical 4	Create the following tables with the fields given below: EMPLOYEE (E_ID, Name, Age, Address, Salary, Dep_ID) DEPARTMENT (Dep_ID, Name, City) Select appropriate primary keys. Input at least 5 meaningful records in the tables. Select appropriate data types for all the fields.	**2**	Demonstrate the ability to create database tables with appropriate fields and primary keys.	3
Practical 5	Create the following tables with the fields given below : STUDENT (Student ID, Name, Programme, Teacher ID) TEACHER (Teacher ID, Department, Name, City, Specialization) Select appropriate primary keys. Input at least 5 meaningful records in the tables. Select appropriate data types for all the fields.	**2**	Create tables with specified fields and primary keys.	3
Practical 6	EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id) (a) Find out the employee id, names, salaries of all the employees (b) List out the employees who works under manager 100 (c) Find the names of the employees who have a salary greater than or equal to 4800 (d) List out the employees whose last name is 'AUSTIN' (e) Find the names of the employees who works in departments 60, 70 and 80 (f) Display the unique Manager_Id	**2**	Execute and analyse SQL queries for specific data retrieval	3
Practical 7	Create Client_master table with the following fields(ClientNO, Name, Address, City, State, bal_due) (a) Insert five records (b) Find the names of clients whose bal_due > 5000 . (c) Change the bal_due of ClientNO " C123" to Rs. 5100 (d) Change the name of Client_master to Client12. (e) Display the bal_due heading as "BALANCE"	**2**	Perform SQL operations to update records in table including inserting new records, updating balances, and modifying table properties	2, 3
Practical 8	Create Teacher table with the following fields (Name, DeptNo, Date of joining, DeptName, Location, Salary)	**2**	Apply SQL transactions using rollback and commit commands	3

	(a) Insert five records (b) Give Increment of 25% salary for Mathematics Department. (c) Perform Rollback command (d) Give Increment of 15% salary for Commerce Department (e) Perform commit command			
Practical 9	Create Sales table with the following fields (Sales No, Salesname, Branch, Salesamount, DOB) (a) Insert five records (b) Calculate total sales amount in each branch (c) Calculate average sales amount in each branch. (d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09 (e) Display the name and DOB of salesman in alphabetical order of the month.	**2**	Apply SQL clauses such as ORDER BY and GROUP BY to perform aggregate calculations	2, 3
Practical 10	Create an Emp table with the following fields: (EmpNo, EmpName, Job, Basic, DA, HRA, PF, GrossPay, NetPay) (Calculate DA as 30% of Basic and HRA as 40% of Basic) (a) Insert Five Records and calculate GrossPay and NetPay. (b) Display the employees whose Basic is lowest in each department. (c) If NetPay is less than <Rs. 10,000 add Rs. 1200 as special allowances . (d) Display the employees whose GrossPay lies between 10,000 & 20,000 (e) Display all the employees who earn maximum salary	**2**	Execute calculations and SQL queries for payroll management	2, 3
Practical 11	Create a table, fill it with customer details and do the following queries: a) Find the total revenue for each customer b) List customers who have made more than one order c) Find the date of the most recent order for each customer d) Calculate the average order amount for each customer e) Retrieve the orders placed in January 2023 f) Find the top 3 customers with the highest total revenue g) Calculate the total revenue for each month in 2023 h) List orders with a total amount greater than the average total amount for all orders i) Find customers who have made orders on	**2**	Analyse and interpret SQL query results.	2, 3

	consecutive days j) Calculate the total revenue for each customer, including orders made by customers who haven't placed any orders.			
Practical 12	An Enterprise wishes to maintain a database to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas Dept (deptno, dname, loc) Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno) a) Update the employee salary by 15%, whose experience is greater than 10 years. b) Delete the employees, who completed 30 years of service. c) Display the manager who is having maximum number of employees working under him? d) Create a view, which contain employee names and their manager	**2**	Implement SQL operations and create database views	3
Practical 13	Using Employee Database perform the following queries a) Determine the names of employee, who earn more than their managers. b) Determine the names of employees, who take highest salary in their departments. c) Determine the employees, who are located at the same place. d) Determine the employees, whose total salary is like the minimum Salary of any department. e) Determine the department which does not contain any employees.	**2**	Formulating and evaluating complex SQL queries	3
Practical 14	Create the following tables: Student (roll-no, name, date-of-birth, course-id) Course (Course-id, name, fee, duration) a) Create a form to accept the data from the user with appropriate validation checks. b) Generate queries to do the following: List all those students who are between 18-19 years of age and have opted for MCA course. List all those courses in which number of students are less than 10.	**2**	Designing and analysing SQL queries for complex conditions	3
Practical 15	Create the following tables: Branch (branch-id, branch-name, branch-city) Customer (customer- id, customer-name, customer-city, branch-id) a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following:	**2**	Analysing and interpreting SQL results based on specified conditions	3

	c) List all those customers who live in the same city as the branch in which they have account. d) List all those customers who have an account in more than one branch.			
Practical 16	Create the following tables: Book (accession-no, title, publisher, year, date-of-purchase, status) Member (member-id, name, number-of-books-issued, max-limit) Book-issue (accession-no, member-id, date-of-issue) (a) Create a form to accept the data from the user with appropriate validation checks. (b) Generate queries to do the following: (c) List all those books which are due from the students to be returned. A book is due if it has been issued 15 days back and yet not returned. (d) List all those members who cannot be issued any more books	**2**	Implementing SQL queries for library management	3
Practical 17	Create a Database of Library Management System	**2**	Designing and implementing database schema for Library domain	2, 3
Practical 18	Create a Database of Hospital Management System.	**2**	Designing and implementing database schema for healthcare domain	2, 3
Practical 19	Create a Database of Railway Reservation System.	**2**	Designing and implementing database schema for transportation domain	2, 3
Practical 20	Create a Database of Time Table Management System.	**2**	Designing and implementing database schema for educational scheduling	2, 3

TEXT BOOKS:

T1: Silberschatz A, Korth HF, Sudarshan S. Database System Concepts. 6th edition. New York: McGraw-Hill; 2017.

T2: Elmasri R, Navathe SB. Fundamentals of Database Systems. 6th edition. Boston: Pearson Education; 2014.

REFERENCE BOOKS:

R1: Ramakrishnan R. Database Management System. 3rd edition. New Delhi: Tata McGraw-Hill Publishing Company; 2007.

R2: Rob P, Coronel C. Database Systems: Design, Implementation, and Management. 5th edition. Boston: Thomson Learning Course Technology; 2003.

R3: Garcia-Molina H, Ullman JD, Widom J. Database System Implementation. 1st edition. United States: Pearson Education; 2000.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand core database concepts, including data, information, metadata, and components of a Database Management System	**1, 2, 5, 12**
2	Understand and apply various data modelling concepts	**1, 2, 3, 5, 12**
3	Understand and apply relational database concepts, relational algebra, and SQL	**1, 2, 3, 5, 12**
4	Implement various normalization techniques ensuring efficient data organization	**1, 2, 3, 5, 12**
5	Understand the concepts of deadlocks, database security and distributed database systems	**1, 2, 5, 12**

SEMESTER – II									
Course Title	**OPERATING SYSTEMS CONCEPTS AND DESIGN**								
Course code	**24MCAM1203R**	**Total credits: 4** **Total hours: 45T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To learn the mechanisms of OS to handle processes and threads and their communication and the mechanisms involved in memory management in contemporary OS. 2. To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols 3. To know the components and management aspects of concurrency management.								
CO1	Understand the concepts of OS, the basic principles used in the design of modern operating system and process.								
CO2	Understanding the concepts of processes, process scheduling including Throughput, Turnaround Time, Waiting Time, Response Time.								
CO3	Understand the concepts of threads and mechanisms for synchronization.								
CO4	Understand the concepts related to deadlock and memory management.								
CO5	Understand the concepts of virtual memory management, file system.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction**:Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**12**	Understand the concept of operating systems, distinguish between different generations and types of operating systems, explain the services provided by an operating system, and compare the structures of layered, monolithic, and microkernel operating systems.	2
II	**Processes**:Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. **Thread**: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.	**12**	Understand the definition of processes and threads, describe the different states and transitions of processes, explain the benefits and types of threads, discuss the objectives and criteria of process scheduling, and analyse various scheduling algorithms including FCFS, SJF, and RR.	2, 3
III	**Inter-process Communication**: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution **Deadlocks**: Definition, Necessary and	**12**	Identify critical sections and race conditions, explain mutual exclusion and its hardware solutions, define deadlocks and its	5

	sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.		necessary conditions, discuss deadlock prevention techniques like Banker's algorithm, and describe deadlock detection and recovery methods.	
IV	**Memory Management**: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition –Internal and External fragmentation and Compaction; Paging. **Virtual Memory**: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).	**12**	Understand the basic concepts of memory management, differentiate between logical and physical address mapping, analyse memory allocation techniques including contiguous allocation and paging, explain virtual memory concepts such as demand paging and page replacement algorithms (e.g., LRU, FIFO), and discuss the advantages of virtual memory.	4
V	**File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management. Disk Management: Disk structure, Disk scheduling, Disk reliability, Disk formatting, Boot-block, Bad blocks.	**12**	Understand the concept of files, various file types and access methods, file operations and directory structures, analyse file system structures and allocation methods (contiguous, linked, indexed), and evaluate disk management aspects including disk structure, scheduling algorithms, reliability, and formatting.	5

TEXT BOOKS:

T1: Silberschatz A, Galvin P, Gagne G. Operating System Concepts Essentials. 9th edition. Singapore: Wiley Asia Student Edition; 2018.

REFERENCE BOOKS:

R1: Crowley C. Operating System: A Design-Oriented Approach. 1st edition. Boston: Irwin Publishing; 1996.

R2: Nutt GJ. Operating Systems: A Modern Perspective. 2nd edition. Boston: Addison-Wesley; 2000.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the concepts of OS, the basic principles used in the design of modern operating system and process.	**1, 2, 3, 5, 12**
2	Understanding the concepts of processes, process scheduling including - Throughput, Turnaround Time, Waiting Time, Response Time.	**1, 2, 4,**
3	Understand the concepts of threads and mechanisms for synchronization.	**3, 5, 12**
4	Understand the concepts related to deadlock and memory management.	**1, 2, 12**
5	Understand the concepts of virtual memory management, file system.	**2, 3, 5, 12**

SEMESTER – II										
Course Title	**PG PROJECT II**									
Course code	**24MCAM1204R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**6**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives	1. To develop the practical skill 2. To enhance Critical Thinking and Problem-Solving Abilities 3. To Improve Communication and Collaboration Skills								
CO1	Demonstrate a sound technical knowledge of the selected project topic.								
CO2	Illustrate problem identification, formulation, and solution.								
CO3	Plan solutions to complex problems utilizing a systems approach.								
CO4	Design the prototype of the solution and analyse the prototype using testing methods.								
CO5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Project Introduction and Planning:** Overview of project goals and objectives. Importance of project planning and management. Project Proposal Development: Identifying a project topic or problem Conducting preliminary research Writing a project proposal Setting timelines and milestones Resource allocation (materials, tools, etc.) Risk assessment and management strategies	**6**	Describe, illustrate, and explain problem	2, 3
II	**Literature Review and Research Methodology:** Identifying and reviewing relevant literature Summarizing and synthesizing existing research Highlighting gaps and formulating research questions Choosing appropriate research methods (qualitative, quantitative, or mixed methods) Designing research instruments (surveys, interviews, experiments, etc.) Ethical considerations in research	**6**	Describe, illustrate and Literature Review	2, 3, 4
III	**Data Collection and Analysis:** Gathering primary data (surveys, experiments, interviews, observations) Ensuring data accuracy and reliability, Storing and organizing data Using statistical tools and software for data analysis, Interpreting data and drawing conclusions, Visualizing data (charts, graphs,	**6**	Describe, illustrate and explain data collection and analysis	2, 3, 4

	tables)			
IV	**Project Implementation and Monitoring:** Project Execution, Implementing the project plan Monitoring progress and making adjustments as needed Documentation and Record Keeping: Keeping detailed records of project activities and outcomes Troubleshooting and problem-solving during implementation, Interim Reporting: Preparing and presenting progress reports Receiving and incorporating feedback	**6**	Describe, illustrate and explain project implementation and execution	6
V	**Project Presentation and Evaluation Final Reporting:** Writing the final project report (structure, content, formatting) Ensuring clarity, coherence, and completeness Project Presentation: Preparing visual aids (slides, posters, etc.) Practicing presentation skills (public speaking, answering questions) Evaluation and Reflection: Assessing project outcomes against objectives Reflecting on the project process and personal learning Identifying areas for future improvement and potential follow-up projects	**6**	Describe, illustrate and explain testing and documentation	2,3, 5

