Four-Year Undergraduate Programme

Bachelor of Technology

Computer Science & Engineering – Artificial Intelligence (AI)

Faculty of Engineering & Technology

Parul University

Vadodara, Gujarat, India

Faculty of Engineering & Technology

Bachelor of Technology in Computer Science & Engineering– Artificial Intelligence (AI)

1. **Vision of the Department**
   To be a distinct hub of education that prepares skilled professionals in the field of computer science and engineering.

2. **Mission of the Department**

M 1	Enhance academic performance by adopting industry-oriented curriculum focusing on the key area of computer education through integrated learning in collaboration with prominent industries.
M 2	Preparing students to face the challenges of the real world through internships and project based learning.
M 3	Foster a research culture that results in a solid knowledge base, high-quality publications, new products and IPR.
M 4	Inculcate ethical consciousness in students so that they can achieve success in their professional endeavors and become responsible citizens.

3. **Program Educational Objectives**
   The statements below indicate the career and professional achievements that the B.Tech - Computer Science engineering curriculum enables graduates to attain.

PEO 1	To develop technical skills (critical investigation, communication, analytical and computer) and human relations skills (group dynamics, team building, organization and delegation) to enable students to transform the acquired knowledge into action.
PEO 2	To inculcate critical analysis and communication skills into students to effectively present their views, both in writing and through oral presentations.
PEO 3	To provide an environment for exploring the Research & Development attitude, to help the students in the Research and Development field

4. **Program Learning Outcomes**
   Program Learning outcomes are statements conveying the intent of a program of study

PLO 1	**Engineering Knowledge**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PLO 2	**Problem Analysis**	Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using the first principles of mathematics, natural sciences, and engineering sciences.
PLO 3	**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 public health and safety, and cultural, societal, and environmental considerations.
PLO 4	**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.
PLO 5	**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.
PLO 6	**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.
PLO 7	**Environment and sustainability**	Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of, and need for sustainable development.
PLO 8	**Ethics**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PLO 9	**Individual and team work**	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings
PLO 10	**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.
PLO 11	**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
PLO 12	**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.

5. Program Specific Learning Outcomes

PSO 1	**Demand as per recent development**	An ability to analyse, design, verify, validate, code and maintain the solution of given problem to derive execution of software system
PSO 2	**Software skill**	An ability to understand, apply and work with one or more domain using knowledge of mathematical techniques and principles with relevant areas of computer science

6. Credit Framework

Table 1: **Semester wise Credit distribution of the programme**

Semester	**Credit**
Semester-1	16
Semester-2	20
Semester-3	22
Semester-4	23
Semester-5	21
Semester-6	24
Semester-7	27
Semester-8	14
Total Credits:	**167**

Table 2: **Category wise Credit distribution of the programme**

Category	**Credit**
Major Core	84
Minor Stream	0
Multidisciplinary	24
Ability Enhancement Course	9
Skill Enhancement Courses	9
Value added Courses	16
Summer Internship	16
Research Project/Dissertation	9
Total Credits:	**167**

7. Program Curriculum

Semester 1						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	-	Web Design	4	3	2	0
2	303191101	Mathematics -1	4	4	0	0
3	303105104	Computational Thinking for Structural Design-I	4	3	2	0
4	-	Principles of Artificial Intelligence	3	3	0	0
5	303106103	Electrical and Electronics Engineering	4	3	2	0
6	-	CDC-1	1	0	2	0
7	-	Student Induction Program with essence of Indian Knowledge System - 3 weeks	0	0	0	0
		Total	**20**	**14**	**8**	**0**
Semester 2						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
8	-	Programming for AI	4	3	2	0
9	303191151	Mathematics - 2	4	4	0	0
10	303105151	Computational Thinking for Structural Design-II	4	3	2	0

11	303192102	Engineering Physics	4	3	2	0
12	-	Design, Technology and Innovation	3	3	0	0
13	-	CDC-2	2	1	2	0
14	303104105	Environmental Science	0	1	0	0
		Total	**21**	**18**	**8**	**0**
Semester 3						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
15	303105201	Data Structure & Algorithms	3	3	0	0
16	303105202	Data Structure & Algorithms Laboratory	2	0	4	0
17	303105203	Database Management System	3	3	0	0
18	303105204	Database Management System Laboratory	1	0	2	0
19	303105205	Object Oriented Programming	2	2	0	0
20	303105206	Object Oriented Programming Laboratory	1	0	2	0
21	303105210	Computer Organization and Microprocessor	3	3	0	0
22	303105211	Computer Organization and Microprocessor Laboratory	1	0	2	0
23	303191202	Discrete Mathematics	4	3	0	1
24	303193203	Professional Communication Skills	2	2	0	0
		Total	**22**	**16**	**10**	**1**
Semester 4						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
25	303105251	Operating System	3	3	0	0
26	303105252	Operating System Laboratory	1	0	2	0
27	303105253	Software Engineering	3	3	0	0
28	303105254	Software Engineering Laboratory	1	0	2	0
29	303105255	Computer Network	3	3	0	0
30	303105256	Computer Network Laboratory	1	0	2	0
31	303105257	Programming in Python with Full Stack Development	3	3	0	0
32	303105258	Programming in Python with Full Stack Development Laboratory	1	0	2	0
33	303191251	Probability, Statistics and Numerical Methods	4	3	0	1
34	303193252	Professional Grooming and Personality Development	1	1	0	0
35	303105259	Competitive Coding	2	0	4	0
		Total	**23**	**16**	**12**	**1**
Semester 5						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
36	303105218	Design and Analysis of Algorithms	3	3	0	0
37	303105219	Design and Analysis of Algorithms Laboratory	2	0	4	0

38	303105306	Theory of Computation	3	3	0	0
39	303105307	Artificial Intelligence	3	3	0	0
40	303105308	Artificial Intelligence Laboratory	1	0	2	0
41	303105309	Enterprise Programming	2	2	0	0
42	303105310	Enterprise Programming Laboratory	1	0	2	0
43	303193304	Professionalism & Corporate Ethics	1	1	0	0
44	303105311	QR (Quant and Reasoning)	3	3	0	0
45		Open Elective 01 (Compulsory Subjects :1)	2	1	2	0
		Total	**21**	**16**	**10**	**0**
Semester 5 Open Elective-1						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	03101346	Basic Aircraft Science	2	1	2	0
2	303104305	Disaster Preparedness and Planning	2	1	2	0
3	303105301	AWS Fundamentals	2	1	2	0
4	303105305	Internet of Things	2	1	2	0
5	303107346	Fundamentals of Communication Engineering	2	1	2	0
Semester 6						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
46	303105336	Project – I	3	0	0	0
47	303105353	Machine Learning	3	3	0	0
48	303105354	Machine Learning Laboratory	1	0	2	0
49	303105349	Compiler Design	3	3	0	0
50	303105350	Compiler Design Laboratory	1	0	2	0
51	303105355	High Performance Computing	3	3	0	0
52	303105356	High Performance Computing Laboratory	1	0	2	0
53		PEC 01 (Compulsory Subjects :1)	3	3	0	0
54		PEC 01-LAB (Compulsory Subjects :1)	1	0	2	0
55		PEC 02 (Compulsory Subjects :1)	3	3	0	0
56		PEC 02-LAB (Compulsory Subjects :1)	1	0	2	0
57	303193353	Employability Skills	1	1	0	0
		Total	**24**	**16**	**10**	**0**
Semester 6 PEC-01						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105344	Data Analytics and Visualization	3	3	0	0
2	303105341	Cyber Security	3	3	0	0
3	303105363	Cloud Computing	3	3	0	0
Semester 6 PEC-01 LAB						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105345	Data Analytics and Visualization Laboratory	1	0	2	0
2	303105342	Cyber Security Laboratory	1	0	2	0
3	303105364	Cloud Computing Laboratory	1	0	2	0
Semester 6 PEC-02						

Sr. No.	Subject Code	Subject Name	Credit	Lect	Lab	Tut
1	303105379	Mobile Application Development	3	3	0	0
2	303105352	MEA(R)N Stack Web Development	3	3	0	0
3	303105354	Devops	3	3	0	0
Semester 6 PEC-02 LAB						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105380	Mobile Application Development Laboratory	1	0	2	0
2	303105353	MEA(R)N Stack Web Development Laboratory	1	0	2	0
3	303105355	Devops Laboratory	1	0	2	0
Semester 7						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
58	303105431	Summer Internship	2	0	0	0
59	303105479	Deep Learning with NLP	3	3	0	0
60	303105480	Deep Learning with NLP Laboratory	1	0	2	2
61	303105433	Project – II	6	0	12	0
62	303105481	Pattern Recognition	3	3	0	0
63	303105482	Pattern Recognition Laboratory	1	0	2	0
64		PEC 03 (Compulsory Subjects :1)	3	3	0	0
65		PEC 03-LAB (Compulsory Subjects :1)	1	0	2	0
66		PEC 04 (Compulsory Subjects :1)	3	3	0	0
67		PEC 04-LAB (Compulsory Subjects :1)	1	0	2	0
68		Open Elective-2	3	3	0	0
		Total	**27**	**15**	**20**	**2**
Semester 7 PEC-03						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105377	Software Testing and Quality Assurance	3	3	0	0
2	303105436	Internet of Things	3	3	0	0
3	303105440	Modern networks	3	3	0	0
Semester 7 PEC-03 LAB						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105378	Software Testing and Quality Assurance Laboratory	1	0	2	0
2	303105437	Internet of Things Laboratory	1	0	2	0
3	303105441	Modern networks Laboratory	1	0	2	0
Semester 7 PEC-04						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105381	Image Processing	3	3	0	0
2	303105443	Block Chain Technologies	3	3	0	0
3	303105444	Augmented Reality and Virtual Reality	3	3	0	0
Semester 7 PEC-04 LAB						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105382	Image Processing Laboratory	1	0	2	0
2	303105446	Block Chain Technologies Laboratory	1	0	2	0

3	303105444	Augmented Reality and Virtual Reality Laboratory	1	0	2	0
Semester 7 Open Elective-2						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
1	303105448	Remote Sensing and Geo Informatics	3	3	0	0
2	303105449	Real Time Systems	3	3	0	0
3	303105450	Cyber Physical Systems	3	3	0	0
4	303105451	Computational Number Theory	3	3	0	0
5	303105452	VLSI System Design	3	3	0	0
Semester 8						
Sr. No.	**Subject Code**	**Subject Name**	**Credit**	**Lect**	**Lab**	**Tut**
69	303105499	Internship	14	0	28	0
		Total	**14**	**0**	**28**	**0**

Semester 1-1

a. Course: Web Design

b. Course Code:

c. Prerequisite: Basic knowledge of programming and computer fundamentals

d. Rationale: The course will cover key concepts, programming languages, and industry practices to develop modern websites.

e. Course Learning Objectives:

CLOBJ 1	Understand the fundamentals of web technologies, architecture, and web hosting principles.
CLOBJ 2	Develop structured and accessible web pages using HTML5 and semantic markup.
CLOBJ 3	Apply CSS3 for styling and layout design, including transitions, animations, and responsive design techniques.
CLOBJ 4	Implement client-side interactivity using modern JavaScript features and DOM manipulation.
CLOBJ 5	Use frontend frameworks and libraries such as Bootstrap, Tailwind CSS, jQuery, and React.js for efficient UI development.
CLOBJ 6	Gain basic knowledge of backend technologies, database integration, version control, and deployment of web applications.

f. Course Learning Outcome:

CLO 1	To learn the concepts of Web design and its types.
CLO 2	To understand the basic concepts of HTML
CLO 3	To understand and learn the scripting languages with the design of web Applications.
CLO 4	Design Website using HTML CSS and JS

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction to Web Design & Technologies:** Basics of Web & Internet, Web Design Principles & Trends, Client-Server Architecture, HTTP/HTTPS, Web Hosting, Domain Names	5%	4
2	**HTML5 & Web Structure :** HTML Elements- Nesting and structuring HTML elements. Attributes, Forms, Tables, Lists, Media Embedding, Semantic HTML, Accessibility	15%	7
3	**CSS3 & Styling:** CSS – Formatting text – Colours and Background – Padding, Borders and Margins – Floating and positioning – Page Layout with CSS – Transition, Transforms and Animation	15%	7
4	**Responsive Web Design:** Sass for Responsive Web Design – Marking Content with HTML5 – Mobile-First or Desktop-First – CSS Grids, CSS Frameworks, UI Kits, and Flexbox for RWD – Designing small UIs by Large Finger -Images and Videos in Responsive Web Design – Meaningful Typography for Responsive Web Design	10%	5
5	**JavaScript & Interactivity :** JS Fundamentals (ES6+), DOM Manipulation & Event Handling- Document Object Model (DOM) to dynamically access, modify, and HTML elements using JavaScript. event listeners, handling user interactions, and manipulating elements in response to events for dynamic web applications, Form Validation & Data Handling	20%	8
6	**Frontend Frameworks & Libraries :** Introduction to Bootstrap, Tailwind CSS, jQuery Basics - Learning Bootstrap for responsive design with prebuilt components, Tailwind CSS for utility-first styling, and jQuery for simplified JavaScript operations. Implementing grids, flex utilities, animations, event handling, and DOM manipulation for efficient web development., Introduction to React.js or Vue.js	20%	8
7	**Basic Backend & Web Deployment :** Server-Side Scripting (PHP/Node.js/Python), Databases (MySQL/MongoDB) , REST APIs & Fetching Data Version Control with Git/GitHub, Web Hosting & Deployment	15%	6

i. Reference Books:

1. **HTML & CSS: Design and Build Websites"** By Jon Duckett
2. **JavaScript: The Definitive Guide** By David Flanagan
3. **Tailwind CSS in Actionl Intelligence-A Modern Approach** By Chris Sev
4. **Learning React** By Alex Banks, Eve Porcello
5. **Node.js Design Patterns** By Mario Casciaro
6. **Web Security for Developers** By Malcolm McDonald
7. **Search Engine Optimization (SEO) For Dummies** By Peter Kent

j. List of Practicals:

1. Study of Visual Studio environment. (Create new web project, Open existing web project, building website, and study of toolbars, menu etc.)
2. Develop a web form to perform add, update, delete operation on List Box control.
3. Design a web page to implement upload and download files functionality using File Upload Control.
4. Develop a web page to implement the concept of state management using Cookie
5. Develop a web page to implement the concept of state management using Session and Application
6. Develop a web page to implement the concept of state management using View State and Query String.
7. Design a page that takes name and message from the user and choose a color by radio button, select a style for ex.-bold, italic, underline from the checkbox and display in label control, when you clicked on display button. And clear the information when you clicked on clear button
8. Write sample application to connect to database, Fetching and inserting data from database and using Data Reader.
9. Develop a User Registration form for any application
10. Develop a web application to Add, Update, View and Delete records from Database data shown in Grid view.