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate a sound technical knowledge of selected project topic.	**1, 2, 4**
2	Illustrate problem identification, formulation and solution.	**1, 2**
3	Plan solutions to complex problems utilizing a systems approach.	**1, 2, 3, 5, 7**
4	Design the prototype of the solution and analyse the prototype using testing methods.	**1, 2, 3, 5, 6, 7, 8**
5	Improve ethical and professional sustainability while working in a team and communicate effectively for the benefit of the society.	**1, 2, 3, 6, 8**

SEMESTER – II									
Course Title	**PDP II - COMMUNICATION MASTERY**								
Course code	**24UMPD1202R**	**Total credits: 4** **Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**4**	**0**	**0**	**2**
Pre-requisite	**Basic Programming Knowledge**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Fall/ I semester of first year of the programme**								
Course Objectives (Minimum 3)	1. Foster effective communication and interpersonal skills for professional and personal success. 2. Equip individuals with the skills to create compelling job application documents and excel in job interviews. 3. Empower individuals to communicate confidently and effectively in public and professional settings.								
CO1	Enable the students to take initiative, guide the discussion, and influence others positively.								
CO2	Improve student's ability in framing different sentences while speaking and writing.								
CO3	Develop writing skills in different areas including paragraph and email.								
CO4	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills								
CO5	Encompass personal growth, career advancement, enhanced communication, and the ability to navigate a variety of professional situations successfully.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Techniques of Effective Reading, Different types of reading. Process of Listening, Types of Listening Understanding Listening Barriers Importance of Effective listening	**12**	Equipped with a diverse set of reading skills that can be applied across various contexts and types of texts, promoting both efficiency and comprehension.	1, 2, 3
II	Understanding difference between Resume, CV and Cover Letter. Creating Resume & CV using different online platforms	**12**	Leverage various online platforms effectively to create professional and visually appealing resumes and CVs,	2, 3
III	Introduction to public speaking Preparation for Public speaking (scripts, non-verbal cues) 1. Understanding and overcoming Fear of Public Speaking 2. Tips Public Speaking 3. Public speaking practical	**12**	Enhance their public speaking abilities, becoming more confident, engaging, and effective communicators.	2, 3, 4
IV	1. Types of interviews- telephonic, virtual & face to face, online interview, personal interview, Panel interview, Group interview. 2. Common interview questions and answering strategies	**12**	Understand consistent use of effective answering strategies help candidates create a positive impression, align their qualifications with the job requirements, and increase	2, 3, 4

			their chances of success in the interview process.	
V	1. Introduction to Dress Code Ethics 2. Purpose and Importance 3. How to Make FIRST Impression What to Wear During Interviews or Any Other Formal Meetings – Male & Female	**9**	Enhanced self-awareness, adaptability, confidence, effective communication, knowledge and skills necessary to present themselves professionally.	1, 5, 6

TEXT BOOKS:

T1: Malarcher C. Developing Listening Skills 1. Seoul: Compass Publishing;

T2: Anderson LE, Bolt SB. Professionalism Skills for Workplace Success. Boston: Pearson Education;

T3: Carnegie D. The Art of Public Speaking. New Delhi: Diamond Pocket Books Pvt Ltd;

T4: Wallwork A. English for Academic CVs, Resumes, and Online Profiles. Cham: Springer International Publishing;

REFERENCE BOOKS:

R1: McKinney A. Real-Resumes for Teachers. Raleigh: Prep Pub;

R2: Williams JW. Listening Skills Training: How to Truly Listen, Understand, and Validate for Better and Deeper Connections. Seattle: Amazon Digital Services LLC - KDP Print US;

OTHER LEARNING RESOURCES:

https://www.coursera.org/specializations/english-interview-resume

https://www.coursera.org/learn/public-speaking

https://www.coursera.org/learn/enpublicspeaking

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enable the students to take initiative, guide the discussion, and influence others positively.	**6, 9, 10**
2	Improve student's ability in framing different sentences while speaking and writing.	**9, 12**
3	Develop writing skills in different areas including paragraph and email.	**9, 12**
4	Plan efficiently for discussions in different platforms by enhancing their thought process and problem-solving skills	**2, 3, 6, 9, 12**
5	Encompass personal growth, career advancement, enhanced communication, and the ability to navigate a variety of professional situations successfully.	**6, 9, 11, 12**

SEMESTER – II									
Course Title	**TEACHING ASSISTANTSHIP**								
Course code	**24MCAM1205R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**4**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	4. Develop teaching skills, including lesson planning, instructional delivery, and classroom management, to support effective learning. 5. Enhance communication and mentoring abilities to guide and assist students in understanding complex concepts. 6. Gain hands-on experience in grading, providing constructive feedback, and managing academic responsibilities in a professional setting.								
CO1	Demonstrate the ability to effectively deliver instructional content and facilitate student learning.								
CO2	Develop strong communication and mentoring skills to assist students in their academic progress.								
CO3	Apply pedagogical techniques to engage students and enhance their understanding of subject matter.								
CO4	Gain experience in assessment, grading, and providing constructive feedback to students.								
CO5	Understand the ethical responsibilities and professional conduct required in a teaching assistantship role.								
The Teaching Assistantship course provides students with practical experience in instructional support, helping them develop essential teaching and mentoring skills. Through hands-on engagement in lesson planning, classroom facilitation, and student assessment, participants enhance their ability to communicate complex ideas effectively. The course also focuses on professional responsibilities, ethical considerations, and fostering a collaborative learning environment. By the end of the course, students will be equipped with the necessary tools to support educators, assist students, and manage academic responsibilities efficiently.									

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate the ability to effectively deliver instructional content and facilitate student learning.	**1,3 & 4**
2	Develop strong communication and mentoring skills to assist students in their academic progress.	**1,2**
3	Apply pedagogical techniques to engage students and enhance their understanding of subject matter.	**7.9,10**
4	Gain experience in assessment, grading, and providing constructive feedback to students.	**5,7**
5	Understand the ethical responsibilities and professional conduct required in a teaching assistantship role.	**5,8**

SEMESTER – II									
Course Title	**LOGICAL REASONING AND QUANTITATIVE APTITUDE**								
Course code	**24MCAM1206R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To help students to develop Analytical Thinking and Problem-Solving Skills. 2. To make it possible for students to Enhance Quantitative and Data Interpretation Abilities. 3. Prepare for Industry-Specific Recruitment Processes.								
CO1	Demonstrate the ability to solve quantitative problems involving arithmetic, algebra, geometry, and data analysis with speed and accuracy.								
CO2	Apply logical reasoning and critical thinking skills to analyze patterns, sequences, and relationships to solve complex puzzles and problems.								
CO3	Exhibit proficiency in solving real-world problems by interpreting data, understanding trends, and making informed decisions using quantitative techniques.								
CO4	Successfully crack industry-level aptitude tests and placement exams by applying advanced problem-solving strategies and time management skills.								
CO5	Communicate solutions effectively and confidently during interviews, group discussions, and professional scenarios, showcasing readiness for industry roles.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	Number Systems and Arithmetic	4	Understand and apply concepts of number systems, LCM, HCF, percentages, profit and loss, and averages.	1,2
II	Algebra and Geometry	4	Solve problems involving equations, inequalities, ratios, proportions, and basic geometry.	2,3
III	Logical Reasoning and Puzzles	5	Develop logical thinking by solving puzzles, seating arrangements, syllogisms, and series completion.	1,2
IV	Data Interpretation and Analysis	4	Interpret data presented in charts, tables, and graphs to make informed decisions and solve problems.	1,2
V	Industry-Specific Aptitude Preparation	5	Prepare for company-specific aptitude tests with practice on previous papers and mock tests.	1,2

Text Books:

T1. Quantitative Aptitude for Competitive Examinations by R.S. Aggarwal
T2. A Modern Approach to Verbal & Non-Verbal Reasoning by R.S. Aggarwal
T3. How to Prepare for Quantitative Aptitude for the CAT by Arun Sharma

Reference Books:

R1. Logical Reasoning and Data Interpretation for the CAT by Nishit K. Sinha
R2. Fast Track Objective Arithmetic by Rajesh Verma
R3. Puzzles to Puzzle You by Shakuntala Devi

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate the ability to solve quantitative problems efficiently.	**1**
2	Apply logical reasoning to analyze patterns and relationships.	**2**
3	Exhibit proficiency in interpreting and analyzing data.	**3**
4	Successfully crack industry-level aptitude tests and placements.	**4**
5	Communicate solutions effectively during interviews and discussions.	**5**

SEMESTER – II									
Course Title	**Field Visit**								
Course code	**24MCAM1208R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**8**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To cultivate an appreciation for the complexities and challenges of professional practice in the field.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyze and interpret field data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								
Course Contents	The organization will provide the students two field visit for this program. The students will be taken to a specific site for learning purposes. The students will visit a relevant site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners regarding the subject or purpose of the visit. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply theoretical concepts to real-world situations.	**1, 3, 10**
2	Conduct field-based research and gather data effectively.	**4, 5**
3	Analyze and interpret field data to draw meaningful conclusions.	**2, 4, 10**
4	Communicate findings clearly and effectively, both orally and in writing.	**8, 6**
5	Demonstrate professional behaviour and teamwork skills in field settings.	**6, 7**

SEMESTER – II									
Course Title	**BRIDGE COURSE III - FUNDAMENTALS OF DISCRETE MATHEMATICS**								
Course code	**24MCAM1209R**	**Total credits: 2** **Total hours: 30T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To provide students with a solid understanding of the fundamental concepts in discrete mathematics and graph theory. 2. To develop students' ability to solve problems related to discrete structures and graph algorithms. 3. To enable students to apply discrete mathematical concepts and graph theory techniques to computer science and related fields.								
CO1	Understand set theory, propositional logic, and proof techniques for logical reasoning.								
CO2	Apply combinatorics and probability to solve counting and probabilistic problems.								
CO3	Understand the fundamental concepts of graph theory, including graph representations, isomorphism, and tree structures.								
CO4	Students will understand spanning trees, connected and disconnected graphs, Eulerian and Hamiltonian paths, planar graphs, and graph coloring.								
CO5	Students will learn Boolean Algebra concepts, simplify expressions, apply logic gates, and design switching circuits and logic systems.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Foundations of Discrete Mathematics:** Set Theory: Sets, subsets, operations on sets, Venn diagrams, Cartesian products. Logic: Propositional logic, predicates, quantifiers, logical connectives, truth tables. Proof Techniques: Direct proof, proof by contradiction, induction, contrapositive.	**4**	Understand the basics of set theory, including sets, subsets, operations, Venn diagrams, and Cartesian products. They will learn propositional logic, predicates, quantifiers, logical connectives, and truth tables, along with mastering proof techniques such as direct proof, proof by contradiction, induction, and contrapositive.	**1,2, 3**
II	**Combinatorics and Discrete Probability:** Combinatorics: Permutations, combinations, Pigeonhole principle, inclusion-exclusion principle. Discrete Probability: Probability theory, conditional probability, Bayes' theorem, expected value.	**8**	Understand permutations, combinations, the Pigeonhole principle, and the inclusion-exclusion principle. They will also learn probability theory, conditional probability, Bayes' theorem, and how to calculate the expected value.	**2,5**
III	**Graph Theory Basics:** Types of graphs (simple, directed, weighted, multigraphs), degree of vertices, adjacency matrix, incidence matrix Isomorphisms, paths, walks, cycles,	**10**	Understand and classify various types of graphs, including simple, directed, weighted, and multigraphs. They will analyze graph structures through the	**1,2, 3,4**

	connectedness,acyclic graph,Cut-edges, cut-vertices, bipartite graphs,graph and map coloring.		degree of vertices and represent them using adjacency and incidence matrices. Students will explore graph isomorphisms, paths, walks, and cycles, and evaluate concepts like connectedness, acyclic graphs, cut-edges, and cut-vertices.	
IV	**More on Graphs:** Content: Trees and forests, characterizations of trees,properties of trees,infinite trees, Spanning trees,enumeration of spanning trees. Connected and disconnected graphs, components, Eulerian and Hamiltonian paths and circuits. Planar graphs, Euler's formula, Kuratowski's theorem,Planarity,coloring and cycles	**7**	Learn about trees and forests, their characterizations and properties, infinite trees, spanning trees, and enumeration of spanning trees. They will differentiate between connected and disconnected graphs, identify components, and analyze Eulerian and Hamiltonian paths and circuits. Additionally, they will study planar graphs, Euler's formula, Kuratowski's theorem, planarity, and graph coloring with cycles.	**2,4, 6**
V	**Basics of Boolean Algebra:** Introduction to Boolean Algebra- definition and basic postulates,Truth tables and basic operations (AND, OR, NOT), Laws and Properties of Boolean Algebra:Commutative, associative, distributive, identity, and complement laws , Simplification of Boolean Expressions using algebraic methods and truth tables. Logic Gates - Basic gates (AND, OR, NOT) and universal gates (NAND, NOR) , Applications of Boolean Algebra in switching circuits and simple logic design	**7**	Gain a foundational understanding of Boolean variables and functions, including truth tables and basic operations like AND, OR, and NOT. They will master key laws such as commutative, associative, distributive, and complement laws, and learn to simplify Boolean expressions using algebraic techniques. Additionally, students will understand the functionality of basic and universal logic gates and explore real-world applications in switching circuits and simple logic design.	**2,3**

TEXT BOOKS:

T1: Discrete Mathematics and Its Applications" by Kenneth H. Rosen
T2: Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited.
T3: "Introduction to Graph Theory" by Douglas B. West.

REFERENCE BOOKS:

R1: Discrete Mathematics for Computer Scientists & Mathematicians, Joe L. Mott, Abraham Kandel, Theodore P. Baker, Pearson, 2nd ed.

OTHER LEARNING RESOURCES:

MIT OpenCourseWare's lecture notes on Discrete Mathematics and Graph Theory - https://ocw.mit.edu/courses/18-225-graph-theory-and-additive-combinatorics-fall-2023/resources/lecture-notes/

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand set theory, propositional logic, and proof techniques for logical reasoning.	**1, 2, 12**
2	Apply combinatorics and probability to solve counting and probabilistic problems.	**1,2, 12**
3	Understand the fundamental concepts of graph theory, including graph representations, isomorphism, and tree structures.	**1, 2, 12**
4	Analyze advanced graph algorithms and their applications in network flows, planarity, and graph coloring.	**1, 2, 12**
5	Explore practical applications of discrete mathematics in computer science, network theory and cryptography.	**1, 2, 12**

SEMESTER – II									
Course Title	**BRIDGE COURSE IV - MATHEMATICS FOR BASIC PROGRAMMING**								
Course code	**24MCAM1210R**	**Total credits: 2** **Total hours: 30T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application in Artificial Intelligence and Deep Learning**								
Semester	**Winter/ II semester of first year of the programme**								
Course Objectives (Minimum 3)	1. To build a strong foundation in mathematics and its application in C programming for problem-solving and algorithm development. 2. To enable students to write efficient programs using arithmetic, logic, control flow, arrays, matrices, and number systems. 3. To develop skills in implementing mathematical solutions for real-world applications, including data processing and optimization tasks.								
CO1	Apply arithmetic operators and mathematical expressions to build programs for real-world applications like calculators and geometry problems.								
CO2	Use Boolean algebra and conditional expressions to implement decision-making and problem-solving programs								
CO3	Implement loops and nested structures to solve mathematical problems, including sequences, prime checking, and matrix patterns.								
CO4	Convert between number systems and optimize programs using bitwise operators and binary manipulations.								
CO5	Use arrays and matrices for complex calculations like summation, searching, and matrix operations to handle and process data effectively.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	**Arithmetic and Basic Math Operations** Arithmetic Operators: Addition, subtraction, multiplication, division, and modulus. Operator precedence and associativity. Mathematical Expressions: Writing and evaluating mathematical formulas. Practical Applications: Building a basic calculator. Programs for area and perimeter calculations.	**6**	Apply arithmetic operators, operator precedence, and mathematical expressions in programming to build calculators and solve geometry-related programming tasks.	1, 2,3
II	**Logical and Boolean Operations** Boolean Algebra: Logical operators (&&, ||,!). Truth tables and their use programming. Conditional Expressions: Ternary operator (? :) and nested conditions. Applications: Writing decision-making programs. Programs for logical problem-solving (e.g., eligibility checks).	**6**	Apply Boolean operators, truth tables, and conditional expressions in C to create decision-making programs and solve logical problems effectively.	3
III	**Control Flow with Mathematical Logic** Loops and Iterations: For, while, and do-while loops. Mathematical Applications: Sum of natural numbers, factorial, and Fibonacci sequence. Checking for prime numbers. Nested Loops: Solving matrix and pattern problems.	**6**	Implement loops and iterations in C to solve mathematical problems, including sequences, prime checking, matrices, and pattern generation.	2, 3, 4

IV	**Number Systems and Bitwise Mathematics** Introduction to Number Systems: Binary, octal, hexadecimal, and their conversions. Bitwise Operations: AND, OR, XOR, NOT, left shift, right shift. Applications: Binary representation of numbers. Bitwise manipulation for optimization.	**6**	Understand number systems and implement bitwise operations in C for conversions, binary representation, and optimization of programming tasks.	2, 3, 4
V	**Arrays and Matrices** Mathematical Use of Arrays: Single-dimensional and multi-dimensional arrays. Summation, average, and searching in arrays. Matrices: Matrix addition, subtraction, and multiplication. Practical Applications: Creating programs for data storage and retrieval.	**6**	Utilize arrays and matrices in C for mathematical operations, including summation, searching, and matrix computations, to manage data effectively.	2, 3, 4

TEXT BOOKS:

T1: "Let Us C" by Yashavant Kanetkar

T2: "Discrete Mathematics and its Applications" by Kenneth H. Rosen

REFERENCE BOOKS:

R1: "The C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie

R2: "Mathematical Logic for Computer Science" by Mordechai Ben-Ari

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply arithmetic operators and mathematical expressions to build programs for real-world applications like calculators and geometry problems.	**1, 5**
2	Use Boolean algebra and conditional expressions to implement decision-making and problem-solving programs	**2, 3**
3	Implement loops and nested structures to solve mathematical problems, including sequences, prime checking, and matrix patterns.	**1, 3**
4	Convert between number systems and optimize programs using bitwise operators and binary manipulations.	**1, 4**
5	Use arrays and matrices for complex calculations like summation, searching, and matrix operations to handle and process data effectively.	**3, 5**

SEMESTER – III									
Course Title	**PREDICTIVE ANALYSIS**								
Course code	**24MCAO2101R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30L+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. Understand model building and evaluation techniques. 2. Use predictive analytics for business decision-making. 3. Gain hands-on experience with tools like IBM Watson and Python								
CO1	Understand and critically apply the concepts and methods of Predictive analytics.								
CO2	Understand and apply IBM SPSS Modeller in Data Mining, what kinds of data can be mined, what kinds of patterns can be mined.								
CO3	Applying and analysing how to use functions, deal with missing values, use advanced field operations, handle sequence data and improve efficiency.								
CO4	Get exposed to the testing of hypothesis and solving assumptions								
CO5	Understanding of IBM Watson Studio with Machine Learning Model.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Analytics Overview:** What is Predictive Analysis, how predictive model works, why predictive modelling, what are the models in Predictive Analysis. How Predictive analytics: Transforming data into future insights, Analytics trends: Past, Present & Future, Towards a Predictive enterprise.	**8**	Learners will understand how predictive models forecast future outcomes using historical data and apply analytics to solve business problems, while also exploring trends and transforming organizations into predictive enterprises.	1, 2
II	**Statistical Analysis:** Define Statistics and its types, Measures of Central Value; Mean, Median and Mode, Measures of Dispersion: Absolute and Relative measures of dispersion - Range, Quartile Deviation, Mean Deviation, Standard Deviation, Coefficient of Variation, Moments, Skewness, Kurtosis.	**8**	Upon completing the unit on statistical analysis, learners will be	1, 2

			able to calculate and interpret measures of central tendency and dispersion, understand data distribution through moments, skewness, and kurtosis, and apply these statistical techniques to analyze and make informed decisions from data.	
III	Data Mining:What is a Data Mining application? Strategy for data mining: CRISP-DM, Steps of Data Mining, Stages and tasks in CRISP-DM, Life Cycle of a Data Mining Project, Skills Needed for Data Mining. Identify nodes and streams, The framework of a Data – mining project, Brief the unit of analysis, Explain the type of dialog box.	**8**	On completing the Data Mining unit, learners will be able to apply the CRISP-DM methodology, manage data mining projects, and effectively analyze and interpret data using various tools and techniques.	1,2
IV	**Unit of Analysis:** Concepts of Unit of analysis (Distinct, Aggregate, SetToFlag), Integrate data, CLEM Expression, Role of Relationship between two fields, Identifying the modeling objective.	**8**	Upon completing this unit, learners will be able to identify units of analysis, integrate data, apply CLEM expressions, and define modeling objectives using field relationships.	1, 2, 3
V	**Predictive Analytics With IBM Watson Studio:**Understanding of IBM Cloud, IBM Watson Studio, Watson studio Components, Creating a	**8**	Learners will be able to use IBM Cloud and	1, 2, 3

	Machine Learning Model, Data preparation, Watson Machine learning, Data Refinery, Watson Studio Neural Network Modeler, IBM Watson Studio jobs, Use case with AutoAI.		Watson Studio tools to create machine learning models, prepare data, build neural network models, and apply AutoAI for real-world use cases.	
Practical				
Practical 1	Introduction to SPSS, Sorting File, Split File, Compute File, Recode File and Select Cases	**2**	Know the SPSS software which is used for statistical analysis, data management, and visualization in various fields like social sciences, health, and market research.	1
Practical 2	Chi- Square Test (Parametric and Non-Parametric Test)	**2**	The Chi-Square Test assesses associations between categorical variables, with the test for independence checking relationships and the goodness-of-fit test comparing observed vs. expected frequencies.	1, 2
Practical 3	Exploratory Factor Analysis	**2**	Exploratory Factor Analysis (EFA) identifies underlying	2, 3

			relationships among variables by grouping them into factors, helping to simplify complex data structures.	
Practical 4	Cluster Analysis	**2**	Identify patterns and segment data into distinct groups.	2
Practical 5	Logistic Regression	**2**	Logistic Regression is a statistical method used to model the relationship between a dependent binary variable and one or more independent variables, predicting the probability of an event occurring.	3
Practical 6	Discriminant Analysis	**4**	Discriminant Analysis is a statistical technique used to classify cases into predefined groups based on predictor variables, aiming to find the best combination of variables that differentiates the groups.	3
Practical 7	Confirmatory Factor Analysis	**4**	Confirmatory Factor Analysis (CFA) is a statistical	3

			method used to test whether a hypothesized factor structure fits the observed data, validating the relationship between measured variables and underlying factors.	
Practical 8	Time Series	**4**	Time Series analysis involves analyzing data points collected or recorded at specific time intervals to identify trends, patterns, and forecast future values.	3
Practical 9	MANOVA	**4**	MANOVA (Multivariate Analysis of Variance) is a statistical test used to examine the effect of independent variables on multiple dependent variables simultaneously, assessing whether group differences exist across several outcomes.	3, 4
Practical 10	Decision Tree Analysis	**4**	Decision Tree Analysis is a predictive modeling technique that maps out	4

			possible decisions, outcomes, and their probabilities, helping to classify or predict data based on input features through a tree-like structure.	

TEXT BOOKS:

T1 : Eric Siegel, Predictive Analytics: The Power to Predict, 1st Edition, Wiley, 2013.

T2 : Anasse Bari, Mohamed Chaouchi, and Tommy Jung, Predictive Analytics for Dummies, 1st Edition, Wiley, 2016.

REFERENCE BOOKS:

R1 : Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. 1st edition. O'Reilly Media, 2013.