Semester 1 - 2

a. Course: Mathematics-I

b. Course Code: 303191101

c. Prerequisite: Knowledge of Mathematics up to 12th science level

d. Rationale: The Mathematics I syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and matrix algebra, preparing students for engineering challenges with optimized problem-solving skills.

e. Course Learning Objectives:

CLOBJ 1	Develop a comprehensive understanding of definite and improper integrals, including the application of integration techniques to find areas and volumes in both Cartesian and Polar coordinates.
CLOBJ 2	Utilize differential equations to model and solve practical scenarios, demonstrating proficiency in various solution techniques.
CLOBJ 3	Analyze the convergence and divergence of sequences and series, employing tests such as the Alternating Series Test and Ratio Test.
CLOBJ 4	Analyze matrix operations and determinants, exploring their properties and applications in solving systems of linear equations.
CLOBJ 5	Apply Fourier series for representing periodic functions, verifying Dirichlet's conditions.
CLOBJ 6	Solve optimization problems using multivariable calculus concepts, such as Lagrange's multiplier.

f. Course Learning Outcomes:

CLO 1	Develop understanding of fundamental mathematical concepts.
CLO 2	Formulate and solve mathematical models for real-world engineering problems.
CLO 3	Integrate knowledge from different mathematical topics to analyze and solve complex engineering problems.
CLO 4	Critically analyze mathematical results, interpret their engineering significance, and make informed decisions based on mathematical outcomes, fostering a deeper understanding of the subject.
CLO 5	Clearly and effectively communicate mathematical ideas, solutions, and reasoning, both in written and oral formats, demonstrating effective communication skills.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
4	-	-	4	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT 1: Improper Integral & Application of Definite Integral** Evaluation of definite and improper integrals, Beta and Gamma functions and their properties. Area bounded by curves in Cartesian and Polar form, Area of a region bounded by function, Area of a region bounded by curves in Parametric form, Volume by slicing, Volume of solid by revolution.	8%	5
2	**UNIT 2: First Order Ordinary Differential Equation** 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, Applications.	15%	9
3	**UNIT 3: Matrices** Matrices & Determinants with Properties, Linear Independence, Rank of Matrix, System of Linear Equations, Consistency of System, Solution of system of Linear Equations by Gauss Jordan and Gauss-Elimination Method, Eigenvalues, Eigenvectors, Symmetric, Skew-symmetric, and orthogonal Matrices, Eigenbases, Diagonalization, Cayley Hamilton Theorem and its Applications, Diagonalization, Orthogonal Transformation, Quadratic form.	25%	15

4	**UNIT 4: Sequences and Series** Basics of Sequences, Bounded and Monotonic Sequences, Series, Convergence of sequence and series, Geometric series, P-series, Cauchy's Integral Test, Comparison Test, Alternating Series, Absolute and Conditional convergence, Ratio test, Cauchy's Root Test, Power series, Taylor's and Maclaurin's series.	17%	10
5	**UNIT 5: Fourier Series** Fourier Series of 2 periodic functions, Dirichlet's conditions for representation by a Fourier series, Fourier Series of a function of period 2, Fourier Series of even and odd functions, Half range series.	10%	6
6	**UNIT 6: Multivariable Calculus (Differentiation)** Functions of Several Variables, Limit, Continuity, Partial Derivatives, Homogeneous function, Euler's Theorem for homogeneous function, Modified Euler's Theorem, Chain Rule, Implicit function, Jacobian, Tangent plane and Normal line, Maximum and Minimum Values, Lagrange's Multiplier, Taylor's and Maclaurin's Series for functions of two variables.	25%	15

i. Reference Books:

1. **Calculus and Analytic Geometry (TextBook)** By G.B. Thomas and R.L. Finney — Addison Wesley
2. **Calculus with early transcendental functions** By James Stewart — Cengage Learning
3. **Higher Engineering Mathematics** By B. S. Grewal — Khanna Publications
4. **Elementary Linear Algebra (Text Book)** By Howard Anton, Chris Rorres — Willy India Edition — 9th Edition
5. **Advanced Engineering Mathematics (Text Book)** By Erwin Kreyszig — Willey India Education
6. **A textbook of Engineering Mathematics** By N.P. Bali and Manish Goyal — Laxmi Publications

Semester 1 - 3

a. Course: Computational Thinking for Structured Design-1

b. Course Code: 303105104

c. Prerequisite: Requires Basic Knowledge of Computer.

d. Rationale: This course is design to provide basic ideas of computer programming. This course also makes help to understand programming language. It will help to develop their logical abilities.

e. Course Learning Objectives:

CLOBJ 1	Programming basics and the fundamentals of C.
CLOBJ 2	Data types in C.
CLOBJ 3	Mathematical and logical operations.
CLOBJ 4	Using if statement and loops.
CLOBJ 5	Arranging data in arrays.
CLOBJ 6	AImplementing pointers.

f. Course Learning Outcomes:

CLO 1	Able to understand the basic knowledge of Computer fundamental and its application in computers.
CLO 2	Able to understand the basic concepts of C programming language.
CLO 3	Able to design and develop various programming problems using C programming concepts.
CLO 4	Able to Implement advance C programming concepts like function, pointer, structure and union etc.
CLO 5	Able to understand the file handling using C Programming language.

g. Teaching & Examination Scheme:
L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours

1	**UNIT-I: Introduction to C language** History of C language, Program Development Steps, Structure of C program	10%	3
2	**UNIT-II: Data Types, User I/O and Operators Data Types** Extended and Derived Data types, Variables User I/O : Formatted, predefined Functions of stdio.h header file Operators: Types of operators, Precedence, Associativity.	10%	6
3	**UNIT-III: Conditional Flow Statements:** Iterative Statements, Jumping Statements and Pointors:Conditional Flow Statements: Simple if,ifelse,else-if ladder,switch case Decision Making using conditional statements Iterative Statements: Control Entry and Control Exit Loops Jumping Statements: break, continue , forward and backward goto. Pointers: Typed:single double,triple..wild, NULL,Const, untyped, void.	15%	9
4	**UNIT-IV: Functions:** Functions :Call by value, call by references, Types of Functions. Pointer Functions: Calling A function through function pointer, Passing A function's address as an Argument to other function, Types of Pointer function Creation. Recursion : Types of Recursions : Direct Recursion, Indirect Recursion, Tail Recursion, No tail/Head Recursion, Tree Recursion, Nested Recursion. Storage classes : Auto, register, static and Extern.	30%	10
5	**UNIT-V: Arrays:** Arrays: Types of arrays, Declaration and Defining an array Pointer and Arrays: Types of Accessing Array elements Subscripting pointer variables Pointer to an array, Array of pointers, Pointers and two dimensional arrays Subscripting pointer To an array, Array of Functions : Strings: Strings v/s character arrays, Initializing strings, Reading and Displaying string Types of string format Specifiers. puts() functions, Multi Line string Input String pointers, Two-dimensional character arrays or array of string Array of pointers to strings, String handling functions.	35%	14

i. Reference Books:

1. **C Programing** By Bala Guru Swamy (TextBook)
2. **C for all** By s.Thammarai Selvi ,R Murugesan, Anuradha Publications.
3. **Programing in C** By Ajay Mittal, Pearson.

j. List of Practicals:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

1. Installation C IDE, Basic Structure of C program. Format Specifiers, Escape Character. Run time input/Output Programs.
2. Write a c program to calculate Area of Rectangle,Perimeter of a Rectangle and Diagonal of a Rectangle.
3. The total distance traveled by vehicle in 't seconds is given by distance s = ut+1/2at? where 'u' and 'a' are the initial velocity (m/sec.) and acceleration(m/sec?). Write a C program to find the distance traveled at regular intervals of time given the values of 'u' and 'a'. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of 'u' and 'a'.
4. Write a C program to find the sum of individual digits of a positive integer.
5. A Fibonacci sequence is defined as follows: the first and second terms in the
6. Write a C program to find the roots of a quadratic equation.
7. Write C programs that use both recursive and non-recursive functions. 1. To find the factorial of a given integer.
8. To find the GCD (greatest common divisor) of two given integers.
9. Write a C program to find the largest integer in a list of integers,
10. Write a C program that displays the position or index in the string S where the string T begins, or -1 if S doesn't contain T
11. Write a C program to generate Pascal's triangle.
12. Write a C program to convert a Roman numeral to its decimal Equivalent.
13. Write a c program to take multiline string input and print individual string length .
14. Write a c program to reverse the individual word of a given string Explanation:input : Welcome To Bytexl output: emocleW oT lxetyB.

Semester 1-4

a. Course: Principles of Artificial Intelligence

b. Course Code:

c. Prerequisite: Basic knowledge of Linear Algebra, Probability, Calculus, Python Programming, Data Structures & Algorithms

d. Rationale: This course provides an overview of AI principles and practices to tackle complex real-world problems. It is designed to develop a fundamental understanding of problem-solving, knowledge representation, reasoning, and learning methods in AI.

e. Course Learning Objectives:

CLOBJ 1	Understand the historical evolution, foundational concepts, and future directions of Artificial Intelligence.
CLOBJ 2	Identify and analyze ethical concerns such as bias, fairness, and privacy in AI applications.
CLOBJ 3	Apply various problem-solving strategies including uninformed, informed, and adversarial search techniques.
CLOBJ 4	Design knowledge representation systems using propositional and first-order logic, and implement rule-based reasoning for decision-making.
CLOBJ 5	Demonstrate probabilistic reasoning using Bayesian networks, Markov models, and handle uncertainty in AI planning.
CLOBJ 6	Explore and implement fundamental machine learning techniques including decision trees, neural networks, and statistical learning.

f. Course Learning Outcome:

CLO 1	Understand the Scope and Evolution of AI
CLO 2	Apply Search Strategies for Problem Solving
CLO 3	Analyze Decision-Making in AI
CLO 4	Implement basic machine learning algorithms and connectionist learning models.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	0	0	50	50	0	0	0	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction:** History, Trends, Key concepts, and Future Directions	5%	3
2	**AI Ethics and Societal Impacts:** Analyze bias, fairness, and privacy concerns in AI applications. Ethical implications and future of AI in society.	15%	6
3	**Problem-Solving and Search Strategies:** Production Systems and AI, Uninformed and informed search algorithms. Adversarial search techniques in game-playing AI.	15%	6
4	**Knowledge Representation and Reasoning:** Predicate Calculus in AI: Syntax and Semantics, Expressivity, Unification, Resolution; Resolution Refutation Systems; Situation Calculus. propositional logic, first-order logic, and rule-based reasoning, Design expert systems for decision-making	20%	9
5	**Probabilistic Reasoning and Uncertainty Handling:** Uncertain Knowledge and Reasoning, Probabilities, Bayesian networks- Planning with State Space Search; Planning Graphs; Partial Order Planning, Markov models, and Hidden Markov Models (HMMs).	15%	7
6	**Decision Making:** Sequential Decision Problems, Algorithms for optimal Policies.	10%	5
7	**Machine Learning:** Learning from Observations: Overview of different forms of Learning, Learning Decision Trees, Computational Learning Theory, Statistical Learning Methods, Neural Networks and Connectionist Learning.	20%	9

i. Reference Books:

1. **Artificial Intelligence** By Patrick Henry Winston,Third Edition, Addison-Wesley Publishing Company, 2004.
2. **Principles of Artificial Intelligence** By Nils J. Nilsson, Illustrated Reprint Edition, Springer Heidelberg, 2014.
3. **Artificial Intelligence: A Modern Approach** By Stuart Russell and Peter Norvig, 3rd Edition, PHI, 2009.
4. **Quest for Artificial Intelligence** By Nils J. Nilsson, First Edition, Cambridge University Press, 2010

Semester 1-5

a. Course: Electrical and Electronics Engineering

b. Course Code: 303106103

c. Prerequisite: Knowledge of Physics and Mathematics up to 12th science level.

d. Rationale: The course provides introductory treatment of the field of Electrical Engineering to the students of various branches of engineering.

e. Course Learning Objectives:

CLOBJ 1	Master analysis techniques including Kirchhoff's laws, simplification methods, superposition, Thevenin's, and Norton's theorems for effective DC circuit analysis.
CLOBJ 2	Solve problems related to Alternating current, alternating voltage, etc, Demonstrate a clear understanding of Pure R, L C circuit and combination of RLC, Series and Parallel combination of R, L and C, etc.
CLOBJ 3	Understand different single phase and three phase circuits.
CLOBJ 4	Learn diode behaviours, rectification techniques, and transistor functions as switches and amplifiers in electronic circuits.
CLOBJ 5	Understand sensors and transducers, their applications, and differentiate between their types and functionalities in electronic systems.

f. Course Learning Outcomes:

CLO 1	UTo Illustrate basic concepts of various laws, principles and theorems associated with DC circuits for networks analysis.
CLO 2	To apply concepts of sinusoidal voltages, power relationships and showcasing knowledge of AC circuit theory using numerical and graphical representation.
CLO 3	To Compare and apply diode and transistor fundamentals, including characteristics, operation, and applications, demonstrating awareness of electronics principles.
CLO 4	To design, and implement various types of voltage regulator circuits, and understanding of power supply concepts and practical applications.
CLO 5	To adept, classify, and apply various electronic sensors and transducers, for understanding of their principles and real-world applications.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT-I: DC Circuits** Electrical circuit elements (R, L and C), voltage and current sources, Kirchhoff current and voltage laws, Mesh and Node analysis, Simplifications of networks using series and parallel combinations and star-delta conversions. Superposition, Thevenin and Norton Theorems..	10%	5
2	**UNIT-II: AC Circuits** AC Circuits Sinusoidal voltages and currents, their mathematical and graphical representation, Concept of instantaneous, peak (maximum), average and R.M.S. values, frequency, cycle, period, peak factor and form factor, phase difference, lagging, leading and in phase quantities and phasor representation. Rectangular and polar representation of phasors, pure inductance, pure capacitance and corresponding voltage- current phasor diagrams and waveforms. Development of the concept of reactance, the study of series R-L, R-C, R-L-C circuit and resonance, study of parallel R-L, R-C and R-L-C circuit, concept of impedance, admittance, conductance and susceptance, the concept of active, reactive and apparent power and power factor,. Voltages, currents and power relations three-phase have balanced star-connected loads and delta-connected loads along with phasor diagrams.	30%	15

3	**UNIT-III: Diode and Transistors** Introduction to Ideal Diode, Effect of temperature Ideal diodes, unbiased diode and Forward and reverse bias of Diode. PIV, surge current, Diode as Uncontrolled switch. Rectifiers: Half wave, Full wave, and bridge wave. Ripple factor, PIV rating. Choke and Capacitor input filter rectifiers, Clipper and Clamper circuits, Voltage multiplier: Construction and working of BJT, Characteristics & specifications of BJT (PNP & NPN transistors), Biased and unbiased BJT, Configuration of the transistor, the concept of gain & BW, Operation of BJT in the cut-off, saturation & active regions (DC analysis), BJT as a switch, Transistor as an amplifier, Voltage divider bias and analysis, VDB load line and Q point.	30%	15
4	**UNIT-IV: Voltage Regulator** Lasers: Interaction of radiation with Matter, Absorption, Spontaneous and Stimulated emission, Characteristics of Lasers, Types of Lasers: Ruby Laser, Helium-Neon Laser, Semiconductor Diode Laser, Applications of Lasers. Fiber Optics: Principle and Structure of Optical Fiber, Numerical Aperture of fiber, Types of Optical Fibers, Attenuation in Optical Fibers, Applications of Optical Fibers.	15%	5
5	**UNIT-V: Sensors and Transducers** Introduction to sensors and Transducers, Comparison between sensors and Transducers, Applications of Sensors and Transducers, Types of Electronic sensors, Types of Transducers.	15%	5

i. Reference Books:

1. **A text book of Electrical technology Vol2** By B.L.Theraja — S. Chand Publication.
2. **Electrical Engineering Fundamentals (TextBook)** By V. D. Toro — Prentice Hall India — 2, Pub. Year 1989.
3. **Electrical and Electronics Technology** By E. Hughes — Pearson — 10, Pub. Year 2010.
4. **Basic Electrical Engineering** By D. P. Kothari and I. J. Nagrath, — Tata McGraw Hill — 3, Pub. Year 2010.
5. **Basic Electrical Engineering** By D. C. Kulshreshtha — McGraw Hill — 1, Pub. Year 2009.
6. **Fundamentals of Electrical Engineering** By Leonard S. Bobrow — Oxford University Press — 2, Pub. Year 1996.

j. List of Practicals:

1. To Study about Various Electrical and Electronics Symbols and demonstrate various measuring instruments used in Basic electrical Engineering laboratory.
2. To Perform and Solve Electrical Networks with Series and Parallel Combinations of Resistors Using Kirchhoff 's Laws.
3. To Obtain Inductance, Power and Power Factor of the Series RL Circuit With AC Supply Using Phasor Diagram.
4. To Obtain Capacitance, Power and Power Factor of the Series RC Circuit With AC Supply Using Phasor Diagram.
5. To Obtain Inductance, Capacitance, Power and Power Factor of the Series R-L-C Circuit With AC Supply Using Phasor Diagram.
6. Verification of superposition theorem with dc source.
7. Verification of Thevenin's theorem with dc source.
8. Verification of Norton's theorems in dc circuits.
9. Verification of Current and Voltage Relations in Three Phase Balanced Star and Delta Connected Loads.
10. To study the cut-section of a dc machine, single phase induction machine and three phase induction machine.
11. Find out the Efficiency and Voltage Regulation of Single Phase Transformer by Direct Load Test.
12. To Plot V-I characteristics Diodes. (a) PN junction diode Characteristics, (b) Zener Diode characteristics.
13. To Observe Rectifier Circuit (a) Half wave Rectifier without filter, (b) Full wave rectifier without filter, (c) Half wave
14. To Observe Response of Clipping and Clamping circuits using diodes (a) Diode Positive Clipper without and with Biased clipper, (b) Diode Negative Clipper without and with Biased clipper, (c) Biased Positive Negative Clipper (Combinational Clipper), and (d) Positive Clamper, and Negative Clamper.
15. Designing of power supply using IC regulator circuit. (a) Designing of +5 Volt DC Power Supply using 7805, (b) Designing of -5 Volt DC Power Supply using 7905, (c) Designing of +12 Volt DC Power Supply using 7812, and (d) Designing of -12 Volt DC Power Supply using 7912.
16. (a)To Plot and Study input-output characteristics of Common Base (B) configuration of the Transistor and (b) To Plot and Study input-output characteristics of common Emitter (CE) configuration of Transistor.
17. To study the Voltage divider bias circuit: (a) To observe the effect of change in base current on the Q-operating point, and (b) To set Q point for operation of a transistor amplifier in the linear region.
18. To plot characteristics of Schottky and Varactor diode.
19. Designing of Linear Adjustable Regulator using IC LM317.
20. Introduction to Sensors and Transducers.

Semester 2-1

a. **Course:** Programming for AI

b. **Course Code:**

c. **Prerequisite:** Linear Algebra, Probability & Statistics, Data Structures & Algorithms

d. **Rationale:** The course aims to develop advanced programming skills for data science and AI, focusing on Python, data manipulation, machine learning, and model deployment. Students will gain hands-on experience with libraries like NumPy, Pandas, scikit-learn, TensorFlow, and Git for real-world applications.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the fundamentals of web technologies, architecture, and web hosting principles.
CLOBJ 2	Apply Python programming concepts for data manipulation, preprocessing, and analysis using libraries such as NumPy and Pandas.
CLOBJ 3	Develop and evaluate machine learning models using scikit-learn, including techniques like classification, regression, and clustering.
CLOBJ 4	Utilize Git and collaborative tools for version control, code sharing, and software development best practices in AI projects.
CLOBJ 5	Deploy AI models using frameworks such as Flask or Django and understand model hosting on cloud platforms like AWS or Google Cloud.
CLOBJ 6	Analyze the ethical implications of AI, addressing concerns such as bias, fairness, privacy, and responsible AI design.

f. **Course Learning Outcome:**

CLO 1	Understand ethical and responsible AI development.
CLO 2	Handle and preprocess large datasets.
CLO 3	Develop AI-optimized code in Python
CLO 4	Apply deep learning techniques using TensorFlow/Keras.
CLO 5	Implement machine learning models using scikit-learn

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction to AI Programming:** Overview of AI and its applications, Python for AI: Basics and best practices, Setting up AI development environments (Jupyter, Colab, VS Code)	5%	4
2	**Data Handling & Preprocessing:** NumPy for numerical computing, Pandas for data manipulation, Data cleaning and handling missing values Feature engineering and dimensionality reduction	10%	5
3	**Machine Learning:** Supervised vs. Unsupervised Learning, Implementing ML models using scikit-learn, Model evaluation, overfitting, and hyperparameter tuning.	15%	7
4	**Deep Learning Frameworks:** Introduction to TensorFlow and Keras, Building and training neural networks, Implementing convolutional and recurrent neural networks, Activation functions, loss functions, and optimization techniques	15%	7
5	**Version Control and Collaboration:** Using Git for version control, Collaborative coding practices and code reviews	20%	8
6	**Deployment of Models::** Basics of deploying models using Flask or Django, understanding cloud platforms for model deployment, Deploying AI models on AWS, Google Cloud	20%	8
7	**Ethics & Responsible AI:** Bias and fairness in AI, Privacy, security, and explainability of AI models, ethical implications of AI applications, Understanding bias and fairness in AI models	15%	6

i. Reference Books:

1. **"Python Machine Learning"** By Sebastian Raschka & Vahid Mirjalili
2. **"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow"** By Aurélien Géron
3. **"Deep Learning with Python"** By François Chollet
4. **"Artificial Intelligence: A Modern Approach"** By Stuart Russell & Peter Norvig
5. **"Machine Learning Yearning"** By Andrew Ng

j. List of Practicals:

1. Implement object-oriented programming, exception handling, and file operations.
2. Perform data cleaning, transformation, and analysis using Pandas and NumPy
3. evaluate classification and regression models using scikit-learn.
4. Implement clustering techniques like K-Means and DBSCAN
5. Deploy a trained ML model using Flask or Django.
6. Implement Random Forest, Gradient Boosting, and Stacking models.
7. Implement feature scaling, encoding, and selection techniques.
8. Implement text classification using RNNs.
9. Perform image classification using a Convolutional Neural Network (CNN).
10. Build and train a simple deep learning model.

Semester 2 - 2

a. Course: Mathematics-II

b. Course Code: 303191151

c. Prerequisite: Knowledge of Mathematics up to 12th science level

d. Rationale: The Mathematics I syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and matrix algebra, preparing students for engineering challenges with optimized problem-solving skills.

e. Course Learning Objectives:

CLOBJ 1	Define and identify ordinary differential equations of higher order. Classify ODEs based on homogeneity and linearity. Solve homogeneous linear ODEs of higher order with constant coefficients, and variable coefficients.
CLOBJ 2	Solve homogeneous linear ODEs of higher order with constant coefficients, variable coefficients
CLOBJ 3	Apply the Method of Undetermined Coefficients to solve nonhomogeneous ODEs. Utilize the Solution by Variation of Parameters for solving nonhomogeneous ODEs. Explore applications of ODEs in real-world scenarios.
CLOBJ 4	Understand power series solutions for ordinary points and regular singular points. Explore properties and applications of Legendre polynomials and Bessel functions.
CLOBJ 5	Define Laplace transform and its inverse. Understand the linearity property of Laplace transforms. Solve ordinary differential equations using Laplace transforms.

f. Course Learning Outcomes:

CLO 1	Demonstrate the ability to translate physical or engineering problems into mathematical equations and solve them.
CLO 2	Develop analytical and critical thinking skills through the process of solving complex mathematical problems.
CLO 3	Understand and interpret mathematical solutions in the context of the given problems.
CLO 4	Communicate mathematical concepts and solutions clearly and effectively, both in written and verbal forms.
CLO 5	Present mathematical arguments and solutions in a logical and organized manner.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
4	-	-	4	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT 1: Higher order ordinary differential equations:** Ordinary differential equations of higher orders, Homogeneous Linear ODEs of Higher Order, Homogeneous Linear ODEs with Constant Coefficients, Euler–Cauchy equations, Nonhomogeneous ODEs, Method of Undetermined Coefficients, Solution by Variation of Parameters, Applications	8%	5
2	**UNIT 2 Power Series:** Power series solutions at ordinary point and regular singular point; Legendre polynomials, Bessel functions of the first kind and their property	15%	9
3	**UNIT 3 Laplace Transform:** UNIT 3 Laplace Transform: Laplace Transform and inverse Laplace transform, Linearity, First Shifting Theorem (s-Shifting), Transforms of Derivatives and Integrals, ODEs, UNIT Step Function (Heaviside Function), Second Shifting Theorem (t-Shifting), Laplace transform of periodic functions, Short Impulses, Dirac's Delta Function, Convolution, Integral Equations, Differentiation and Integration of Transforms, Solution of ordinary differential equation by Laplace transform	25%	15
4	**UNIT 4 Fourier Integral :** Fourier Integral, Fourier Cosine Integral and Fourier Sine Integral	17%	10
5	**UNIT 5 Vector Calculus:** Gradient of scalar field, Directional Derivative, Divergence and curl of Vector field, Scalar line integrals, vector line integrals, scalar surface integrals, vector surface integrals, Theorems of Green, Gauss and Stokes.	10%	6

6	**UNIT 6 Multivariable Calculus (Integration):** Multiple Integration: Double integrals (Cartesian), change of order of integration in double integrals, Change of variables (Cartesian to polar), Triple integrals (Cartesian)	25%	15

i. Reference Books:

1. **Advanced Engineering Mathematics (TextBook)** By Erwin Kreyszig — Willey India Education
2. **Calculus with early transcendental functions** By James Stewart — Cengage Learning
3. **Higher Engineering Mathematics** By B. S. Grewal — Khanna Publications
4. **Calculus and Analytic Geometry (TextBook)** By G.B. Thomas and R.L. Finney — Addison Wesley A text book of Engineering Mathematics By N.P. Bali and Manish Goyal — Laxmi Publications

Semester 2 - 3

a. Course: Computational Thinking for Structured Design-2

b. Course Code: 303105151

c. Prerequisite: A foundational understanding of logic and problem-solving is a prerequisite for computational thinking in structured design.

d. Rationale: Computational thinking is integral for structured design as it fosters a systematic approach to problem-solving, breaking down complex issues into manageable components. By applying computational thinking principles, individuals can create well-organized and efficient structured designs, promoting clarity, maintainability, and scalability in software development. This methodology aligns with the logical and stepby-step nature of structured design, enhancing the overall effectiveness of the development process.

e. Course Learning Objectives:

CLOBJ 1	Develop a deep understanding of foundational computational thinking concepts and their application in problem-solving.
CLOBJ 2	Demonstrate proficiency in creating structured designs using appropriate programming constructs and methodologies.
CLOBJ 3	Apply algorithmic thinking to decompose complex problems into manageable components, enhancing systematic problem-solving abilities.
CLOBJ 4	Evaluate and refine structured designs through critical analysis, promoting clarity, efficiency, and scalability in software solutions.