R2 : Linda Miner, Gary Miner, and Dursun Delen, Practical Predictive Analytics and Decisioning Systems for Medicine, First Edition, Morgan Kaufmann, 2013

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand and critically apply the concepts and methods of Predictive analytics.	**1, 2, 3, 4, 5, 12**
2	Understand and apply IBM SPSS Modeller in Data Mining, what kinds of data can be mined, what kinds of patterns can be mined.	**1, 2, 3, 4, 5**
3	Applying and analysing how to use functions, deal with missing values, use advanced field operations, handle sequence data and improve efficiency.	**1, 2, 3, 5**
4	Get exposed to the testing of hypothesis and solving assumptions	**1, 2, 3, 5**
5	Understanding of IBM Watson Studio with Machine Learning Model.	**1, 2, 5, 7, 8**

SEMESTER – III									
Course Title	**COMPUTER NETWORK AND INFORMATION SECURITY**								
Course code	**24MCAO2102R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30L+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To gain a basic understanding of computer networking concepts and familiarise students with the fundamental taxonomy and terminology of computer networking. 2. To introduce students to advanced networking concepts in order to prepare them for entry-level Advanced courses in computer networking. 3. To enable the student to gain expertise in specific areas of networking, such as network design and maintenance.								
CO1	Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP. Apply channel allocation, framing, error and flow control techniques.								
CO2	Describe the functions of Network Layer. Explain the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism.								
CO3	Explain the functions offered by session and presentation layer and their Implementation.								
CO4	Explain the different protocols used at application layer i.e. HTTP, SNMP, SMTP, FTP, TELNET and VPN.								
CO5	Understand design issues in Network Security and to understand security threats, security services and mechanisms to counter.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introductory Concepts**: Goals and applications of networks, Categories of networks, Organization of the Internet, ISP, Network structure and architecture (layering principles, services, protocols and standards), The OSI reference model, TCP/IP protocol suite, Network devices and components. Physical Layer: Network topology design, Types of connections, Transmission media, Signal transmission and encoding, Network performance and transmission impairments, Switching techniques and multiplexing.	**6**	Understand the fundamentals of computer networks, including network types, Internet structure, ISP roles, and principles of network layering such as the OSI and TCP/IP models.	1, 2
II	**Link layer**: Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols). Medium Access Control and Local Area Networks: Channel allocation, Multiple access protocols, LAN standards, Link layer switches &	**6**	Understand the concept of framing in data communication, error detection methods, flow control mechanisms, communication channels in a network, multiple access protocols such as ALOHA,	1, 2

	bridges (learning bridge and spanning tree algorithms).		CSMA/CD, and CSMA/CA, LAN standards, ink layer switches and bridges in a network.	
III	Network Layer: Point-to-point networks, Logical addressing, Basic internetworking (IP, CIDR, ARP, RARP, DHCP, ICMP), Routing, forwarding and delivery, Static and dynamic routing, Routing algorithms and protocols, Congestion control algorithms, IPv6.	**6**	Understand the architecture and functioning of point-to-point networks, which involve direct connections between two devices.	1,2
IV	Transport Layer: Process-to-process delivery, Transport layer protocols (UDP and TCP), Multiplexing, Connection management, Flow control and retransmission, Window management, TCP Congestion control, Quality of service	**6**	Gain a solid foundation in transport layer concepts, enabling them to effectively design, implement, and manage transport protocols and network performance.	1, 2, 3
V	Application Layer: Domain Name System, World Wide Web and Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Remote login, Network management, Data compression, Cryptography – basic concepts.	**6**	Gain a comprehensive understanding of essential networking and data management concepts, equipping them with the knowledge to effectively work with various Internet protocols and technologies, ensuring secure and efficient communication.	1, 2, 3
Practical				
Practical 1	Familiarization with networking components and devices LAN adapter, Hub, Switches, Routers etc.	**2**	Become familiar with key networking components and devices such as LAN adapters, hubs, switches, and routers, understanding their functions and roles within a network.	1
Practical 2	Familiarization with Transmission media and tools: Co-axial cable, UTP cable, Crimping tool, Connectors etc.	**2**	Gain practical knowledge of transmission media and tools by identifying and understanding the uses of Co-axial cable, UTP cable, Crimping tool, and Connectors.	1, 2
Practical 3	Preparing the UTP cable for cross and direct connection using crimping tool.	**2**	Gain practical knowledge of transmission media and tools by identifying and understanding the uses of Co-axial cable, UTP cable,	2, 3

			Crimping tool, and Connectors.	
Practical 4	Introduction to various interior and exterior routing protocols.	**2**	Understand and describe the fundamental concepts and operational distinctions between various interior and exterior routing protocols such as OSPF, BGP, and EIGRP.	2
Practical 5	Installation and introduction of simulation tools packet tracer/GNS3.	**2**	Successfully install and navigate simulation tools such as Packet Tracer and GNS3, gaining proficiency in configuring virtual network environments.	3
Practical 6	Study of various LAN topologies and their creation using network devices, cables, and Computer.	**2**	Identify and construct various LAN topologies using network devices, cables, and computers to understand their structural differences and practical applications.	3
Practical 7	Configuration of TCP/IP protocols in Window/LINUX.	**2**	Configure TCP/IP settings on Windows and Linux systems to ensure proper network communication.	3
Practical 8	Configuration of TELNET protocols on router for remote access.	**2**	Configure TELNET on routers to enable remote network management and access.	3
Practical 9	Working with Sniffers for monitoring network communication using a) Ethereal b) Wire shark c) Snort d) tcp dump.	**2**	Utilize network sniffers like Ethereal, Wireshark, Snort, and tcpdump to monitor and analyze network traffic.	3, 4
Practical 10	Implementation and Performance evaluation of various cryptographic algorithms in C/C++ a) DES b) RSA	**2**	Implement and evaluate the performance of cryptographic algorithms such as DES and RSA using C/C++ to understand their efficiency and security implications.	4
Practical 11	Using IP TABLES on Linux and setting the filtering rules.	**2**	Implement and manage IP Tables on Linux systems to control network traffic through effective filtering rules.	3
Practical 12	Using open SSL for web server - browser communication.	**2**	Configure and utilize OpenSSL to secure web server-browser communications through SSL/TLS protocols.	3

Practical 13	Configuring S/MIME for e-mail communication.	**2**	Configure S/MIME to secure email communications by ensuring confidentiality and integrity of messages.	3
Practical 14	Understanding the buffer overflow and format string attacks.	**2**	Understand and identify vulnerabilities such as buffer overflows and format string attacks in software applications.	2, 4
Practical 15	Using NMAP for ports monitoring and Secure Socket programming	**2**	Use NMAP for monitoring network ports and implement secure socket programming to enhance network security.	3, 4

TEXT BOOKS:

TI: Forouzan B. Data Communication and Networking. 5th edition. New York: McGraw Hill; 2012.

T2: Tanenbaum A. Computer Networks. 5th edition. Upper Saddle River, NJ: Prentice Hall; 2010.

REFERENCE BOOKS:

R1: Stallings W. Data and Computer Communication. 10th edition. Boston: Pearson; 2013.

R2: Kurose J, Ross K. Computer Networking: A Top-Down Approach. 7th edition. Boston: Pearson; 2017.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP. Apply channel allocation, framing, error and flow control techniques.	**1, 2, 3, 4, 5, 12**
2	Describe the functions of Network Layer. Explain the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism.	**1, 2, 3, 4, 5**
3	Explain the functions offered by session and presentation layer and their Implementation.	**1, 2, 3, 5**
4	Explain the different protocols used at application layer i.e. HTTP, SNMP, SMTP, FTP, TELNET and VPN.	**1, 2, 3, 5**
5	Understand design issues in Network Security and to understand security threats, security services and mechanisms to counter.	**1, 2, 5, 7, 8**

SEMESTER – III									
Course Title	**USABILITY DESIGN OF SOFTWARE APPLICATIONS**								
Course code	**24MCAO2103R**	**Total credits: 2**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T	**2**	**0**	**0**	**0**	**0**	**0**	**2**
Pre-requisite	**NIL**	**Co-requisite**	**NIL**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects. 2. Topics include process models, software requirements, software design, software testing, Risk Management, Quality assurance etc. 3. Knowledge to acquaint themselves with the special challenges of starting new ventures.								
CO1	Understand various software characteristics and analyse software Development Models.								
CO2	Understanding project planning, and Risk Management.								
CO3	To sensitize the students to the fundamentals of User Centred Design and User Experience their relevance and contribution to businesses								
CO4	Analysis of different software testing strategies.								
CO5	Apply estimation techniques, schedule project activities, Quality Assurance and compute pricing.								

Unit-No.	Content	Contact Hour	Learning Outcome	KL
I	The Product and The Process; Evolving Role of Software, Software Characteristics; The Process; Software Engineering: A Layered Technology, The Software Process, Software Process Models	**9**	Gain a solid foundation in understanding the nature of software and the processes involved in its development. This knowledge will enable them to effectively participate in and manage software development projects, ensuring the delivery of high-quality software products.	1, 2
II	Software Project Planning; Project Planning Objectives, Software Scope, Project Estimation Technique; Empirical estimation techniques, Heuristic estimation techniques. Risk Management; Reactive Vs. Proactive Risk Strategies, Software Risk, Risk Identification, Risk Projection,	**9**	Gain a comprehensive understanding of software project planning and risk management	1, 2
III	Project Scheduling and Tracking; Basic Concepts, defining a Task set for the Software Project, Selecting	**9**	Gain essential skills in project scheduling, tracking, and quality assurance in software	1, 2, 3

	Software Engineering Tasks, Defining a Task Network, Scheduling, The Project Plan; Software Quality Assurance; Quality Concepts, Software Reviews, Software Reliability, The ISO 9000 Quality Standards;		engineering	
IV	Analysis and Design; Basic Concepts and Principles: Analysis Modelling, The Data Dictionary; Software Reuse, Economics of Software Reuse; Reengineering: Software Reengineering, Reverse Engineering, Forward Engineering, The Economics of Reengineering	**9**	Gain a comprehensive understanding of analysis and design principles in software engineering, as well as the strategic aspects of software reuse and reengineering	1, 2
V	Introduction and Aspects of User Centred Design, Basics of User Centered Design, Product Appreciation Assignment – Evaluating the product from user centered design aspects such as Functionality, ease of use, ergonomics, and aesthetics. Software Testing; Different Software Testing Methods: Software Testing Fundamentals, White Box Testing, Black Box Testing.	**9**	Gain essential skills in user-centered design principles and software testing methodologies	1, 2, 3

TEXT BOOKS:

T1: Fundamentals of Software Engineering, Rajiv Mall, Prentice-Hall India, Fourth Edition, 2016

T2: Interaction Design: Beyond Human-Computer Interaction, Jennifer Preece, Helen Sharp, Yvonne Rogers, Wiley publications, 4th Edition, 2015

REFERENCE BOOKS:

R1: Software Engineering A Practitioner's Approach, R. S.Pressman, Tata McGraw-Hill, Seventh Edition, 2014

R2: About Face The Essentials of Interaction Design, Alan Cooper and Robert Riemann, Wiley Publications, 4th Edition, 2014.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand various software characteristics and analyse software Development Models.	**1, 2, 5, 12**
2	Understanding project planning, and Risk Management.	**2, 3, 7, 11**
3	To sensitize the students to the fundamentals of User Centred Design and User Experience their relevance and contribution to businesses	**1, 3, 9, 12**
4	Analysis of different software testing strategies.	**1, 3, 5, 12**
5	Apply estimation techniques, schedule project activities, Quality Assurance and compute pricing.	**2, 3, 4, 11, 12**

SEMESTER – III									
Course Title	**OBJECT ORIENTED PROGRAMMING USING JAVA**								
Course code	**24MCAO2104R**	**Total credits: 3**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30T+30P	**2**	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Basic Programming**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To teach how to decompose the problem into a set of objects. 2. To teach how Objects interact with each other to solve the problem. 3. To teach how to create new type of objects to model elements from the problem space.								
CO1	Understanding fundamental principles of OO programming, OO analysis, design, and development.								
CO2	Apply inheritance and polymorphism concepts of OOPs on computing problem.								
CO3	Design applications for a range of problems using file and exception handling.								
CO4	Implementation of object-oriented based projects.								
CO5	Demonstrate the use of various OOPs concepts with the help of programs.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Principles of Objective Oriented Programming**: Object Oriented Programming Paradigm, Basic Concepts of Object-Oriented Programming, Benefits of Object-Oriented Programming, Object Oriented Languages, Applications of Object-Oriented Programming. Token Expressions & Control Structures: Tokens, Keywords, Identifiers and Constants, Data Types, Type Compatibility, Variables, Operators, Implicit Conversions, Operator Precedence, Control Structures.	**6**	Gain a solid foundation in object-oriented programming principles and fundamental concepts of token expressions and control structures	1, 2
II	**Array and String**: Single Array & Multidimensional Array, Library Classes-String, String Buffer & Wrapper Class, Command line arguments and Various String Operations. **Functions**: Function prototyping, function components, passing parameters, call by reference, return by reference, inline functions, default arguments, overloaded function. Pointers: Array of objects, pointers to objects, this pointer, dynamic allocation operators, dynamic objects.	**6**	Gain a comprehensive understanding of essential programming concepts related to arrays, strings, functions, and pointers	1, 2
III	**Object and Classes**: Making sense of core object concepts (Encapsulation,	**6**	Gain a solid foundation in object-oriented	1, 2

	Abstraction, Polymorphism, Classes, Messages Association, Interfaces), Implementation of class and Objects. Structures and classes. Classes objects and memory static class data. **Constructors & Destructors**: Constructors, Parameterized Constructors, Copy Constructors, Dynamic Constructors, Destructors.		programming principles, including encapsulation, abstraction, polymorphism, classes, constructors, and destructors	
IV	**Inheritance**: Concept of inheritance. Derived class and based class. Derived class constructors, member function, inheritance in the English distance class, class hierarchies, inheritance and graphics shapes, public and private inheritance, aggregation: Classes within classes, inheritance and program development. **Polymorphism**: Definition of Polymorphism, Compile Time Polymorphism, Runtime Polymorphism, Defining Operator Overloading, Overloading Operators, Rules for Overloading Operators, Type Conversions, Virtual Function	**6**	Gain a comprehensive understanding of inheritance and polymorphism in object-oriented programming	1, 2
V	**Exception handling**: Try, throw, and catch, exceptions and derived classes, function exception declaration, unexpected exceptions, exception when handling exceptions, resource capture and release. **Streams and Files**: Streams classes, Stream Errors, Disk File I/O with streams, file pointers, error handling in file I/O with member function, overloading the extraction and insertion operators, memory as a stream object, command line arguments, and printer output.	**6**	Gain proficiency in exception handling techniques to manage errors and exceptional conditions effectively in software development	1, 2, 3

Practical

Practical	Content	Contact Hour	Learning Outcome	KL
Practical 1	Program to show swap of two numbers by using third variable, and without using a third variable	**2**	Perform basic variable manipulation programmatically, handle user inputs and outputs, and write well-structured and documented code.	5, 6
Practical 2	Program to show leap year or not.	**2**	Gain proficiency in using conditional statements to	5, 6

			solve problems, handling user inputs and outputs, and writing well-structured and documented code	
Practical 3	Program to use switch statement.	**2**	Gain proficiency in using switch statements to implement decision-making logic	5, 6
Practical 4	Program to display first 10 natural number and their sum.	**2**	Gain proficiency in using loops, arithmetic operations, and output handling in programming	5, 6
Practical 5	Program to print star pattern of the following: * * * * * * * * * * * * * * *	**2**	Gain proficiency in using nested loops, controlling pattern formation	5, 6
Practical 6	Program to find Fibonacci series up to a range.	**2**	Gain proficiency in using loops, handling numerical sequences, managing user inputs and outputs	5, 6
Practical 7	Program to check a given number is Armstrong or not.	**2**	Gain proficiency in using loops for iterative operations, handling mathematical computations	5, 6
Practical 8	Program to find whether the given number is prime or not.	**2**	Gain proficiency in using loops for iterative operations, handling numerical operations	5,6
Practical 9	Program to display 1+3+5+. . .+n.	**2**	Gain proficiency in using loops for iterative operations, handling numerical operations involving odd numbers	5, 6
Practical 10	Program to show sum of 10 elements of array and show the average. find the maximum and minimum number in an array.	**2**	Gain proficiency in working with arrays, using loops for iterative operations, performing basic mathematical computations	5, 6
Practical 11	Program to demonstrate the use of constructors.	**2**	Gain proficiency in understanding the role and usage of constructors in object-oriented programming	5, 6
Practical 12	Program to demonstrate inheritance.	**2**	Develop proficiency in using destructors for resource management and understanding inheritance concepts	5, 6

Practical 13	Program to demonstrate function overloading.	**2**	Develop proficiency in using function overloading effectively in programming, understanding its benefits in code organization and polymorphism	5, 6
Practical 14	Program to demonstrate operator overloading.	**2**	Develop proficiency in using operator overloading effectively in programming, understanding its benefits in code customization and usability	5, 6
Practical 15	Program to demonstrate virtual functions.	**2**	Develop proficiency in using virtual functions effectively in programming	5, 6

Text Book:

T1: Silberschatz A, Galvin P, Gagne G. Operating System Concepts Essentials. 9th edition. Singapore: Wiley Asia Student Edition; 2018.

T2: Stallings W. Operating Systems: Internals and Design Principles. 5th edition. New Delhi: Prentice Hall of India; 2004.

Reference Book:

R1: Crowley C. Operating System: A Design-Oriented Approach. 1st edition. Boston: Irwin Publishing; 1996.

R2: Nutt GJ. Operating Systems: A Modern Perspective. 2nd edition. Boston: Addison-Wesley; 2000.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understanding fundamental principles of OO programming, OO analysis, design and development.	**1, 2, 3, 12**
2	Apply inheritance and polymorphism concepts of OOPs on computing problem.	**1, 3, 4, 12**
3	Design applications for a range of problems using file and exception handling.	**1, 3, 4, 12**
4	Implementation of object-oriented based projects.	**3, 4, 5, 11, 12**
5	Demonstrate the use of various OOPs concepts with the help of programs.	**1, 3, 4, 5, 12**

SEMESTER – III									
Course Title	**PROGRAMME ELECTIVE -I – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**								
Course code	**24MCAO2105R**	**Total credits: 3 Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Basic Programming Skills and Mathematics**	**Co-requisite**	**NIL**						
Programme	**Master of Computer Application on Artificial Intelligence and Deep Learning**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To acquire a comprehensive understanding of AI and machine learning concepts, methodologies, and practical implementation skills using industry-relevant tools. 2. To master data preprocessing, analysis, feature engineering, and apply principles of model evaluation and optimization for enhanced performance. 3. To develop an understanding of the ethical dimensions of AI, focusing on bias, fairness, and transparency, and learn strategies for responsible societal impact.								
CO1	Gain a comprehensive grasp of foundational AI and machine learning concepts, encompassing algorithms and methodologies								
CO2	Develop proficient hands-on skills in implementing AI and machine learning models, utilizing industry-relevant programming languages and frameworks								
CO3	Acquire expertise in preprocessing and analyzing data, mastering techniques for feature selection and engineering to enhance model performance								
CO4	Demonstrate a sound understanding of model evaluation principles and optimization techniques, ensuring the ability to enhance model efficiency and effectiveness								
CO5	Explore the ethical dimensions of AI, examining issues related to bias, fairness, and transparency, and develop strategies for addressing societal implications responsibly.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to AI and Machine Learning**: Define AI, Machine Learning, and Deep Learning, Types of Machine Learning: Supervised, Unsupervised, Reinforcement Learning, History and applications of AI	**6**	Gain a foundational understanding of AI and Machine Learning concepts, types, historical development, and practical applications. They will be equipped to comprehend the fundamental principles and significance of AI in modern technology and society, preparing them for further studies or careers in AI and related fields.	1, 2
II	**Supervised Learning Fundamentals**: Concepts of linear models, cost functions, gradient descent, Implementation for prediction tasks (housing prices, etc.), Logistic Regression: Classification concepts, sigmoid function, decision boundaries, Implementation for	**6**	Develop a solid understanding of supervised learning fundamentals, including linear models, logistic regression, decision trees, random forests, and their practical applications in prediction and classification tasks	1, 2

	binary classification problems (email spam, medical diagnosis) Decision Trees & Random Forests: Building and interpreting decision trees, handling overfitting, Ensemble methods and the power of random forests			
III	**Deep Learning**: Neural Networks Neurons, activation functions, backpropagation, Building simple neural networks (Multilayer Perceptron), Convolutional Neural Networks (CNNs): Image processing, kernels, pooling layers, CNN architectures for image classification tasks (MNIST, CIFAR-10)	**6**	Develop a comprehensive understanding of deep learning concepts, neural network architectures, and practical skills in building and training models for image processing and classification tasks using Convolutional Neural Networks (CNNs)	1, 2
IV	**Unsupervised Learning and Advanced Topics**:Clustering - K-Means, hierarchical clustering, applications **Dimensionality Reduction**: Principal Component Analysis (PCA), feature visualization **Natural Language Processing** (NLP): Text preprocessing, word embeddings Introduction to Recurrent Neural Networks (RNNs)	**6**	Gain a comprehensive understanding of advanced topics in unsupervised learning, dimensionality reduction, natural language processing, and recurrent neural networks	1, 2, 3
V	**Applying AI**: Best Practices & Ethics: Bias & Fairness in AI, Transparency & Explainability, Fairness and Bias: **Types of bias in AI Transparency and Explainability**: The need for explainable AI (XAI), Methods for understanding model decisions, Balancing transparency with trade secrets and intellectual property **Privacy and Data Security**: Data protection principles in AI contexts, Strategies for secure data collection, storage, and use	**6**	Develop a comprehensive understanding of best practices and ethical considerations when applying AI	1, 2
Practical				
Practical 1	Write a Python script to demonstrate basic data types and their operations to refresh Python programming skills.	**2**	Understand and demonstrate the use of Python's basic data types and operations.	2
Practical 2	Implement a simple linear regression model from scratch using Python to predict housing prices based on dataset features.	**2**	Develop a simple linear regression model from scratch to predict housing prices and understand the underlying	3

			mathematics.	
Practical 3	Develop a logistic regression model to classify emails as spam or not spam.	**2**	Develop and evaluate a logistic regression model to classify emails as spam or not spam, demonstrating understanding of classification thresholds and decision boundaries.	3
Practical 4	Build and visualize a decision tree using the scikit-learn library to classify patients based on medical diagnosis data.	**2**	Construct and visualize a decision tree to accurately classify medical diagnosis data, demonstrating understanding of tree-based learning algorithms.	3
Practical 5	Implement a Random Forest model to predict credit risk from financial datasets.	**2**	Develop a Random Forest model to predict credit risk from financial datasets and understand the concept of ensemble learning.	3
Practical 6	Create a Python program to simulate the gradient descent algorithm showing how it optimizes the cost function.	**2**	Create and demonstrate a Python simulation of the gradient descent algorithm to optimize a cost function, illustrating the process of parameter optimization.	3
Practical 7	Write a Python program to build a basic neural network using numpy to understand forward and backward propagation.	**2**	Develop a basic neural network using numpy to understand the principles of forward and backward propagation in neural computation.	5
Practical 8	Use TensorFlow and Keras to build a convolutional neural network to classify images from the CIFAR-10 dataset.	**2**	Build and train a convolutional neural network using TensorFlow and Keras for image classification, demonstrating the application of CNNs in deep learning.	5
Practical 9	Implement K-means clustering to segment customers based on shopping data.	**2**	Implement and evaluate the effectiveness of K-means clustering for customer segmentation, demonstrating understanding of clustering techniques in unsupervised learning.	5
Practical 10	Apply Principal Component Analysis using scikit-learn to reduce the dimensions of a dataset while preserving as much variance as possible.	**2**	Apply Principal Component Analysis to effectively reduce the dimensions of a dataset while preserving maximum variance, demonstrating the concept of feature reduction.	5

Practical 11	Write a script to preprocess text data, including tokenization, removing stopwords, and generating word embeddings.	**2**	Demonstrate proficiency in preprocessing text data by performing tokenization, removing stopwords, and generating word embeddings, understanding their impact on natural language processing tasks.	5
Practical 12	Develop a Recurrent Neural Network to predict the next word in a sentence using TensorFlow.	**2**	Develop a Recurrent Neural Network using TensorFlow to predict the next word in a sentence, demonstrating understanding of sequential data and RNN architecture.	5
Practical 13	Create a program to analyze and report potential biases in a dataset used for training an AI model.	**2**	Critically analyze and identify potential biases in AI training datasets, demonstrating understanding of the implications of biased data on AI fairness and performance.	5
Practical 14	Implement a method to interpret the decisions made by a machine learning model, such as using SHAP or LIME.	**2**	Implement and use methods like SHAP or LIME to interpret and explain the decisions made by machine learning models, demonstrating the ability to apply techniques for model transparency.	4, 5
Practical 15	Write a Python script that demonstrates encryption and decryption methods to protect data used in AI applications.	**2**	Understand and demonstrate the application of encryption and decryption techniques to protect data integrity and privacy in AI applications.	4, 5

TEXT BOOKS:

T1: Russell S, Norvig P. Artificial Intelligence: A Modern Approach. 4th edition. Upper Saddle River, NJ: Pearson; 2020.