f. Course Learning Outcomes:

CLO 1	Develop proficiency in breaking down complex problems into manageable components, demonstrating a mastery of foundational computational thinking concepts.
CLO 2	Apply structured design principles to create efficient and well-organized algorithms, fostering a systematic approach to problem-solving in various domains.
CLO 3	Demonstrate the ability to design and implement structured programs using appropriate programming languages, showcasing practical skills in translating algorithms into executable code.
CLO 4	Cultivate a problem-solving mindset, emphasizing analytical thinking, algorithmic reasoning, and code optimization for developing scalable and maintainable software solutions.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT-I Dynamic Memory Allocation:** malloc, calloc, realloc and free, Array of pointers, Programing Applications, Dangling Pointer	10%	6
2	**UNIT-II Preprocessor Directives:** File Inclusion, Macros, Conditional Compilation and Pragmas.	10%	6
3	**UNIT-III: Enumerators, Structures, Unions:** **Enumerators:** Enumerator Types **Structures:** Declaration Initialization Accessing Structures, Complex Structures, Structure and Functions Array of structures Arrays within structures Anonymous structures Nested structures pointers in structures Self-referential structures Structure Padding **Unions:** Bit fields **Typedef**	15%	15
4	**UNIT-IV: Searching and Sorting:** Selection sort, Bubble Sort, ,Insertion sort, Quick sort and Merge Sort Linear and Binary Searching Techniques	30%	3
5	**UNIT-V: Data Structures: List- Linear List: :** Singly Linked List - CRUD operations Double Linked List -CRUD operations Circular Linked List- CRUD operations	35%	15

i. Text Books:

1. **Fundamentals of Data Structures in C, 2ND eDITION** By E.Horowitz, S,.Sahni and Susan Anderson- Freed, Universities Press (TextBook)
2. **Computer Programming & Data Structures** By E. Balaguruswamy,4th Edition TMH
3. **C & Data Structures** By P . Padmanabham,Third Edition,B.S Publications
4. **Classic Data Structures** By D.samanta

j. List of Practicals:

1. Write a c program to increase or decrease the existing size of an 1D array. 2. Write a c program on 2D array to Increase & Decrease i) No of subarrays ii) elements in the subarrays.
2. Write a to display present date and time using c language. 2. Write a c program to demonstrate pre-processor directives i) Macros ii) Conditional Compilation.
3. Write a C program that uses functions to perform the following Operations. i) Reading a complex number ii) Writing a complex number iii) Addition of two complex numbers iv) Multiplication of two complex numbers 2. Write a c program to store records of n students based on roll_no, name, gender and 5 subject marks i) Calculate percentage each student using 5 subjects. ii) Display the student list according to their percentages.
4. Write a C program to store n employee records based on EMP_ID,EMP_NAME,EMP_DEPTID,EMP_PHNO,EMP_SALARY and display all the details of employees using EMP_NAME in sorted order.
5. Write a c program to implement selection Sort & Bubble sort 2. Write a C program to reverse the elements within a given range in a sorted list. Example : input : 10 9 1 2 4 3 4 6 7 8 10 3 8 output: 1 2 8 7 6 4 4 3 9 10 the sorted list of given array elements is 1 2 3 4 4 6 7 8 9 10 , after reversing the elements with in the range 3 and 8 is 1 2 8 7 6 4 4 3 9 10.
6. Write a c program to implement Insertion sort & Quick sort
7. Write a c program to sort the given n integers and perform following operations i) Find the products of every two odd position elements ii) Find the sum of every two even position elements Explanation: Input : 9 1 9 8 3 5 4 7 2 6 Output: 3 15 35 63 6 10 14 The sorted list of given input is 1 2 3 4 5 6 7 8 9, the product of alternative odd position elements is 1*3 = 3,3*5=15,5*7=35... and the sum of two even position elements 2+4 =6,4+6=10.
8. Write a C Program to implement Merge Sort.
9. Write a c program to sort in ascending order and reverse the individual row elements of an mxn matrix.
10. Write a c program to perform linear Search. 2. Write a c program to perform binary search.
11. Write a c program to Create a single Linked list and perform Following Operations A. Insertion At Beginning B. Insertion At End C. Insertion After a particular node D. Insertion Before a particular node E. Insertion at specific position F. Search a particular node G. Return a particular node H. Deletion at the beginning I. Deletion at the end J. Deletion after a particular node K. Deletion before a particular node L. Delete a particular node M. Deletion at a specific position.
12. Write a program to Reverse a singly Linked list. 2. Write a c program to check whether the created linked list is palindrome or not.
13. Write a c program to Create a Circular Linked list and perform Following Operations A. Insertion At Beginning B. Insertion At End C. Insertion After

a particular node. D. Insertion Before a particular node E. Insertion at specific position F. Search a particular node G. Return a particular node H. Deletion at the beginning I. Deletion at the end J. Deletion after a particular node K. Deletion before a particular node L. Delete a particular node M. Deletion at a specific position

14. Write a c program to Create a Circular single Linked list and perform Following Operations A. Insertion After a particular node B. Insertion Before a particular node C. Search a particular node D. Return a particular node E. Deletion before a particular node F. Delete a particular node.

15. Write a c program to Create a Circular Double Linked list and perform Following Operations A. Insertion After a particular node B. Insertion Before a particular node C. Search a particular node D. Return a particular node E. Deletion before a particular node F. Delete a particular node.

Semester 2 - 4

a. Course: Engineering Physics.

b. Course Code: 303192102

c. Prerequisite: Knowledge of Physics and some basic concepts in Mathematics like differentiation, integration, limit, differential equation, vector calculus up to 12th science level.

d. Rationale: Knowledge of physics is essential for all Engineering branches because physics is the foundation subject of all the branches of engineering and it develops the scientific temperament and analytical capability of engineering students.

e. Course Learning Objectives:

CLOBJ 1	Understand the basics of quantum mechanics, including Schrödinger's equations and the physical significance of wave functions.
CLOBJ 2	Apply the Schrödinger equation to analyze particles in one-dimensional potential boxes, emphasizing practical implications and tunneling effects.
CLOBJ 3	Master concepts of energy bands, semiconductor classification, E-k diagrams, and semiconductor device analysis including P-N junction diodes.
CLOBJ 4	Comprehensively understand material classification, focusing on magnetic materials, nanomaterials, and analyzing physical, thermal, electrical, optical, and magnetic properties.
CLOBJ 5	Gain expertise in laser principles, types, and applications, as well as fiber optics principles and applications. Understand optoelectronic devices, their functionalities, and practical applications.

f. Course Learning Outcomes:

CLO 1	Understand the basics of quantum mechanics, including Schrödinger's equations and the physical significance of wave functions.
CLO 2	Apply the Schrödinger equation to analyze particles in one-dimensional potential boxes, emphasizing practical implications and tunneling effects.
CLO 3	Master concepts of energy bands, semiconductor classification, E-k diagrams, and semiconductor device analysis including P-N junction diodes.
CLO 4	Comprehensively understand material classification, focusing on magnetic materials, nanomaterials, and analyzing physical, thermal, electrical, optical, and magnetic properties.
CLO 5	Gain expertise in laser principles, types, and applications, as well as fiber optics principles and applications. Understand optoelectronic devices, their functionalities, and practical applications.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Topics**	**Weightage**	**Teaching Hours**
1	**UNIT-I: Modern Physics** Introduction about quantum Mechanics, Schrödinger's equations, Time dependent and Time Independent Wave Equation, Physical Significance of the wave Function, Application of Schrödinger equation in particles in One Dimensional Potential Box and Tunneling effects.	20%	9

2	**UNIT-II: Band Theory of Semiconductors** Energy bands in solids, Classification of Materials into Semiconductors & Insulators, Density of state, E-k diagram, Kronig-Penny model (to introduce origin of band gap), and Effective mass. Direct and indirect band gap. Carrier Concentration in semiconductors, Fermi Level in Intrinsic and Extrinsic Semiconductors, P-N junction diode, Ohmic and Schottky Junction.	20%	9
3	**UNIT-III: Materials** Classification of materials: Magnetic materials, Nanomaterials based on semiconductors and metal oxides, Basic characteristic properties of nanomaterials, Novel Materials. Physical, Thermal, Electrical, Optical and Magnetic properties of materials.	20%	9
4	**UNIT-IV: Laser and Fiber Optics** Lasers: Interaction of radiation with Matter, Absorption, Spontaneous and Stimulated emission, Characteristics of Lasers, Types of Lasers: Ruby Laser, Helium-Neon Laser, Semiconductor Diode Laser, Applications of Lasers. Fiber Optics: Principle and Structure of Optical Fiber, Numerical Aperture of fiber, Types of Optical Fibers, Attenuation in Optical Fibers, Applications of Optical Fibers.	20%	9
5	**UNIT-V: Devices** Optoelectronic Devices: Photoconductive cell, photovoltaic cell, Photodiode, Phototransistor, LED, IR emitters, Opto coupler, X-ray diffractometer, Quantum devices and their applications.	20%	9

i. Reference Books:

1. **Semiconductor Optoelectronics: Physics and Technology**By J. Singh, McGraw-Hill Inc. (1995)
2. **Fundamentals of Photonics** By B. E. A. Saleh and M. C. Teich, John Wiley & Sons, Inc., (2007)
3. **Semiconductor Devices: Physics and Technology** By S. M. Sze, Wiley (2008)
4. **Engineering Physics** By HK Malek and A. K. Singh- McGraw Hill Publication
5. **Semiconductor Optoelectronic Devices** By P. Bhattacharya-Prentice Hall of India
6. **Fundamentals of Physics** By Halliday, Resnick and Walker

j. List of Practicals:

1. I-V characteristics of light emitting diode in forward bias.
2. I-V characteristics of Zener diode in reverse bias.

3. Determination of Velocity of ultrasonic waves in water.
4. Determination of Dielectric constants of Dielectric samples.
5. Measurement of Band gap of semiconductor material.
6. Measurement of Hall coefficient RH and carrier concentration in a semiconductor.
7. Measurement of Planck's constant using LED.
8. Measurement of wavelength of laser light using diffraction grating.
9. Measurement of Numerical aperture of an optical Fiber.
10. Moment of Inertia of a flywheel.
11. Measurement of power loss in an optical fibre.
12. B-H Curve tracing.
13. Determination of Young's modulus.
14. Determination of thermal conductivity. (Searle's method or Lee's method)

Semester 2 - 5

a. Course: Environmental Science

b. Course Code: 303104105

c. Prerequisite: Knowledge of Physics, Chemistry and Mathematics up to 12th science level and Biology up to 10th science level

d. Rationale: Basic knowledge of the environment is essential for all human beings for a good life and sustainable existence.

e. Course Learning Objectives:

CLOBJ 1	Apply systems thinking to analyze the city as a system, demonstrating application.
CLOBJ 2	Evaluate the role of smart citizens and approaches for citizen engagement.
CLOBJ 3	Identify sources and stressors of water resources, demonstrating understanding.
CLOBJ 4	Analyze the causes, effects, and control measures of population explosion.

f. Course Learning Outcomes:

CLO 1	Understand the interrelation and interdependency of organisms and their interactions with the environment.
CLO 2	Identify eco-friendly measures in engineering projects.
CLO 3	Understand preventive steps for environmental protection.
CLO 4	Act as a responsible individual who is aware of efficient usage of resources and securing sustainable development.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
1	0	0	Audit	-	50	-	-	-	50

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT-I: ENVIRONMENTAL HEALTH, ECOLOGY AND QUALITY OF LIFE** Environmental education: Objective and scope, Impact of technology on the environment, Environmental disasters: Case studies, Global environmental awareness to mitigate stress on the environment, Structure and function of an ecosystem, Ecological pyramids, Pyramid of number, Pyramid of energy and pyramid of biomass.	25%	7
2	**UNIT-II: POLLUTION PREVENTION** Air & Noise pollution - Sources & their Effects, Case studies of Major Catastrophes, Structure and composition of the atmosphere, Water, Soil, Marine, Thermal & Marine Pollution: The story of fluoride contamination, Eutrophication of lakes, control measures, Measuring water quality: Water quality index, Waste water treatment (general) primary, secondary and tertiary stages, Municipal Solid waste management: Sources and effects of municipal waste, Biomedical waste, Hazardous waste.	20%	6
3	**UNIT-III: POPULATION GROWTH, GLOBAL ENVIRONMENTAL CHALLENGES & LATEST DEVELOPMENTS** Population Explosion - Causes, Effects and Control, an International initiative in population-related issues, Urbanization, Growth of the world's large cities, Water resources: Sources of water, Stress on water resources, Climate Change, Global Warming and Green House Effect, Acid Rain, Depletion of Ozone layer, Variation in concentrations of GHG gases in ambient air during last millennium, Role of Environmental Information System (ENVIS) in India and similar programs run by EPA(USA), Role of soft tools like Quantum GIS, Autodesk Building Information Modeling (BIM) and City Finance Approach to Climate-Stabilizing Targets (C- FACT), Life Cycle Assessment, Bioinformatics and Optimization tools for sustainable development.	25%	7

4	**UNIT-IV: SMART CITIES** Introduction to smart cities - about smart cities, what is a smart city, world urbanization, case studies of Songdo, Rio De Janeiro, what makes cities smart. City as a system of systems – Introduction, systems thinking, Milton Keynes Future Challenges, Rich picture as city challenges, Wicked problems, Development of smart city approach – core elements, open data, sustainability, privacy and ethics, development processes. Smart Citizens – their role, engaging citizens, IES Cities, Energy systems, Approaches for Citizen Engagement, co-creating smart cities, cities unlocked, living labs, city problems, crowdsourcing ideas, redesigning cities for citizens, all age-friendly cities, mobility on demand, motion maps, Infrastructure, Technology and Data – urban infrastructure and its technology, future of lighting, IoT, connected objects, sensing the city, NOx eating paints and air quality sensors, safest, smart citizen kit, sensing your city, Sensored City, Cyber security for data power, open, shared and closed data, satellite data, open data revolution, Smart City Project Data. Innovation – smart innovations, smart city ecosystem, data-driven innovations for smart cities. Standards and Capacity Building – the role of Standard, BSI smart city Standards, HyperCat, ITU Smart Sustainable cities, Smart City Readiness, Lessons Learnt from Amsterdam. Smart Measurements - metrics and indicators, city indicators, WCCD data portal, value proposition, integrated reporting, smart city learning and education, urban data school.	30%	10

i. Reference Books:

1. **Textbook of Environmental Studies For Undergraduate Courses (Text Book)** By Dr Erach Bharucha — Orient BlackSwan — Second Edition, Pub. Year 2013.
2. **Basics of Environmental Studies** By U K Khare — Tata McGraw Hill.
3. **Environmental Studies** By Anindita Basak — Drling Kindersley(India)Pvt. Ltd Pearson.
4. **Environmental Sciences** By Daniel B Botkin & Edward A Keller — John Wiley & Sons.
5. **Air Pollution** By M N Rao , H .V N Rao — McGraw Hill Publishing Company Limited, New Delhi.