T2: Goodfellow I, Bengio Y, Courville A. Deep Learning. 1st edition. Cambridge: MIT Press; 2016.

T3: Barocas S, Hardt M, Narayanan A. Fairness and Machine Learning. 1st edition; 2023.

REFERENCE BOOKS:

R1: Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2nd edition. Sebastopol: O'Reilly Media; 2019.

R2: Müller AC, Guido S. Introduction to Machine Learning with Python: A Guide for Data Scientists. 2nd edition. Sebastopol: O'Reilly Media; 2023.

R3: Bishop CM. Pattern Recognition and Machine Learning. 1st edition. New York: Springer; 2006.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Gain a comprehensive grasp of foundational AI and machine learning concepts, encompassing algorithms and methodologies	**1, 2, 12**
2	Develop proficient hands-on skills in implementing AI and machine learning models, utilizing industry-relevant programming languages and frameworks	**1, 3, 5, 12**
3	Acquire expertise in preprocessing and analyzing data, mastering techniques for feature selection and engineering to enhance model performance	**1, 3, 4, 12**
4	Demonstrate a sound understanding of model evaluation principles and optimization techniques, ensuring the ability to enhance model efficiency and effectiveness	**1, 3, 4, 12**
5	Explore the ethical dimensions of AI, examining issues related to bias, fairness, and transparency, and develop strategies for addressing societal implications responsibly.	**7, 8, 9, 12**

SEMESTER – III									
Course Title	**PROGRAMME ELECTIVE -I - BUSINESS INTELLIGENCE**								
Course code	**24MCAO2105R**	**Total credits: 3 Total hours: 45T**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			3	**0**	**0**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**NIL**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To learn how managers utilise business analytics to create, address, and assist management decision-making for company challenges. 2. To become accustomed to the procedures required to create, present, and analyse business data. 3. Get knowledge on how to use Excel and its add-ons to solve business challenges								
CO1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.								
CO2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.								
CO3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.								
CO4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods								
CO5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Overview of Business Analytics**: Introduction to Analytics, The Challenge of Decision Making, What Is Business Intelligence, The Business Intelligence Value Proposition, The Combination of Business and Technology	**9**	Define and understand the fundamentals of business analytics, the decision-making challenges, the concept and value of business intelligence, and the synergistic relationship between business and technology.	1, 2
II	**Introduction to Data Visualization**: Visualization/ Data Issues, Organization/sources of data, Importance of data quality, Dealing with missing or incomplete data, Data Classification.	**9**	Master the principles of data visualization, including data organization, the significance of data quality, strategies for handling missing data, and data classification techniques.	2, 3

III	**Descriptive Analytics**: Data Visualization and Analytics- Charts (Bars-Pie-Line-Scatter-Map-Bubble-Box & Whisker-Tree map - Heat map-Circle and Area) - Worksheet, Dashboard and Story Board creation	**9**	Create and interpret diverse charts and visualizations such as bar, pie, line, scatter, map, bubble, box & whisker, tree map, heat map, and area charts, and develop comprehensive worksheets, dashboards, and storyboards.	3, 6
IV	**Predictive Analytics**: Linear Regression, Multi-linear Regression and Time Series Forecasting. Prescriptive Analytics: Linear optimization, Integer optimization, Non-linear programming, Optimization of Network models and Monte Carlo Simulation	**9**	Apply linear and multi-linear regression models and time series forecasting for predictive analytics, and utilize optimization techniques	3, 4
V	**Applications**: Practical demonstration using R or Python or SPSS by industry experts on Emerging trends in business analytics and intelligence	**9**	Demonstrate proficiency in using software tools like R, Python, or SPSS to implement advanced analytics techniques	3, 5, 6

TEXT BOOKS:

T1: Sharda R, Delen D, Turban E, Aronson J, Liang TP. Business Intelligence and Analytics: Systems for Decision Support. 10th edition. Boston: Pearson Education; 2014.

T2: Powell SG, Barker KR. Management Science: The Art of Modeling with Spreadsheets (W/CD). 4th edition. Hoboken, NJ: John Wiley & Sons; 2014.

REFERENCE BOOKS:

R1: Linoff GS, Berry MJ. Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management. 3rd edition. Hoboken, NJ: John Wiley & Sons; 2011.

R2: Frank B, Green B, Harris T, Van De Vanter K. Business Intelligence Strategy: A Practical Guide for Achieving BI Excellence. New York: MC Press; 2010.

R3: Hair JF, Black WC, Babin BJ, Anderson RE, Tatham RL. Multivariate Data Analysis. 7th edition. Boston: Pearson Education; 2009.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Understand the basic concepts of Business Analytics, including the roles of Business Intelligence and the value proposition of integrating business with technology to enhance decision-making.	**1, 2, 12**
2	Demonstrate the ability to organize and source data, address data quality issues, and classify data effectively to prepare for advanced visualization techniques.	**1, 3, 5, 12**
3	Create and interpret a variety of data visualizations (such as bar charts, pie charts, scatter plots, heat maps) and utilize these visualizations in dashboards and storyboards to convey meaningful analytics insights.	**1, 3, 5, 12**
4	Apply predictive analytics techniques such as linear and multi-linear regression and time series forecasting, alongside utilizing prescriptive analytics methods	**1, 3, 4, 12**
5	Synthesize knowledge gained from industry experts on emerging trends in business analytics and intelligence through practical demonstrations.	**1, 6, 12**

SEMESTER – III									
Course Title	**PG RESEARCH PROJECT I**								
Course code	**24MCAO2107R**	**Total credits: 8**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**8**	**24**	**0**	**8**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To have a basic knowledge and understanding of surveys and experiments and its clinical implications in clinical practice. 2. Investigate a particular subject or phenomenon in detail to gain a deeper understanding of it. 3. Summarize and synthesize existing research related to the topic to identify gaps, trends, and key findings. 4. Formulate and test a specific hypothesis or set of hypotheses related to the research question.								
CO1	Enable Students a thorough understanding of how survey /experiments can provide useful causal inferences.								
CO2	Knowledge of how to design and analyze simple and complex experiments/ surveys								
CO3	Ability to evaluate experimental research / surveys and apply these methods in their own research.								
CO4	Enable students the basic knowledge and understanding in undertaking surveys and experiments into their clinical practice								
CO5	Enables the students to develop new skills and strategies in designing their survey/Experiments which can be implemented in patient care.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Surveys as Research Methodology**:What is survey research Survey strengths Survey weakness	**6**	Gain a foundational understanding of survey research methodology, its applications, strengths, and weaknesses. They will be equipped to design effective surveys, analyze survey data, and critically evaluate the reliability and validity of survey findings in research contexts.	1, 2
II	**Survey Process**:Survey design, sample selection, sample size, degree of precision, statistical power, effect size as a determinant of power, survey instrument development, Types of Survey	**6**	Develop a comprehensive understanding of the survey process, from design and sampling to instrument development and types of surveys	1, 2
III	**Qualitative Methods**:Unstructured & Semi-structured Interviewing Coding Responses to Open-Ended Questions	**6**	Develop essential skills in qualitative research methods, particularly in conducting unstructured and semi-structured interviews and coding responses from open-ended questions	1, 2, 3
IV	Survey Data Processing and Basic	**6**	By effectively processing and	1, 2,

	Data Analysis		analyzing survey data, researchers can uncover insights, trends, and correlations that inform decision-making, policy formulation, or further research directions.	3
V	**Advanced Survey Techniques and Applications** Survey Scaling Methods, Survey Data Collection Modes, Ethical Considerations in Survey Research, Applications of Survey Research,Future Trends in Survey Research	**6**	Gain a comprehensive understanding of survey research methods, including their design, strengths, weaknesses, and applications in various fields.	4, 5, 6

TEXT BOOKS:

T1: McGuire, W. G. (1997). Creative hypothesis generating in psychology: Some useful heuristics. Annual Review of Psychology, 48, 1-30.

T2: Beatty, P., & Hermann, D. (2002). To answer or not to answer: Decision processes related to survey item nonresponse. In D. A. Dillman, J. L. Eltinge, R. M. Groves, & R. J. A. Little (Eds.). (2002). Survey nonresponse (pp. 71-86). New York: Wiley3.

T3: Fink, A. (2019). Conducting research literature reviews: From the internet to paper. Sage publications.

T4: Cooper, H. (1998). Cooper, Harris, Synthesizing Research: A Guide for Literature Reviews, Thousand Oaks, CA: Sage, 1998.

T5: Hart, C. (2018). Doing a literature review: Releasing the research imagination.

OTHER LEARNING RESOURCES:

R1: Frey, J. H., & Oishi, S. M. (1995). How to conduct interviews by telephone and in person. Thousand Oaks, CA: Sage

R2: Fowler, F. J. (1995). Improving survey questions: Design and evaluation. Thousand Oaks, CA: Sage.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Enable Students a thorough understanding of how survey /experiments can provide useful causal inferences.	**1, 2, 12**
2	Knowledge of how to design and analyse simple and complex experiments/ surveys	**1, 3, 4, 12**
3	Ability to evaluate experimental research / surveys and apply these methods in their own research	**3, 4, 12**
4	Enable students the basic knowledge and understanding in undertaking surveys and experiments into their clinical practice	**1, 3, 12**
5	Enables the students to develop new skills and strategies in designing their survey/Experiments which can be implemented in patient care.	**3, 4, 12**

SEMESTER – III									
Course Title	**PDP III - CORPORATE PROFICIENCY**								
Course code	**24MOSY2101R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
		Total hours: 30P	**0**	**0**	**2**	**0**	**0**	**0**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**NIL**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. Develop Proficiency in Writing 2. Cultivate Leadership and Communication Skills 3. Master Interview and Conflict Management Techniques								
CO1	Proficiency in writing concise and structured paragraphs.								
CO2	Understanding of different leadership styles and the distinction between managers and leaders.								
CO3	Mastery of email etiquette and effective communication strategies.								
CO4	Preparation and execution of successful job interviews, including non-verbal cues and appropriate attire.								
CO5	Competence in identifying and implementing conflict management strategies in professional settings.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Writing Skills** i. Writing short paragraphs (topic sentence, supporting sentences and concluding sentence on a given topic) ii. Writing business/professional letters/ Letters of application for employment iii. Technical writing (Report Writing, Proposal drafting)	**12**	Develop proficiency in crafting structured paragraphs, formal business letters, and technical documents such as reports and proposals.	3
II	**Self-Management and Leadership Skills**: i. Concepts of Leadership Styles ii. Manager VS Leader iii. How to be an Effective Leader	**12**	Understand diverse leadership styles, differentiate between managers and leaders, and develop effective leadership strategies.	2, 3
III	**Email Etiquette**: i. Different parts of emails and usage i. Magic Words & Wonder Phrases, iii. Effective Email Communication, iv. Responding effectively to a client's expectations	**12**	Master the components of effective email communication, including structure, persuasive language, and responsiveness to client expectations.	2, 3
IV	**Interview Skills**:	**12**	Prepare for interviews by	2, 3

	i. Preparation before the interview ii. Non-verbal cues for interview skills iii. Dress code for interview iv. Interview dos and don'ts		mastering content, non-verbal cues, dress code, and understanding key interview conduct.	
V	**Conflict Management**: i. Introduction to Conflict Management ii. Conflict Management Strategies	**12**	Gain insights into conflict management principles and strategies to handle workplace conflicts effectively.	3

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Proficiency in writing concise and structured paragraphs.	**1, 5., 9, 12**
2	Understanding of different leadership styles and the distinction between managers and leaders.	**2, 6, 9**
3	Mastery of email etiquette and effective communication strategies.	**5, 9**
4	Preparation and execution of successful job interviews, including non-verbal cues and appropriate attire.	**5, 6, 9, 11**
5	Competence in identifying and implementing conflict management strategies in professional settings.	**2, 3, 7, 9, 12**

SEMESTER – III									
Course Title	**Field Visit**								
Course code	**24MCAO2108R**	**Total credits: 1**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**0**	**0**	**8**	**1**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To expose students to real-world applications of theoretical knowledge. 2. To enhance observational, analytical, and research skills through fieldwork. 3. To cultivate an appreciation for the complexities and challenges of professional practice in the field.								
CO1	Apply theoretical concepts to real-world situations.								
CO2	Conduct field-based research and gather data effectively.								
CO3	Analyze and interpret field data to draw meaningful conclusions.								
CO4	Communicate findings clearly and effectively, both orally and in writing.								
CO5	Demonstrate professional behaviour and teamwork skills in field settings.								
Course Contents	The organization will provide the students two field visit for this program. The students will be taken to a specific site for learning purposes. The students will visit a relevant site and then they will be guided and give a tour of the sites for observations. Followed by an interaction with professionals and practitioners regarding the subject or purpose of the visit. The students will have to maintain a journal to document the observations and reflections, and discussion on experiences and initial findings.								

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Apply theoretical concepts to real-world situations.	**1, 3, 10**
2	Conduct field-based research and gather data effectively.	**4, 5**
3	Analyze and interpret field data to draw meaningful conclusions.	**2, 4, 10**
4	Communicate findings clearly and effectively, both orally and in writing.	**8, 6**
5	Demonstrate professional behaviour and teamwork skills in field settings.	**6, 7**

SEMESTER – III									
Course Title	**Summer Internship**								
Course code	**24MCAO2106R**	**Total credits: 4**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**0**	**16**	**0**	**8**	**4**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ III semester of the second year of the programme**								
Course Objectives	1. To apply theoretical knowledge to real-world projects, gaining practical skills. 2. To enhance technical, analytical, and communication abilities in a professional setting. 3. To build connections with mentors and industry professionals for future career opportunities.								
CO1	Gain practical experience by applying academic knowledge to actual industry projects.								
CO2	Develop and enhance technical skills in relevant tools and methodologies.								
CO3	Strengthen critical thinking and problem-solving abilities by tackling real challenges.								
CO4	Improve teamwork, communication, and collaboration in a professional environment.								
CO5	Explore different career paths and gain industry insights through mentorship and networking opportunities.								
Course Contents	The course content for a summer internship includes an introduction to industry practices and organizational structure, followed by project assignment with clear goals and expectations. Interns will gain hands-on experience with relevant tools and technologies, including software and platforms commonly used in the field. The program also focuses on data collection, analysis, and problem-solving, where interns will engage in real-world tasks, collaborate with teams, and improve their technical, analytical, and communication skills, all while gaining valuable insights into potential career paths through mentorship and networking.								

SEMESTER – IV									
Course Title	**ARTIFICIAL INTELLIGENCE AND DEEP LEARNING**								
Course code	**24MCAM2201R**	**Total credits: 3** **Total hours: 30L+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Fall/ IV semester of the second year of the programme**								
Course Objectives	1. Grasp core concepts, algorithms, and techniques in AI and deep learning. 2. Gain practical experience in building machine learning and deep learning models. 3. Apply AI and DL methods to tackle challenges in areas like computer vision and NLP.								
CO1	Remember and understand relevance of AI in today's world.								
CO2	Gain an idea about Machine Learning application in real world problem.								
CO3	Develop an understanding deep learning concept in AI.								
CO4	Apply and Analyze data by applying various machine learning algorithms.								
CO5	Enhance problem-solving abilities with IBM Cloud service.								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Introduction to Artificial Intelligence:** Overview of Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL), Natural Language Processing (NLP), Computer Vision. Describe the types of AI, Explain what the main focus of AI is, List of practical applications of AI, Describe the eras of computing, Explain the difference between deterministic and probabilistic systems, Road Map of AI.	**8**	Upon completion, learners will understand AI, ML, DL, NLP, and Computer Vision, differentiate between system types, explore AI applications, and outline the AI development roadmap.	1, 2
II	**Introduction to Machine Learning:** Introduction to Machine Learning, Supervised Learning & Unsupervised Learning. Supervised Learning Algorithms: Linear Regression, Logistic Regression, Suppor Vector Machine, Decision	**8**	Upon completing the Machine Learning course, learners	1, 2

	Tree,Random Forest, KNN. Unsupervised Learning Algorithms: K-Means Clustering, Hierarchical Clustering.		will be able to understand and apply supervised learning algorithms like Linear Regression, Logistic Regression, and Decision Trees, as well as unsupervised learning techniques such as K-Means and Hierarchical Clustering, to solve real-world problems.	
III	**Introduction to Deep Learning:** Introduction to Deep Learning: Neural Networks, Layers, and Activation Functions, Back Propagation, Optimizers, Deep Learning Algorithm: ANN, CNN, RNN, LSTM withTensorflow and Keras, Introduction to Autoencoders, Generative Adversarial Networks (GANs), Restricted Boltzmann Machines (RBMs) and Applications.	**8**	Upon completing the Deep Learning course, learners will understand and implement neural networks, including ANN, CNN, RNN, LSTM, GANs, and Autoencoders, using TensorFlow and Keras for various applications.	1,2
IV	**Introduction to Natural Language Processing and Computer Vision:** Introduction to Natural Language Processing, NLP application in real world, components of NLP, Vartual Agents overview, Chatbot and its parts, Tokenization, Stemming & Lemmatization in NLP, BagofWords,TF-IDF,Word2Vec, Word Embedding in NLP. Introduction to Computer Vision, Image Classification and Object Recognition, Feature Detection and Description	**8**	Upon completing the course, learners will understand and apply NLP techniques like tokenization, stemming, and word embeddings, as well as computer	1, 2, 3

	algorithms with OpenCV.		vision methods for image classification and object recognition using tools like OpenCV.	
V	**Evolution from DeepQA to Watson services on IBM Cloud:** Explain the DeepQA architecture was;Explain why IBM decided to commercialize Watson and Describe the evolution of Watson services from the original DeepQA architecture to the present. Introduction to IBM Watson: explore Watson Studio, IBM's integrated environment for data scientists, developers, and domain experts, to build and deploy ML models. Creating Chatbots using Watson Assistant: learn the basics of Watson Assistant, including intents, entities, dialog flows, and integration capabilities. Natural Language Understanding (NLU), Language Translator, Text to Speech, Speech to Text on IBM Cloud.	**8**	Upon completing the course, learners will understand the evolution of Watson from DeepQA, explore Watson Studio for building and deploying machine learning models, and grasp the significance of Watson's transition to IBM Cloud services.	1, 2, 3
Practical				
Practical 1	Implementation of uninformed Algorithm	**2**	Uninformed algorithms, like BFS and DFS, explore all possible paths in a problem space without prior knowledge of the goal, ensuring an exhaustive search to find a solution, typically used for pathfinding or graph traversal.	1
Practical 2	Implementation of DFS for water jug problem	**4**	In the DFS water jug problem, the algorithm explores all	1, 2

			possible states of water distribution in the jugs by recursively checking each possible action, ultimately finding a solution to reach the desired amount of water in one of the jugs.	
Practical 3	Implementation of DFS for Tic-Tac-Toe problem	**4**	The implementation of DFS for the Tic-Tac-Toe problem explores all possible game states, recursively checking each move to find the optimal solution or outcome	2, 3
Practical 4	Implementation N-Qeen problem experiment outcome	**4**	The implementation of the N-Queens problem using DFS explores all possible placements of queens on an N×N chessboard, recursively trying to place queens one by one in safe positions, ensuring no two queens threaten each other, and ultimately finding all valid solutions.	2

Practical 5	Implementation of Travel salesman problem	**4**	The implementation of the Traveling Salesman Problem (TSP) using algorithms like DFS or dynamic programming explores all possible routes between cities, aiming to find the shortest possible route that visits each city once and returns to the starting point, optimizing travel time or distance.	3
Practical 6	Python libraries for Deep learning	**6**	Python libraries for deep learning, such as **TensorFlow**, **Keras**, and **PyTorch**, provide robust frameworks	3
Practical 7	Implementation of face recognition using DL.	**4**	The implementation of face recognition using deep learning leverages convolutional neural networks (CNNs) to extract features from facial images, compare them to a database of known faces, and accurately	3

			identify individuals, achieving high accuracy in tasks like security, authentication, and surveillance.	

TEXT BOOKS:

T1 : Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, 1st Edition, MIT Press, 2016
T2 : Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall, 2010

REFERENCE BOOKS:

R1 : Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd edition. O'Reilly Media, 2019.

R2 : Michael Nielsen, Neural Networks and Deep Learning, First Edition, Determination Press, 2015

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Remember and understand relevance of AI in today's world.	**1, 2, 3, 4, 5, 12**
2	Gain an idea about Machine Learning application in real world problem.	**1, 2, 3, 4, 5**
3	Develop an understanding deep learning concept in AI.	**1, 2, 3, 5**
4	Apply and Analyze data by applying various machine learning algorithms.	**1, 2, 3, 5**
5	Enhance problem-solving abilities with IBM Cloud service.	**1, 2, 5, 7, 8**

SEMESTER – IV									
Course Title	**PROGRAMME ELECTIVE II - DATA ENGINEERING**								
Course code	**24MCAM2202R**	**Total credits: 3 Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Basic Programming and Database management skills**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Applications**								
Semester	**Winter/ IV semester of the second year of the programme**								
Course Objectives	1. To acquire mastery of data mining fundamentals and techniques 2. To gain skills to design and implement mathematical models for effective decision making in business environments 3. To develop an understanding of natural language processing (NLP) techniques, focusing on text pre-processing, sentiment analysis, and topic modelling								
CO1	Develop an understanding of the data mining process and issues								
CO2	Understand various techniques for data mining								
CO3	Apply the techniques in solving data mining problems using data mining tools and systems								
CO4	Design mathematical model for decision making using business intelligence								
CO5	Analyse and apply Natural Language Processing								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Introduction to Data Mining: Overview of Data Mining: Understanding its importance and applications. Data Mining Process: Steps from data collection to knowledge discovery. Types of Data: Structured, unstructured, and semi-structured data.	**6**	Understand the Concept and Importance of Data Mining, Identify Applications of Data Mining, Understand the Steps of the Data Mining Process, Recognize the Challenges in Data Mining	1, 2, 3
II	Data Pre-processing and Transformation: Data Cleaning: Techniques to handle missing values, outliers, and noise Data Transformation: Normalization, scaling, and discretization, Feature selection and dimensionality reduction Data Integration and Warehousing: Data warehouse concepts (OLTP vs. OLAP), ETL (Extract, Transform, Load) processes	**6**	Understand Data Cleaning Techniques, Handle Outliers and Noise, Apply Data Normalization and Scaling, Implement Discretization, Conduct Feature Selection and Dimensionality Reduction, Understand Data Warehouse Concepts	3, 4
III	Data Mining Techniques: Classification: Decision trees	**6**	Understand Decision Trees, Apply k-Nearest Neighbors (k-	3, 4

	(e.g., C4.5, ID3), k-Nearest Neighbors (k-NN), Support Vector Machines (SVMs), Clustering: k-Means clustering, Hierarchical clustering Association Rule Mining: Apriori algorithm, Measures of interestingness (support, confidence), Market Basket Analysis		NN), Utilize Support Vector Machines (SVMs), Apply k-Means Clustering, Understand Hierarchical Clustering, Apply the Apriori Algorithm, Measure Interestingness, Conduct Market Basket Analysis	
IV	Model Evaluation and Business Applications: Performance Metrics: Confusion matrix, accuracy, precision, recall, F1-score, ROC curves and AUC Model Selection and Optimization: Cross-validation, hyperparameter tuning Business Applications: Case studies in customer segmentation, churn prediction, fraud detection, recommender systems	**6**	Understand and Use the Confusion Matrix, Evaluate Models with Accuracy, Precision, Recall and F1-Score, Analyze Models Using ROC Curves and AUC, Apply Cross-Validation Techniques, Perform Hyperparameter Tuning, Understand and Apply Customer Segmentation	4
V	Natural Language Processing (NLP): NLP Fundamentals: Text preprocessing (tokenization, stemming, lemmatization), Bag-of-words, TF-IDF Sentiment Analysis: Lexicon-based and machine learning approaches Topic Modelling: LDA (Latent Dirichlet Allocation) Applications: Text classification, chatbots, information extraction	**6**	Understand and Implement Text Preprocessing Techniques, Apply Bag-of-Words Model, Implement Lexicon-Based Approaches, Apply Machine Learning Approaches, Understand and Implement Latent Dirichlet Allocation (LDA), Perform Text Classification, Develop and Implement Chatbots	4
Practical				
Practical 1	Implement a program to demonstrate the steps of the data mining process using a simple dataset, from data collection to knowledge discovery. Provide insights into the importance and applications of data mining.	**2**	Understand the steps of the data mining process and recognize its importance and applications.	2, 3
Practical 2	Write a program to perform data cleaning on a dataset. Include techniques to handle missing values, outliers, and noise.	**2**	Understand and apply data cleaning techniques.	3, 4