Semester 3 - 1

a. Course: Design of Data Structure

b. Course Code: 303105201

c. Prerequisite: Computer Programming and Basic Syntaxes

d. Rationale: Data structure is a subject of primary importance in Information and Communication Technology. Organizing or structuring data is important for implementation of efficient algorithms and program development. Efficient problem solving needs the application of appropriate data structure during program development.

e. Course Learning Objectives:

CLOBJ 1	Gain familiarity with Principles of OSS, Open-Source Standards, Requirements for Software, OSS success, Free Software, Examples, Licensing, Free Vs. Proprietary Software, Free Software Vs. Open-Source Software, Public Domain.
CLOBJ 2	Acquire Knowledge regarding Open-Source History, Open Source Initiatives, Open Standards Principles, Methodologies, Philosophy, Software freedom, Open-Source Software Development, Licenses, Copyright vs. Copy left, Patents, Zero marginal cost, Income-generation Opportunities, Internationalization
CLOBJ 3	Acquire knowledge of Community and Communication, Contributing to Opensource Projects Introduction to GitHub, interacting with the community on GitHub, Communication and etiquette, testing open-source code, reporting issues, contributing code. Introduction to Wikipedia, contributing to Wikipedia or contributing to any prominent open-source project of student's choice. Open-Source Ethics and Social Impact: Open source vs. closed source, Open-source Government, Ethics of Opensource,
CLOBJ 4	Understand GNU/Linux, Android, Free BSD, Open Solaris. Open-Source Hardware, Virtualization Technologies, Containerization Technologies: Docker, Development tools, IDEs, Debuggers, Programming languages, LAMP, Open-Source Database technologies
CLOBJ 5	Demonstrate apache Web server, BSD, GNU/Linux, Android, Mozilla (Firefox), Wikipedia, Drupal, WordPress, Git, GCC, GDB, GitHub, Open Office, LibreOffice Study

f. Course Learning Outcomes:

CLO 1	Use different types of data structures, operations and algorithms.
CLO 2	Apply searching and sorting operations on files
CLO 3	Use stack, Queue, Lists, Trees and Graphs in problem solving.
CLO 4	Implement all data structures in a high-level language for problem solving.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	4	5	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Topics**	**Weightage**	**Teaching Hours**
1	**Introduction:** Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, dynamically allocated arrays. Performance analysis of an algorithm and space and time complexities	10%	6
2	**Stacks, Recursion and Queue:** Stacks: Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. Recursion - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Queues: Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Deque, Priority Queues and its problems	15%	8

3	**Linked Lists:** Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists	10%	5
4	**Searching and Sorting:** Interpolation Search Sorts: Selection Sort, Insertion Sort, Bubble Sort, Quick Sort, Merge Sort, Radix Sort	10%	5
5	**Trees:** Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - In Order, Post Order, Pre Order; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression.	10%	4
6	**Red Black Trees and AVL Trees:** Introduction-Operations on Red Black Trees, AVL tree Construction, Operations on AVL Trees	15%	8
7	**Hashing:** Hash Table organizations, Hashing Functions, Static and Dynamic Hashing	15%	3
8	**Graphs:** Definitions, Terminologies, Matrix and Adjacency List Representation of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search.	15%	5

i. Reference Books:

1. **Fundamentals of Data Structures in C, 2ND EDITION** By E.Horowitz, S,.Sahni and Susan Anderson- Freed, Universities Press (TextBook)
2. **Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed** By McGraw Hill, 2014.

j. List of Practicals:

1. Implement Stack and its operations like (creation push pop traverse peek search) using linear data structure
2. Implement Infix to Postfix Expression Conversion using Stack.
3. Implement Postfix evaluation using Stack.
4. Implement Towers of Hanoi using Stack.
5. Implement queue and its operations like enqueue, dequeue, traverse, search.
6. Implement Single Linked lists and its operations(creation insertion deletion traversal search reverse).
7. Implement Double Linked lists and its operations(creation insertion deletion traversal search reverse).

8. Implement binary search and interpolation search.
9. Implement Bubble sort, selection sort, Insertion sort, quick sort ,merge sort.
10. Implement Binary search Tree and its operations (creation, insertion, deletion).
11. Implement Traversals Preorder In-order Post-order on BST.
12. implement Graphs and represent using adjacency list and adjacency matrix and implement basic operations with traversals (BFS and DFS).

Semester 3 - 2

a. **Course:** Database Management System

b. **Course Code:** 303105203

c. **Prerequisite:** Basic Computer Knowledge

d. **Rationale:** The course will enable students to understand the different issues involved in the design and implementation of a database system as well execute various database queries using SQL.

e. **Course Learning Objectives:**

CLOBJ 1	Understand DBMS and FPS
CLOBJ 2	Study the use of DBMS language, SQL
CLOBJ 3	Acquire knowledge of the different types of Model and E-R Diagram.
CLOBJ 4	Understand different Data Models, Constraints and keys, and relational algebra Model
CLOBJ 5	Study the use of transaction, database recovery, concurrency control and deadlock
CLOBJ 6	Study the use of Query Processing
CLOBJ 7	Understand different relational database designs
CLOBJ 8	Acquire knowledge of the security
CLOBJ 9	Understand the PL/SQL practical

f. **Course Learning Outcomes:**

CLO 1	Understand basic concepts of Database
CLO 2	Understand Relational Models and their importance.
CLO 3	Build a properly structured database for a given problem or application.
CLO 4	Learn how various transactions are managed in real-time scenarios.
CLO 5	Understand the evaluation parameters of a query as well as the security parameters of the database.
CLO 6	Implement SQL concepts to build dynamic database applications.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**Introduction:** Introduction and applications of DBMS, File Processing System and its limitations, ANSI/SPARC Model, Data Independence, Client-Server Architecture, Users & DBA, Database Architecture.	10%	3
2	**SQL:** Data Definition Language (DDL) commands, Data Manipulation Language (DML) commands, Data Control Language (DCL) commands, Transaction Control Language (TCL) commands. Predicates & Clauses: Logical Operators (AND / OR), Relational Operators, BETWEEN Predicate, IN & NOT IN Predicate, LIKE Predicate. Functions in SQL: Aggregate Functions, Character Functions, Arithmetic Functions, Date Functions, Conversion Functions.	10%	4
3	**Data Models:** Hierarchical Model, Network Model, Relational Model, Object-Oriented Model. **E-R Diagram:** Introduction to E-R Diagram, Entities, Attributes & its types, Relationships, Mapping Cardinalities, Participation Constraints, Weak Entity Sets, Specialization, Generalization, Aggregation.	10%	5
4	**Relational Data Model:** Introduction, Degree, Cardinality. **Constraints & Keys:** Primary Key, Foreign Key, Super Key, Candidate Key, Not Null Constraint, Check Constraint. **Relational Algebra Operations:** Selection, Projection, Cross-Product, Rename, Joins (Natural & Outer Join), Set Operators (Union, Intersection, Set Difference), Aggregate Functions.	10%	4

5	**Relational Database Design:** Functional Dependency – definition, trivial and non-trivial FD, Armstrong's Axioms/Inference Rules, Closure of FD, Closure of Attributes, Candidate Key, Finding a Candidate Key, Decomposition (Lossy & Lossless), Database Anomalies, Normalization – 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.	20%	6
6	**Transaction:** Introduction, ACID Properties, Transaction Life Cycle, Scheduling, Serial Schedule, Interleaved Schedule, Transaction Operations, Serializability (View & Conflict), Two-Phase Commit Protocol. **Database Recovery:** Introduction, Log Based Recovery, Shadow Paging, Checkpoints. **Concurrency Control:** Introduction, Lock Based Protocol, Two Phase Lock Protocol, Intention Locking, Multiple Granularity, Time-based Protocol. **Deadlock:** Introduction, Deadlock Detection, Deadlock Recovery, Deadlock Prevention (Wait-Die, Wound-Wait & Timeout-Based Approach).	20%	12
7	**Query Processing:** Introduction, Layers of Query Processing, Measures of Query Cost, File Scans (Linear & Binary Search), Materialized View, Pipelining. **Query Optimization:** Introduction, Equivalence Rules, Cost-Based Query Optimization.	10%	3
8	**Security:** Data Security, Data Integrity, Authentication, Authorization, Encryption, Decryption, Access Control (DAC, RBAC, MAC), Intrusion Detection, SQL Injection.	5%	2
9	**PL/SQL Concepts:** Views, PL/SQL Block, Cursors, Triggers, Stored Procedures, Stored Functions.	5%	3

i. Reference Books:

1. **Database System Concepts** By Abraham Silberschatz, Henry Korth, S. Sudarshan — McGraw Hill International — 6th Edition
2. **An Introduction to Database Systems** By C. J. Date, A. Kannan, S. Swamynathan — Pearson Education

j. Practical List:

1. Create the following:
   i. **DBMS Concepts:**
      - Define DBMS (Database Management System). Explain the advantages of DBMS over File Processing System (FPS).
      - List 15 applications of Database. Explain any 2 applications and describe how a Database can be helpful in managing those applications.

   ii. **Database Creation:**
      - Create a database with the following details:

      A. **Student Details:** Create using Excel.

B. **Employee Details:** Create using MS Access.

C. **Facebook:** Create using Excel.

2. Create the following Tables:

- Important Instructions:
  - Use `varchar2(30)` datatype for Alphanumeric Characters and Special Symbols, `number` datatype for Numbers, `date` datatype for Date.
  - Use same table and column name (Capital and Small Case) as mentioned in this file.
  - Insert proper data (Capital and Small Case) as mentioned in this file.
- Employee
  - `Emp_name Street City`
  - Adam `Spring` Pittsfield
  - Brooks `Senator` Brooklyn
  - Curry `North` Rye
  - Demalo `SunShine` San Deago

3. Simple Queries:

i. Describe `deposit`, `branch`.

ii. Describe `borrow`, `customers`.

iii. List all data from table `DEPOSIT`.

iv. List all data from table `BORROW`.

v. List all data from table `CUSTOMERS`.

vi. List all data from table `BRANCH`.

vii. Give account no and amount of depositors.

viii. List all data from `SAILORS`.

ix. List `Boat Name` and its `color`.

x. List `Employee name` and its `city`.

xi. List all the details of `Clients`.

xii. Describe various `products` and its `price`.

xiii. Describe `sailor's name`, `age` and its `rating`.

xiv. Describe the `managers` of various `employees`.

xv. Describe the details of `Loan` for `customers`.

xvi. Describe the `date` of travel of various `sailors`.

4. Simple Queries:

i. Give name of depositors having amount greater than 4000.

ii. List the employees having salary less than 22000.

iii. List the sailors having age more than 25.

iv. List the boats travelling on 10-Oct-98.

v. List the details of boat "Interlake".

vi. List the details of the red colored boat.

vii. List the details of clients whose city is Mumbai.

viii. List Client Name, due balance and city of the clients having balance greater than 1500.

ix. Describe the details of products having selling price less than 500.
x. List the products for which quantity ordered is less than 120 and cost price is greater than 250.
xi. Display account details having amount greater 2200.
xii. Display all the customers staying in Nagpur.
xiii. Display the names of sailors having rating greater than 7.
xiv. Display the orders made in the month of June.
xv. List all the accounts created in the month of March.

5. "Like" Queries:

i. Display all customers whose name start with 'M'.
ii. Display all the customers whose name ends with 'L'.
iii. Display all loan details whose branch starts with 'A'.
iv. Display the details of sailors whose name is minimum 6 characters long.
v. Display the details of Employees whose address starts with 'S'.
vi. List the details of the boat ending with 'e'.
vii. List the details of clients having 'h' as a 3rd character in his/her name.
viii. List Client Name, due balance and city whose pin code starts with 4.
ix. List all customers whose city contains 'a' as second character.
x. List client names and city whose state has 'a' as fourth or fifth character.

6. "Aggregate Functions & DML" Queries:

i. List total deposit from `deposit`.
ii. Give Maximum loan given to a customer.
iii. Describe the average age of all the sailors.
iv. Count total number of customers.
v. Count total number of customer's cities.
vi. Display total target for the salesman.
vii. Update the salary of the employee having 10000 to 11500.
viii. Update the city of client from Bangalore to Bengaluru.
ix. Give the 15% hike in the salary of all the Employees. Rename that column to "New Salary".
x. Increase the sell price of all products by 20% and label new column as "New Sell Price". (Do not update the table)
xi. Provide the count of customers staying in "Bombay".

7. "Join" Queries:

i. Find the salary of Adam.
ii. Find the city where Brooks work.
iii. Display the sailor's details whose boat is booked for 9th May, 98.
iv. Display the day of ride and sailor name for boat 103.
v. Display the sailor name and its age for Red colored and 101 boat.
vi. Display the sailor details whose boat is never booked.
vii. Display the sailor name that has Red or Green Boat.

viii. Display all sailor details and boat details and who has Interlake boat.

ix. Display sailor's rating with boat details or the trip on 10th October, 98.

x. Display the sailor id and name whose age is more than 42 or who has Blue colored boat.

xi. Display name and rating of sailor whose boat name is Clipper.

xii. List products whose selling price is more than 500 and less than equal to 750.

xiii. Describe the second highest salary of an employee.

xiv. Display the date of travel and sailor's name whose age is between 35 and 65.

xv. List all the employees working for "FBC".

8. "Join" Queries:

i. Display all the employee name and the city where they work.

ii. Display the employee name and company's name having salary more than 15000.

iii. Find the average rating and age of all sailors.

iv. List various products available.

v. Display the names of salesman who have salary more than 2850.

vi. Change the cost price of Trousers to 950.

vii. List all the clients having "a" as a second character in their names.

viii. List all the products whose `QtyonHand` is less than `Reorderlvl`.

ix. Print the description and total `qty` sold for each product.

x. Find out all the products which have been sold to "Ivan Bayross".

xi. Find the names of all clients who have purchased Trousers.

xii. Find the products and their quantities for the orders placed by client `C00001` and `C00002`.

xiii. List the client details who place order no. `O19001`.

xiv. List the name of clients who have placed orders worth Rs. 10000 or more.

xv. Find the total of `Qty` ordered for each Order.

9. "Miscellaneous" Queries:

i. Find the average rate for each Order.

ii. Give the loan details of all the customers.

iii. List the customer name having loan account in the same branch city they live in.

iv. Provide the loan details of all the customers who have opened their accounts after August'95.

v. List the order information for client `C00001` and `C00002`.

vi. List all the information for the order placed in the month of June.

vii. List the details of clients who do not stay in Maharashtra.

viii. Determine the maximum and minimum product price. Rename the output as "Max_Price" and "Min_Price".

ix. Count the number of products having price less than or equal to 500.

x. List the order number and the day on which client placed an order.

xi. List the month and the date on which an order is to be delivered.

xii. List the date, 25 days after today's date.

xiii. Find the total of all the billed orders in the month of June.

xiv. List the products and orders from customers who have ordered less than 5 units of "Pull Overs".

xv. Find the list of products and orders placed by "Ivan Bayrosss" and "Mamta Muzumdar".

xvi. List the clients who placed order before June'04.

xvii. List all the clients who stays in "Bengaluru" or "Mangalore".

10. PL/SQL Block:

i. Write a PL/SQL Block to Add 2 Numbers.

ii. Write a PL/SQL Block to find Area of Rectangle, Triangle, and Square.

iii. Write a PL/SQL Block to find Maximum of 3 numbers.

iv. Write a PL/SQL Block to print sum of N Numbers using For Loop.

v. Write a PL/SQL Block to generate Fibonacci series of N numbers.

Semester 3 - 3

a. **Course:** Object Oriented Programming with JAVA

b. **Course Code:** 303105205

c. **Prerequisite:** Basic knowledge of software applications

d. **Rationale:** This course provides a broad introduction to software engineering. The various process models required to develop software are also described. Moreover, the functional and non-functional requirements are also described.

e. **Course Learning Objectives:**

CLOBJ 1	Gain the Knowledge of the concept with the Object-oriented programming, OOPs principles.
CLOBJ 2	Understand Data types, variables, operators.
CLOBJ 3	Understand the concept of Control statements.
CLOBJ 4	Demonstrate the use of Arrays, Array values, and memory storage Structure.
CLOBJ 5	Demonstrate the use of various OOPs concepts with the help of programs.
CLOBJ 6	Study the use of Inheritance with Examples.
CLOBJ 7	Understand the concept of Strings, Packages, and Interfaces.
CLOBJ 8	Demonstrate the Concept of Exception Handling.
CLOBJ 9	Gain the knowledge of multi-threading.
CLOBJ 10	Understand the knowledge of Collections Framework.

f. Course Learning Outcomes:

CLO 1	Describe the procedural and object-oriented paradigm with concepts of streams, classes, functions, data, and objects.
CLO 2	Understand dynamic memory management techniques using pointers, constructors, destructors, etc.
CLO 3	Describe the concept of function overloading, operator overloading, virtual functions, and polymorphism.
CLO 4	Classify inheritance with the understanding of early and late binding, usage of exception handling, and generic programming.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
2	0	2	3	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Topics**	**Weightage**	**Teaching Hours**
1	**Design introduction:** Object-oriented programming, OOPs principles, encapsulation, inheritance, and polymorphism; Java as an OOP & internet enabled language, importance of Java, Java usage in industry, the byte code, compiling, and running of simple Java program, JVM, JDK, JRE.	10%	3
2	**Data types, variable, operators:** Data types, variables, dynamic initialization, scope and lifetime of variables, type conversion and casting, operators.	10%	3
3	**Control statements:** Conditional Statements, Looping Statements, Jump Statements.	10%	3
4	**Arrays:** Array, Array values and memory storage Structure, Types of Arrays.	8%	3

5	**Object-oriented programming:** Classes and objects: concepts of classes and objects, declaring objects, assigning object reference variables, methods, constructors, access control, garbage collection, usage of static with data and methods, usage of final with data, overloading methods and constructors, parameter passing - call by value, recursion, nested classes.	18%	5
6	**Inheritance:** Inheritance Basics, member access rules, Usage of super key word, forms of inheritance, Method Overriding, Abstract classes, Dynamic method dispatch, Using final with inheritance.	8%	2
7	**Strings, Packages and Interfaces:** String handling functions, Packages, Class path, importing packages, differences between classes and interfaces, Implementing & Applying interface, enumerations in Java.	12%	4
8	**Exception Handling:** Exceptions, Types of Exceptions, Handling of Exceptions.	8%	2
9	**Multi Threading:** Thread, Usage of threads, Types of threads, Handling Threads.	10%	3
10	**Collections Framework:** Functional Programming, Collections, Hierarchy of collections.	6%	2

i. Reference Books:

1. **Introduction to Java Programming (Comprehensive Version)** By Daniel Liang; Pearson (TextBook)
2. **Core Java Volume-II Fundamentals** By Horstmann & Cornell; Pearson
3. **Complete Reference Java 2** By Herbert Schildt; TMH

j. List of Practicals:

1. Write a program to display Hello World message in the console window.
2. Write a program to perform arithmetic and bitwise operations in a single source program without object creation.
3. Write a program to perform arithmetic and bitwise operations by creating individual methods and classes, then create an object to execute the individual methods of each operation.
4. Write a Java program to display the employee details using Scanner class.
5. Write a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.
6. The Fibonacci sequence is defined by the following rule: The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence.

7. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that integer.
8. Write a Java program to multiply two given matrices.
9. Write a Java program for sorting a given list of names in ascending order.
10. Write a Java program for Method overloading and Constructor overloading.
11. Write a Java program to represent Abstract class with an example.
12. Write a program to implement multiple Inheritances.
13. Write a program to demonstrate method overriding and super keyword.
14. Write a Java program to implement Interface using extends keyword.
15. Write a Java program to create inner classes.
16. Write a Java program to create a user-defined package.
17. Write a Java program that displays the number of characters, lines, and words in a text.
18. Write a Java program that checks whether a given string is a palindrome or not. E.g., MADAM is a palindrome.
19. Write a Java program that reads a line of integers and then displays each integer and the sum of all integers. (Use StringTokenizer class).
20. Write a Java program for creating a single try block with multiple catch blocks.
21. Write a program for multiple try blocks and multiple catch blocks including finally.
22. Write a program to create a user-defined exception.
23. Write a Java program for producer and consumer problem using Threads.
24. Write a Java program that implements a multi-threaded application with three threads. The first thread generates a random integer every 1 second, and if the value is even, the second thread computes the square of the number and prints it. If the value is odd, the third thread will print the value of the cube of the number.
25. Write a program to create a dynamic array using ArrayList class and print the contents of the array object.
26. Write programs to implement add, search, and remove operations on ArrayList object.

Semester 3 - 4

a. **Course:** Computer Organization and Microprocessor

b. **Course Code:** 303105210

c. **Prerequisite:** Basic understanding of computer system

d. **Rationale:** This course provides detail of computer system's functional components, their characteristics, performance and interactions including system bus, different types of memory and input/output organization and CPU. This course also covers the architectural issues such as instruction set program and data types. On top that, the students are also introduced to the increasingly important area of parallel organization. This course also serves as a basic to develop hardware-related projects. And hence it is an important course for all students of computer engineering branch.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the Introduction to Microprocessor 8085.
CLOBJ 2	Study the Microprocessor architecture and interfacing.
CLOBJ 3	Demonstrate the Programming methods with Instructions of 8085.
CLOBJ 4	Learn different Additional Programming techniques.
CLOBJ 5	Understand 8085 Interrupts.
CLOBJ 6	Use different Computer Organization - Register Transfer and Basic Computer Design Register.
CLOBJ 7	Work on different Computer Organization - Assembler and Memory Organization Assembler.

f. **Course Learning Outcomes:**

CLO 1	Explain 8085 microprocessor architecture.
CLO 2	Design assembly language programs for the 8085 microprocessor.
CLO 3	Design interfacing with various hardware using the 8085 microprocessor.
CLO 4	Debug programs written in assembly language.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT-I: Introduction to Microprocessor 8085** Introduction to Microprocessor 8085, Instruction set and computer languages, 8085 Programming Model, Instruction Data Format and storage.	10%	4
2	**UNIT-II: Microprocessor Architecture and Interfacing** Microprocessor architecture and its operations, Memory and I/O devices, Memory interfacing, Interfacing I/O Devices.	20%	8
3	**UNIT-III: Programming Methods with Instructions** 8085 Instructions, Looping, Counting and Indexing, Logic operations, Rotate and Compare.	15%	5
4	**UNIT-IV: Additional Programming Techniques** Counter, Time delay, Stack & Subroutines, Restart, Call and Return Instructions, Code conversion.	15%	8
5	**UNIT-V: 8085 Interrupts** Interrupt structure of 8085 microprocessor, Processing of vectored and non-vectored interrupts, Latency time and response time; Handling multiple interrupts.	10%	5
6	**UNIT-VI: Computer Organization - Register Transfer and Basic Computer Design** Register Transfer: Register Transfer language, Bus design using multiplexer and Tri-state buffer, Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic Logical Shift Unit. Basic Computer Design: Memory-Reference Instructions, Register Reference Instructions, I/O Reference Instructions, Interrupt, Design of Accumulator Unit.	15%	8

7	**UNIT-VII: Computer Organization - Assembler and Memory Organization** Assembler: Machine Language, Assembly Language, Assembler, Program loops, Programming Arithmetic and Logic operations, Subroutines, I/O Programming. Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Flash memory, Associative memory, Cache memory, Virtual memory.	15%	8

i. Reference Books:

1. **Microprocessor Architecture, Programming, and Applications with the 8085** (Text Book)
2. **Computer System Architecture** By M.Morris Mano — PHI — 3rd Edition
3. **Microprocessor 8085 and its Interfacing** By Sunil Mathur — PHI Learning Pvt. Ltd
4. **8085 Microprocessor And its Applications** By A. NagoorKani — TMH Education Pvt. Ltd

j. List of Practicals:

1. Introduction to Microprocessor 8085 Microprocessor, Instruction set and computer languages, 8085 Programming Model, Instruction Data Format and storage.
2. Microprocessor architecture and interfacing Microprocessor architecture and its operations, Memory and I/O devices, Memory interfacing, Interfacing I/O devices.
3. Programming methods with Instructions 8085 Instructions, Looping, Counting and Indexing, Logic operations Rotate and Compare.
4. Additional Programming techniques Counter, Time delay, Stack & Subroutines, Restart, Call and Return Instructions, Code conversion.
5. 8085 Interrupts Interrupt structure of 8085 microprocessor, Processing of vectored and non-vectored interrupts, Latency time and response time; Handling multiple interrupts.
6. Computer Organization - Register Transfer and Basic Computer Design Register Transfer: Register Transfer language, Bus design using multiplexer and Tri-state buffer, Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic Logical Shift Unit. Basic Computer Design: Instruction codes, Computer registers, Computer instructions, Timing and Control, Instruction cycle, Memory-Reference Instructions, Register Reference Instructions, I/O Reference Instructions, Interrupt, Design of Accumulator Unit.

7. Computer Organization - Assembler and Memory Organization Assembler: Machine Language, Assembly Language, Assembler, Program loops, Programming Arithmetic and Logic operations, Subroutines, I/O Programming. Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Flash memory, Associative memory, Cache memory, Virtual memory.