Practical 3	Implement a program to demonstrate data normalization, scaling, and discretization. Include feature selection and dimensionality reduction techniques.	**2**	Apply data transformation techniques and conduct feature selection and dimensionality reduction.	3, 4
Practical 4	Use a data mining tool (e.g., Weka, Orange) to implement decision tree classification on a given dataset. Analyse the results.	**2**	Apply decision tree classification using a data mining tool.	3, 4
Practical 5	Use a data mining tool to perform k-means clustering on a dataset. Visualize and interpret the results.	**2**	Apply k-means clustering using a data mining tool.	3, 4
Practical 6	Write a program to implement the Apriori algorithm for market basket analysis. Measure the interestingness of the rules using support and confidence.	**2**	Apply the Apriori algorithm for market basket analysis and measure interestingness.	3, 4
Practical 7	Implement a program to evaluate a classification model using performance metrics such as confusion matrix, accuracy, precision, recall, F1-score, ROC curves, and AUC.	**2**	Evaluate models using various performance metrics.	3, 4
Practical 8	Write a program to perform text preprocessing, including tokenization, stemming, and lemmatization on a given text dataset.	**2**	Understand and implement text preprocessing techniques.	3, 4
Practical 9	Implement a program to perform sentiment analysis using both lexicon-based and machine learning approaches. Compare their performance.	**2**	Apply sentiment analysis using different approaches and analyse their performance.	4, 5
Practical 10	Use a data mining tool or library (e.g., Gensim in Python) to implement Latent Dirichlet Allocation (LDA) for topic modeling on a text dataset. Analyse the results.	**2**	Understand and apply Latent Dirichlet Allocation (LDA) for topic modeling.	3, 4

TEXT BOOKS:

T1: Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition

T2: Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining, 2nd Edition

REFERENCE BOOKS:

R1: Christopher M. Bishop, Pattern Recognition and Machine Learning, 1st Edition

R2: Jure Leskovec, Anand Rajaraman and Jeff Ullman, Mining of Massive Datasets, 3rd Edition

R3: Steven Bird, Ewan Klein, and Edward Loper, Natural Language Processing with Python, 1st Edition

OTHER LEARNING RESOURCES:

https://onlinecourses.nptel.ac.in/noc21_cs69/preview

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Develop an understanding of the data mining process and issues	**1, 2, 4**
2	Understand various techniques for data mining	**1, 2, 5**
3	Apply the techniques in solving data mining problems using data mining tools and systems	**3, 5, 11**
4	Design mathematical model for decision making using business intelligence	**1, 3, 5**
5	Analyse and apply Natural Language Processing	**1, 2, 4, 5**

SEMESTER – IV									
Course Title	**PG RESEARCH PROJECT II**								
Course code	**24MCAM2203R**	**Total credits: 12**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			0	**0**	**4**	**8**	**56**	**0**	**12**
Pre-requisite	**Nil**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Winter/ IV semester of the second year of the programme**								
Course Objectives	1. Develop foundational research skills, mastering data collection and analysis techniques for accurate interpretation. 2. Cultivate effective documentation practices, emphasizing clarity, transparency, and ethical considerations. 3. Acquire proficiency in utilizing research tools, fostering continuous learning, and building a comprehensive research portfolio.								
CO1	Demonstrate proficiency in applying statistical analysis techniques to interpret data accurately and effectively in various contexts.								
CO2	Develop the capability to effectively interpret and draw meaningful conclusions from data sets, enhancing their decision-making skills.								
CO3	Engage in in-depth discussions and critically evaluate theories, methodologies, and research findings to enhance understanding and foster intellectual growth.								
CO4	Articulate the scope and limitations of features within projects, ensuring a thorough understanding of project boundaries and deliverables.								
CO5	Master the skills necessary to deliver professional and coherent final presentations								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	**Statistical Analysis**: Introduction to Statistical concepts	**4**	Understand Fundamental Statistical Concepts, Apply Descriptive Statistical Techniques, Conduct Basic Statistical Tests	1, 2, 3
II	**Data interpretation**: Parametric and Non-parametric Tests: t-tests, ANOVA, Chi-square tests Interpretation of Results: Drawing meaningful conclusions	**8**	Understand the Difference Between Parametric and Non-Parametric Tests, Conduct and Interpret t-tests, Perform and Analyse ANOVA (Analysis of Variance), Execute and Interpret Chi-Square Tests and draw interpretations	3, 4
III	**Discussions**: Implications of findings Addressing limitations	**8**	Analyse the Broader Impact of Research Findings, Draw Practical and Theoretical Conclusions, Communicate Findings Effectively, Identify and Acknowledge Research Limitations, Propose Solutions and Future Research Directions	3, 4
IV	**Future scope of the study**: Research Design and Methodology, Emerging Trends in Research	**8**	Identify Potential Areas for Future Research, Design Follow-Up Studies, Predict the Impact of Future Research, Construct	4

	Collaboration and Interdisciplinary Research		Robust Research Designs, Implement Advanced Methodologies, Ensure Rigorous Data Collection and Analysis	
V	**Final presentation of the thesis:** Scientific Writing, Effective Oral Presentation, Ethical Considerations, Research Communication	**4**	Demonstrate Proficiency in Scientific Writing, Effectively Communicate Research Findings, Adhere to Academic Standards and Guidelines, Deliver Clear and Engaging Oral Presentations, Communicate Research Findings Effectively	4

TEXT BOOKS:

T1: John W. Creswell and J. Creswell, Research Design: Qualitative, Quantitative and Mixed Methods

T2: Alan Agresti and Barbara Finlay, Statistical Methods for the Social Sciences, 3rd Edition

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Demonstrate proficiency in applying statistical analysis techniques to interpret data accurately and effectively in various contexts.	**1, 2, 4**
2	Develop the capability to effectively interpret and draw meaningful conclusions from data sets, enhancing their decision-making skills.	**2, 4, 9**
3	Engage in in-depth discussions and critically evaluate theories, methodologies, and research findings to enhance understanding and foster intellectual growth.	**2, 6, 12**
4	Articulate the scope and limitations of features within projects, ensuring a thorough understanding of project boundaries and deliverables.	**3, 6, 11**
5	Master the skills necessary to deliver professional and coherent final presentations	**6, 9, 12**

SEMESTER – IV									
Course Title	**PROGRAMME ELECTIVE II - PATTERN RECOGNITION**								
Course code	**24MCAM2202R**	**Total credits: 3 Total hours: 30T+30P**	**L**	**T**	**P**	**S**	**R**	**O/F**	**C**
			2	**0**	**2**	**0**	**0**	**0**	**3**
Pre-requisite	**Fundamentals of probability, linear algebra, and Digital Image Processing**	**Co-requisite**	**Nil**						
Programme	**Master of Computer Application**								
Semester	**Winter/IV semester of the second year of the programme**								
Course Objectives	1. Pattern recognition which has several important applications on classical computer engineering problems like word/sentence-based searches, etc. 2. The emphasis of the course is on algorithms for pattern recognition 3. The representation of patterns and classes and the proximity measures are an important aspect of pattern recognition								
CO1	Summarize the various techniques involved in pattern recognition.								
CO2	Categorize the various pattern recognition techniques into supervised and unsupervised								
CO3	Understand feature selection and extraction techniques.								
CO4	Illustrate the artificial neural network-based pattern recognition								
CO5	Discuss the applications of pattern recognition in various real-world applications								

Unit-No.	**Content**	**Contact Hour**	**Learning Outcome**	**KL**
I	Introduction and Mathematical Preliminaries: Pattern Recognition: Definition, Applications and Examples, Clustering Vs Classification, Supervised Vs Unsupervised, Basic of Linear Algebra, Vector Spaces, Basics of Probability, Basics of Estimation Theory, Decision Boundaries, Decision Regions, Metric Spaces	**6**	Explain Pattern Recognition, Understanding different Clustering Techniques and their theory. Applying in real world scenario.	1, 2, 4
II	Classification: Bayes Decision Rules, Error Probability, Examples, Normal Distribution, Linear Discriminant Function, Non-Linear Decision Boundaries, Mahalanobis Distance, K-NN Classifier, Single and Multi-Layer Perceptron, Training Set, Test Set, Standardization and Normalization	**6**	Explaining different Classification technique and understanding their interpretation by applying in different datasets.	1, 2, 4
III	Clustering: Basics, Similarity/Dissimilarity Measures, Clustering Criteria, Different distance functions and similarity measures, within cluster distance criterion, K-means algorithm, Single linkage and complete linkage algorithms, MST, K-medoids, DBSCAN, Data sets: Visualization, Unique Clustering	**6**	Explaining different Clustering technique and understanding their interpretation by applying in different datasets.	1, 2, 4
IV	Feature Selection and Extraction: Problem	**6**	Explaining	1, 2,

	statement and Uses, Branch and Bound Algorithm, Sequential Forward / Backward Selection Algorithms, (l, r) algorithm, Probabilistic separability based criterion functions, interclass distance based criterion functions, Feature Extraction, PCA + Kernel PCA		different Feature Selection and Extraction technique and understanding their role in enhancing models.	3, 4
V	Recent Advances in Pattern Recognition: Structural PR, SVMs, FCM, Soft-Computing and Neuro-Fuzzy Techniques, Real-Life Examples	**6**	Understanding Recent Advance Pattern Recognition Technique with Real life Examples	1,2,3, 4
Practical				
Practical 1	Implement a program to demonstrate the difference between clustering and classification using simple datasets. Explain the key differences and provide real-world scenarios where each technique is applicable.	**3**	Explain Pattern Recognition, understanding different clustering techniques and their theory. Applying in real-world scenarios.	2, 3, 4
Practical 2	Write a program to classify a dataset using both a supervised learning algorithm (e.g., K-NN) and an unsupervised learning algorithm (e.g., K-means). Compare their results and discuss the differences.	**3**	Understanding the differences between supervised and unsupervised learning techniques and their applications.	2, 3, 4
Practical 3	Implement the Bayes decision rule for classification and test it on a sample dataset. Calculate the error probability and discuss the results.	**3**	Explain different classification techniques and understand their interpretation by applying them to different datasets.	2, 3, 4
Practical 4	Write a program to implement the sequential forward selection algorithm for feature selection. Apply it to a dataset and analyse the selected features' impact on model performance.	**3**	Explain different feature selection techniques and understand their role in enhancing models.	2, 3, 4
Practical 5	PCA for feature extraction and apply it to a high-dimensional dataset. Visualize the reduced features and discuss their significance.	**3**	Explain feature extraction techniques and understand their role in enhancing	3, 4

			models.	
Practical 6	Write a program to implement a single-layer and multi-layer perceptron neural network for pattern recognition. Train the network on a dataset and evaluate its performance.	**3**	Illustrate neural network-based pattern recognition.	2, 3, 4
Practical 7	Implement an SVM classifier and apply it to a real-world dataset (e.g., handwritten digit recognition). Discuss the results and the classifier's effectiveness.	**3**	Understanding recent advances in pattern recognition techniques with real-life examples.	2, 3, 4
Practical 8	Develop a pattern recognition system for a real-life application (e.g., spam email detection). Use appropriate feature selection, extraction, and classification techniques. Discuss the system's effectiveness and potential improvements.	**3**	Discuss the applications of pattern recognition in various real-world scenarios.	3, 4
Practical 9	Implement the DBSCAN clustering algorithm and apply it to a dataset with noise. Visualize the clusters and discuss the results.	**3**	Explain different clustering techniques and understand their interpretation by applying them to different datasets.	2, 3, 4
Practical 10	Implement a simple neuro-fuzzy system for pattern recognition. Apply it to a dataset and analyse the results. Discuss how neuro-fuzzy techniques combine neural networks and fuzzy logic for pattern recognition.	**3**	Understanding recent advances in pattern recognition techniques with real-life examples.	2, 3, 4

TEXT BOOKS:

T1: J.I. Tou & R.C. Gonzalez, Pattern Recognition Principles, Addition-Wesley.

T2: MR. Schalkoff, Pattern Recognition - Statistical, Structural and Neural Approaches, John Wiley, 1992.

RELATIONSHIP BETWEEN COURSE OUTCOMES (CO) AND PROGRAM OUTCOMES

CO PO Mapping		
SN	**Course Outcome (CO)**	**Mapped Program Outcome**
1	Summarize the various techniques involved in pattern recognition.	**1, 2, 4**
2	Categorize the various pattern recognition techniques into supervised and unsupervised	**1, 2, 6**
3	Understand feature selection and extraction techniques.	**1, 4, 5**
4	Illustrate the artificial neural network-based pattern recognition	**3, 5, 6**
5	Discuss the applications of pattern recognition in various real-world applications	**7, 8, 9**