Semester 3 - 5

a. **Course:** Discrete Mathematics

b. **Course Code:** 303191202

c. **Prerequisite:** Knowledge of Mathematics up to 12th science level

d. **Rationale:** The Mathematics I, Mathematics-II syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and vector calculus, preparing students for engineering challenges with optimized problem-solving skills.

e. **Course Learning Objectives:**

CLOBJ 1	Apply mathematical techniques to solve diverse real-world problems across different topics in Discrete Mathematics.
CLOBJ 2	Develop and apply analytical and critical thinking skills to understand, analyze, and evaluate mathematical structures and proofs.
CLOBJ 3	Recognize and interpret mathematical solutions within the context of specific problems, demonstrating practical applications in various fields.
CLOBJ 4	Clearly and effectively communicate mathematical concepts and solutions in both written and verbal forms, adapting to diverse topics.
CLOBJ 5	Present mathematical arguments and solutions in a unified, logical, and organized manner, emphasizing clarity, coherence, and precision.
CLOBJ 6	Establish a comprehensive foundation for more advanced courses in mathematics and related disciplines by demonstrating a thorough understanding of fundamental concepts.

f. Course Learning Outcomes:

CLO 1	Demonstrate proficient problem-solving skills, translating real-world problems into mathematical formulations and applying appropriate techniques for solutions.
CLO 2	Develop integrated analytical and critical thinking skills by engaging with a wide range of mathematical structures, proofs, and problem-solving techniques presented throughout the entire syllabus.
CLO 3	Understand and interpret mathematical solutions within the context of specific problems, recognizing the practical applications of discrete mathematics in diverse fields covered in all units.
CLO 4	Communicate mathematical concepts and solutions clearly and effectively, both in written and verbal forms, adapting communication styles to the diverse topics covered in each unit.
CLO 5	Present mathematical arguments and solutions in a unified, logical, and organized manner, emphasizing clarity, coherence, and precision across all units.
CLO 6	Lay a solid foundation for more advanced courses in mathematics and related disciplines.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
4	-	-	4	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT 1: Sets, Relation and Function** Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Size of a Set, Finite and Infinite Sets, Countable and Uncountable Sets, Cantor's Diagonal Argument and The Power Set Theorem, Schroeder-Bernstein Theorem.	8%	5
2	**UNIT 2: Principles of Mathematical Induction** The Well-Ordering Principle, Recursive Definition, The Division Algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic. Basic Counting Techniques: Inclusion and Exclusion, Pigeon-Hole Principle, Permutation and Combination.	15%	9
3	**UNIT 3: Propositional Logic** Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The Use of Quantifiers. Proof Techniques: Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency.	25%	15
4	**UNIT 4: Algebraic Structures and Morphism** Algebraic Structures with One Binary Operation: Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups. Algebraic Structures with Two Binary Operations: Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form.	17%	10
5	**UNIT 5: Graphs and Trees** Graphs and Their Properties: Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring Maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph: Definition, Properties and Example. Rooted Trees, Trees and Sorting, Weighted Trees and Prefix Codes, Bi-connected Component and Articulation Points, Shortest Distances.	10%	6

i. Reference Books:

1. **Discrete Mathematics and its Applications** By Kenneth H. Rosen , Tata McGraw – Hill
2. **Discrete Mathematical Structure and Its Application to Computer Science** By J.P. Tremblay and R. Manohar, Tata McGraw-Hill
3. **Discrete Mathematics with Applications, 4th Edition** By Susanna S. Epp, Wadsworth Publishing Co. Inc.
4. **Elements of Discrete Mathematics: A Computer-Oriented Approach** By C. L. Liu and D. P. Mohapatra, 3rd Edition, Tata McGraw – Hill.

Semester 3 - 6

a. Course: Professional Communication Skills

b. Course Code: 303193203

c. Prerequisite: Knowledge of English language in practical life

d. Rationale: Knowledge and application of English, Aptitude and Management Skills are crucial for better employability as well as professionalism.

e. Course Learning Objectives:

CLOBJ 1	Students will be able to demonstrate the ability to communicate clearly and persuasively in oral presentations.
CLOBJ 2	Students will practice active listening techniques to enhance understanding in professional interactions.
CLOBJ 3	Students will write professional emails, memos, and reports with clarity and conciseness.
CLOBJ 4	Students will understand and practice time management strategies effectively.
CLOBJ 5	Students will be able to demonstrate skills in resolving conflicts and negotiating effectively.
CLOBJ 6	Students will use digital communication tools and platforms effectively.

f. Course Learning Outcomes:

CLO 1	To develop advanced communication skills.
CLO 2	To become more proficient in formal writing.
CLO 3	To apply interpersonal communication skills to be more productive at the workplace.
CLO 4	To identify, set, and achieve goals with the help of time management.
CLO 5	To use a range of vocabulary to communicate effectively.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
0	2	0	2	0	100	0	0	0	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**Technical Writing: Email etiquette & Email writing, Letter Writing (Types of Letters & Layout)** Trains students on detailed email and letter writing etiquette. Students will be able to write formal letters following certain stipulated formats. They will learn different types of letters for different official purposes.	10%	4
2	**Interpersonal Communication at Workplace: Dynamics of communication** To develop the confidence to handle a wide range of demanding situations more effectively at the workplace. To enable the students to analyse their own interpersonal communication style.	10%	2
3	**Debate: The three minute debate planner** To enable the students to generate effective critical thinking into primary issues in the given topic. Students will be able to resolve controversies and recognize strengths and weaknesses of arguments.	10%	4
4	**Goal setting & Tracking** To enable the students to define strategies or implementation steps to attain the identified goals and make progress every day.	10%	2
5	**Time Management & Task Planning (Case-study)** To enable the students to identify their own time wasters and adopt strategies to reduce them. To enable students to clarify and prioritize their objectives and goals by creating more planning time.	5%	2
6	**Reading Comprehension: Intermediate level** To enable the students to develop the knowledge, skills, and strategies they must possess to become proficient and independent readers.	5%	2

7	**Listening Skills: Small everyday conversation & comprehension** Provides practice on understanding accents and day-to-day conversations. Listening to English conversations in different contexts.	10%	2
8	**Information design and writing for print and online media: Blog Writing** To enable students to design information that is targeted to specific audiences in specific situations to meet defined objectives. To create blogs and share their own knowledge and experience with the world.	5%	2
9	**Advanced vocabulary Building** The students will expand their vocabulary so as to enhance their proficiency in reading and listening to academic texts, writing, and speaking. The students will attain vocabulary to comprehend academic and social reading and listening texts. The students will develop adequate speaking skills to communicate effectively.	10%	4
10	**Picture Perception** To prepare the students for a test for basic intelligence and IQ, generally done on the first day of SSB (Sashastra Seema Bal is one of India's Central Armed Police Forces).	5%	1
11	**Appreciation, Apology and Acknowledgement letters** To enable the students to maintain productive business relationships through different types of letters. To enable the students to express their feelings without speaking out loud.	10%	2
12	**The Art of Negotiation** To enable the students to reach an agreement for mutual benefits through negotiation. To enable the students to learn a process by which compromise or agreement is reached while avoiding argument and dispute.	5%	2
13	**Activity Session (Game of Truth)** To make the students think of the significance of certain things in their life. To make them share their thoughts and perceptions of matters in life with others.	0%	1

i. Reference Books:

1. **Business Corespondence and Report Writing** By SHARMA, R. AND MOHAN.
2. **Communication Skills** By Kumar S And Lata P; New Delhi Oxford University
3. **Practical English Usage** By MICHAEL SWAN
4. **A Remedial English Grammar for Foreign Students** By F.T. WOOD

5. **On Writing Well** By William Zinsser; Harper Paperbacks, 2006; 30th anniversary edition

Semester 4 - 1

a. **Course:** Operating System

b. **Course Code:** 303105251

c. **Prerequisite:** Fundamentals of Computer Systems

d. **Rationale:** This course is an introduction to the theory and practice behind modern computer operating systems. Topics will include what an operating system does (and doesn't) do, system calls and interfaces, processes, concurrent programming, resource scheduling and management, virtual memory, deadlocks, algorithms, programming, and security. The approach of the subject is from both a theoretical perspective as well as a practical one.

e. **Course Learning Objectives:**

CLOBJ 1	Gain familiarity with the generation of Operating System, types of operating System, and the concept of a virtual machine.
CLOBJ 2	Solve problems related to Scheduling Algorithm and concepts of threading, multi-threading, etc.
CLOBJ 3	Acquire knowledge of Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, and Strict Alternation, Peterson's Solution, Semaphores, Event Counters, Monitors, Message Passing, and Classical IPC Problems.
CLOBJ 4	Understand Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection, and Recovery.
CLOBJ 5	Demonstrate a clear understanding of Memory Management, Memory allocation, and Paging.
CLOBJ 6	Study 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, etc.

f. Course Learning Outcomes:

CLO 1	Distinguish different styles of operating system design.
CLO 2	Understand device and I/O management functions in operating systems as part of a uniform device abstraction.
CLO 3	Understand disk organization and file system structure.
CLO 4	Give the rationale for virtual memory abstractions in operating systems.
CLO 5	Understand the main principles and techniques used to implement processes and threads as well as the different algorithms for process scheduling.
CLO 6	Understand the main mechanisms used for inter-process communication.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				T	**CE**	**P**	**Theory**	**P**	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Content**	**Weightage**	**Teaching Hours**
1	**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.	5%	3

2	**PROCESSES, THREAD & PROCESS SCHEDULING:** 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.	20%	9
3	**INTER-PROCESS COMMUNICATION:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer/Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dining Philosopher Problem etc.	15%	6
4	**DEADLOCKS:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.	10%	5
5	**MEMORY MANAGEMENT & VIRTUAL MEMORY:** 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: Principle of operation-Page allocation, Hardware support for paging, Protection and sharing, Disadvantages of 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).	30%	13

6	**I/O SYSTEMS, FILE & DISK MANAGEMENT:** 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. File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance. Disk Management: Disk structure, Disk scheduling algorithms - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.	20%	9

i. Reference Books:

1. **Operating System Concepts Essentials** By Avi Silberschatz, Peter Galvin, Greg Gagne — 9th Edition Wiley Asia Student Edition.(TextBook)
2. **Operating Systems Internals and Design Principles** By William Stallings — PHI — 5th Edition
3. **Operating System: A Design-oriented Approach** By Charles Crowley — 1st Edition - Irwin Publishing
4. **Operating Systems: A Modern Perspective** By Gary J. Nutt — Addison-Wesley — 2nd Edition
5. **Design of the Unix Operating Systems** By Maurice Bach — Prentice-Hall of India — 8th Edition
6. **Understanding the Linux Kernel** By Daniel P. Bovet, Marco Cesati — O'Reilly and Associates — 3rd Edition

j. Practical List

1. Study of Basic commands of Linux.
2. Study the basics of shell programming.
3. Write a Shell script to print given numbers sum of all digits.
4. Write a shell script to validate the entered date (e.g., Date format: dd-mm-yyyy).
5. Write a shell script to check if the entered string is a palindrome or not.
6. Write a Shell script to say "Good morning", "Good afternoon", or "Good evening" as you log in to the system.
7. Write a C program to create a child process.
8. Find the largest number from three numbers supplied as command line arguments.
9. Print patterns using a for loop in C.

10. Write a Shell script to determine whether a given file exists or not.

11. Write a C program for process creation using the gcc compiler.

12. Implement the First-Come-First-Served (FCFS) Scheduling Algorithm and Round Robin Scheduling Algorithm.

13. Implement the Banker's Algorithm.

Semester 4 - 2

a. **Course:** Software Engineering

b. **Course Code:** 303105253

c. **Prerequisite:** Basic knowledge of software applications

d. **Rationale:** This course provides a broad introduction to software engineering. The various process models required to develop software are also described. Moreover, the functional and non-functional requirements are also described.

e. **Course Learning Objectives:**

CLOBJ 1	Student will be able to understand about Software.
CLOBJ 2	Student will be able to create some Software.
CLOBJ 3	Student will be able to create some design about WAN or LAN.
CLOBJ 4	Understand different types of software.
CLOBJ 5	Demonstrate a software for testing purposes.
CLOBJ 6	Study about the Use Case study, CASE Tools, and Advanced Practices of System Dependability and Security.

f. **Course Learning Outcomes:**

CLO 1	Prepare and perform Software Requirement Specification and Software Project Management Plan.
CLO 2	Ensure the quality of software product, different quality standards, and software review techniques.
CLO 3	Apply the concept of Functional Oriented and Object-Oriented Approach for Software Design.
CLO 4	Understand modern Agile Development and Service Oriented Architecture Concept of Industry.
CLO 5	Analyze, design, verify, validate, implement, and maintain software systems.
CLO 6	Execute a Project Management Plan, tabulate Testing Plans, and reproduce effective procedures.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Content	Weightage	Teaching Hours
1	**Introduction:** Study of Different Models, Software Characteristics, Components, Applications, Layered Technologies, Processes, Methods and Tools, Generic View Of Software Engineering, Process Models - Waterfall model, Incremental, Evolutionary process models - Prototype, Spiral And Concurrent Development Model; **Agile Development:** Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools.	10%	6
2	**Software Project Management:** Management Spectrum, People – Product – Process – Project, W5HH Principle, Importance of Team Management; Planning a Software Project: Scope and Feasibility, Effort Estimation, Schedule and Staffing, Quality Planning, Risk Management - Identification, Assessment, Control, Project Monitoring Plan, Detailed Scheduling.	10%	5
3	**Requirements Engineering:** Problem Recognition, Requirement Engineering Tasks, Processes, Requirements Specification, Use Cases and Functional Specification, Requirements Validation, Requirements Analysis.	10%	5
4	**Structured System Design:** Design Concepts, Design Model, Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design, Alternative Architectural Designs, Modeling Component Level Design and Its Modeling, Procedural Design, Object-Oriented Design. **Data Oriented Analysis & Design:** Difference between Data and Information, E-R Diagram, Dataflow Model, Control Flow Model, Control and Process Specification, Data Dictionary.	15%	5

5	**Coding and Unit Testing:** Programming Principles and Guidelines, Programming Practices, Coding Standards, Incremental Development of Code, Management of Code Evaluation, Unit Testing - Procedural Units, Classes, Code Inspection, Metrics - Size Measure, Complexity Metrics, Cyclomatic Complexity, Halstead Measure, Knot Count, Comparison of Different Metrics.	10%	4
6	**Software Testing and Quality Assurance:** Concepts, Psychology of Testing, Levels of Testing, Testing Process - Test Plan, Test Case Design, Execution, Black-Box Testing – Boundary Value Analysis – Pairwise Testing – State Based Testing, White-Box Testing Criteria and Test Case Generation and Tool Support; **Quality Assurance:** Quality Control, Assurance, Cost, Reviews, Software Quality Assurance, Approaches to SQA, Reliability, Quality Standards - ISO9000 and 9001.	15%	7
7	**CASE Tools and Advanced Practices of System Dependability and Security:** Computer Aided Software Engineering Tools, SCRUM Developments, Dependable System, Reliability Engineering, Safety Engineering, Security Engineering, Resilience Engineering.	15%	5
8	**Advanced Software Engineering:** Software Reuse, Component Based Software Engineering, Distributed Software Engineering, Service-Oriented Software Engineering, Real-Time Software Engineering, Systems Engineering, Systems of System.	15%	5

i. Text Books and Reference Books:

1. **Software Engineering** By R.Pressman; 6th Edition (Text Book)
2. **Internet working with TCP/IP Principles, Protocols and Architecture**
3. **Software Engineering** By Sommerville
4. **Data Communication and Networking**

j. List of Practicals:

1. Project Definition and objective of the specified module and Perform Requirement Engineering Process.
2. Identify Suitable Design and Implementation model from the different software engineering models.
3. Prepare Software Requirement Specification (SRS) for the selected module.
4. Develop Software Project Management Planning (SPMP) for the specified module.
5. Do Cost and Effort Estimation using different Software Cost Estimation models.

6. Prepare System Analysis and System Design of identified Requirement Specification using structure design as DFD with data dictionary and Structure Chart for the specific module.
7. Designing the module using Object-Oriented approach including Use Case Diagram with scenarios, Class Diagram, State Diagram, Collaboration Diagram, Sequence Diagram, and Activity Diagram.
8. Defining Coding Standards and walkthrough.
9. Write the test cases for the identified module.
10. Demonstrate the use of different Testing Tools with comparison.
11. Define security and quality aspects of the identified module.

Semester 4 - 3

a. **Course:** Computer Network

b. **Course Code:** 303105255

c. **Prerequisite:** Knowledge of Computer and Information system

d. **Rationale:** This course is designed to provide basic knowledge about data & signals. It also provides basic concepts of computer networks and a firm foundation for understanding how data communication occurs in the Transmission Medium. It will help to develop logical abilities and practically set up the network.

e. **Course Learning Objectives:**

CLOBJ 1	Student will be able to understand about network.
CLOBJ 2	Student will be able to create some network connection.
CLOBJ 3	Student will be able to create some design about some WAN or LAN.
CLOBJ 4	Understand different types of switching, routers, and their tables.
CLOBJ 5	Demonstrate addressing mapping with the network.
CLOBJ 6	Study the use of TCP/IP protocol, etc.

f. **Course Learning Outcomes:**

CLO 1	Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs), and Wireless LANs (WLANs) and describe the function of each block.
CLO 2	Understand the functions of the different layers of the OSI Protocol.
CLO 3	Understand and design for a given requirement (small scale) of wide-area networks (WANs), local area networks (LANs), and Wireless LANs (WLANs) based on available market components.
CLO 4	Learn to address problem-related TCP/IP protocol developed for network programming.
CLO 5	Configure DNS, DDNS, TELNET, EMAIL, FTP, WWW, HTTP, SNMP, Bluetooth, and Firewalls using open-source available software and tools.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**Data Communication Components:** Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum	25%	11
2	**Data Link and Medium Acsess Sub Layer:** Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA.	25%	11
3	**Network Layer:** Switching, Logical addressing - IPV4, IPV6; Address mapping - ARP, RARP, BOOTP and DHCP - Delivery, Forwarding and Unicast Routing protocols	20%	8
4	**Transport Layer:** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.	15%	6
5	**Application Layer:** Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography.	15%	6

i. Reference Books:

1. **Computer Networks** By Andrew S. Tanenbaum and David J. Wetherall — PEARSON Edition (Text Book)
2. **Internetworking with TCP/IP Principles, Protocols, and Architecture** By Douglas E. Comer
3. **Basic Electrical Engineering** By D. C. Kulshreshtha, McGraw Hill, Pub. Year 2009
4. **Electrical and Electronics Technology** By E. Hughes Pearson, Pub. Year 2010

j. List of Practicals:

1. Experiments on Simulation Tools: (CISCO PACKET TRACER).
2. Experiments of Packet capture tool: Wireshark.
3. To study the behavior of generic devices used for networking: (CISCO PACKET TRACER).
4. Data Link Layer (Error Correction).
5. Virtual LAN.
6. Wireless LAN.
7. Inter networking with routers: 1: Experiment on same subnet 2: Perform Experiment across the subnet and observe functioning of Router via selecting suitable pair of Source and destination.
8. Implementation of SUBNETTING.
9. Routing at Network Layer.
10. Experiment on Transport Layer.

Semester 4 - 4

a. **Course:** Programming in Python with Full Stack Development

b. **Course Code:** 303105257

c. **Prerequisite:** Basic knowledge of Programming and web applications

d. **Rationale:** This course provides a broad introduction to Python programming and development of web applications. Developing and using Python as a scripting language for automating tasks and data processing. Moreover, building and deploying web applications using popular Python frameworks such as Django and Flask.

e. **Course Learning Objectives:**

CLOBJ 1	Gain familiarity with the fundamental concepts of web development and basic Python programming concepts.
CLOBJ 2	Gain knowledge to define functions in Python. Acquire knowledge of OOPS concepts.
CLOBJ 3	Acquire knowledge of how to work with modules and packages in Python.
CLOBJ 4	Understand and use the Flask framework.
CLOBJ 5	Understand and use the Django framework.
CLOBJ 6	Study the use of RESTful APIs.

f. **Course Learning Outcomes:**

CLO 1	Understand the fundamental concepts of web development.
CLO 2	Create and manipulate data using a variety of databases, including SQL and NoSQL.
CLO 3	Build and deploy web applications using a popular Python web framework, such as Django or Flask.
CLO 4	Design and implement APIs (application programming interfaces) that enable different applications to communicate with each other.
CLO 5	Test and debug web applications, and deploy them to production environments.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT-I: Introduction to Python Programming** Introduction to Python and basic programming concepts, variables, data types, conditional statements and loops, Lists, Sets, Tuples, Dictionaries: Working with strings, lists, sets, tuples and dictionaries, including common operations and built-in functions.	15%	6
2	**UNIT-II: Functions and OOPS Concepts** Defining and using functions, including the use of arguments and return values. OOPS Concepts: Object, class, abstraction, encapsulation, polymorphism, Inheritance. Exceptions and File handling: Handling exceptions and working with files.	20%	5
3	**UNIT-III: Modules and Packages** Working with modules and packages in Python. Introduction to popular Python libraries for specific tasks, such as data analysis, web development, or game development. PyCharm IDE: GIT- Git Integration with PyCharm IDE, PyTests. Python connectivity with Databases MYSQL, MongoDB CRUD operations.	15%	5
4	**UNIT-IV: Flask Framework** Introduction to Flask and web development with Python, Installation in Virtual Environment. Creation Routing App Settings URL Building HTTP methods Templates Working with Static, Media Files. Sending Form Data to Template. Flask App with Database connectivity Sqlite3, MySQL. Handling Exceptions and Errors Flash Message Working with Mails. Authenticating and authorizing users with Flask-Login, Deploying a Flask application to a web server.	20%	10

5	**UNIT-V: Django Framework** Introduction to Django framework, Django Project Installation in Virtual Environment. Phases in Django Project Creation Create a Project. Creation of Apps and their Structure. Working with ADMIN Console. Creating Views URL Mapping. Template System Working with Models. Form Processing static, media files, Django App Deployment.	20%	10
6	**UNIT-VI: RESTful APIs** Introduction to RESTful APIs and the REST architectural style, Understanding the HTTP protocol and its role in RESTful APIs, Designing and implementing RESTful APIs using common HTTP methods, such as GET, POST, PUT, and DELETE, Using URLs and resource representations to identify and transfer data in RESTful APIs, Implementing best practices for designing and implementing RESTful APIs, such as using HTTP status codes, versioning, and error handling, Consuming RESTful APIs using common tools and libraries, such as cURL, Postman, and the requests library in Python, Building scalable and secure RESTful APIs using common frameworks and libraries Flask or FastAPI.	10%	6

i. Reference Books:

1. **Fluent Python, 2nd Edition** By Luciano Ramalho (TextBook)
2. **Learn Python3 the Hard Way** By Zed Shaw
3. **Django for Beginners: Build websites with Python and Django** By William S. Vincent.
4. **Learning Django Web Development** By Samuli Natri.
5. **Flask Web Development with Python** By Miguel Grinberg.
6. **Mastering Flask** By Jack Stouffer.
7. **Building RESTful Python Web Services** By Gastón C. Hillar.
8. **Building Web APIs with FastAPI** By Samuel Colvin.

j. List of Practicals:

1. **Set-1**
    1. A program that converts temperatures from Fahrenheit to Celsius and vice versa.
    2. A program that calculates the area and perimeter of a rectangle.
    3. A program that generates a random password of a specified length.
    4. A program that calculates the average of a list of numbers.
    5. A program that checks if a given year is a leap year.

6. A program that calculates the factorial of a number.
7. A program that checks if a given string is a palindrome.
8. A program that sorts a list of numbers in ascending or descending order.
9. A program that generates a multiplication table for a given number.
10. A program that converts a given number from one base to another.

2. **Set-2**
    1. A program that models a bank account, with classes for the account, the customer, and the bank.
    2. A program that simulates a school management system, with classes for the students, the teachers, and the courses.
    3. A program that reads a text file and counts the number of words in it.
    4. A program that reads a CSV file and calculates the average of the values in a specified column.
    5. A program that reads an Excel file and prints the data in a tabular format.
3. **Set-3**
    1. A program that creates a simple web server and serves a static HTML page.
    2. A program that creates a web application that allows users to register and login.
    3. A program that creates a web application that allows users to upload and download files.
    4. A program that creates a web application that displays data from a database in a tabular format.
    5. A program that creates a web application that accepts user input and sends it to a server-side script for processing.
4. **Set-4**
    1. A program that creates a web application that uses a template engine to generate dynamic HTML pages.
    2. A program that creates a web application that supports AJAX requests and updates the page without reloading.
    3. A program that creates a web application that uses Django's built-in debugging features to troubleshoot errors and exceptions.
    4. A program that creates a web application that implements user authentication and authorization.
    5. A program that creates a web application that integrates with third-party APIs to provide additional functionality.
5. **Set-5**
    1. A program that creates a simple RESTful API that returns a list of users in JSON format.
    2. A program that creates a RESTful API that allows users to create, read, update, and delete resources.
    3. A program that creates a RESTful API that authenticates users using a JSON Web Token.

4. A program that creates a RESTful API that paginates the results of a query to improve performance.
5. A program that creates a RESTful API that supports data validation and error handling.

Semester 4 - 5

a. Course: Probability Statistics and Numerical Method

b. Course Code: 303191258

c. Prerequisite: Knowledge of Mathematics up to 12th science level

d. Rationale: The Mathematics I, Mathematics-II syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and vector calculus, preparing students for engineering challenges with optimized problem-solving skills.

e. Course Learning Objectives:

CLOBJ 1	Understand fundamental concepts of probability, probability spaces, conditional probability, and Bayes' Rule for making informed statistical decisions.
CLOBJ 2	Analyze discrete and continuous random variables, compute expectations and variances, and explore key distributions such as Binomial, Poisson, and Normal.
CLOBJ 3	Develop skills in hypothesis testing, including large sample tests for proportions, means, standard deviations, and chi-square tests for goodness of fit and independence.
CLOBJ 4	Demonstrate proficiency in numerical methods for solving linear equations, finding roots of algebraic and transcendental equations using techniques like Gauss-Jacobi, Gauss-Seidel, Bisection, Newton-Raphson, and Regula-Falsi.
CLOBJ 5	Master the concepts of finite differences, interpolation using Newton's Forward and Backward Difference Formula, Newton's Divided, and Lagrange's Formula for Unequal Intervals.
CLOBJ 6	Gain expertise in numerical integration techniques such as the Trapezoidal rule, Simpson's 1/3rd and 3/8th Rules, Gaussian Quadrature, and solve ordinary differential equations using methods like Taylor's series, Euler, Modified Euler, and Runge-Kutta of the fourth order for first and second-order equations.

f. Course Learning Outcomes:

CLO 1	Demonstrate proficient problem-solving skills, translating real-world problems into mathematical formulations and applying appropriate techniques for solutions.
CLO 2	Develop integrated analytical and critical thinking skills by engaging with a wide range of mathematical structures, proofs, and problem-solving techniques presented throughout the entire syllabus.
CLO 3	Understand and interpret mathematical solutions within the context of specific problems, recognizing the practical applications of discrete mathematics in diverse fields covered in all units.
CLO 4	Communicate mathematical concepts and solutions clearly and effectively, both in written and verbal forms, adapting communication styles to the diverse topics covered in each unit.
CLO 5	Present mathematical arguments and solutions in a unified, logical, and organized manner, emphasizing clarity, coherence, and precision across all units.
CLO 6	Lay a solid foundation for more advanced courses in mathematics and related disciplines.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
4	-	-	4	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**UNIT 1: Correlation, Regression and Curve Fitting** Correlation and Regression – Rank correlation, Curve Fitting by The Method of Least Squares- Fitting of Straight Lines, Second Degree Parabolas and More General Curves.	18%	8
2	**UNIT 2: Probability and Probability Distributions** Probability Spaces, Conditional Probability, Bayes' Rule, Discrete and Continuous Random Variables, Independent Random Variables, Expectation and Variance of Discrete and Continuous Random Variables, Distribution and Their Properties: Binomial Distribution, Poisson Distribution, Normal Distribution.	23%	10
3	**UNIT 3: Testing of Hypothesis** Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations. Test for single mean, difference of means, Test for ratio of variances, Chi-square test for goodness of fit and independence of attributes.	25%	15
4	**UNIT 4: Solution of a System of Linear Equations, Roots of Algebraic and Transcendental Equations** Gauss-Jacobi and Gauss-Seidel Methods, Solution of Polynomial and Transcendental Equations: Bisection Method, Newton-Raphson Method and Regula-Falsi Method.	18%	10
5	**UNIT 5: Finite Differences and Interpolation** Finite Differences, Relation between Operators, Interpolation using Newton's Forward and Backward Difference Formula. Newton's Divided and Lagrange's Formula for Unequal Intervals.	10%	6
6	**UNIT 6: Numerical Integration** Trapezoidal rule, Simpson's 1/3rd and 3/8th Rules, Gaussian Quadrature Formulae.	6%	2

i. Text Books:

1. **"Numerical Methods in Engineering & Science with Programs in C and C++"** By B. S. Grewal, Khanna Publishers.
2. **"Introduction to Numerical Analysis"** By C.E. Froberg, Addison-Wesley.
3. **"Introduction to Probability Theory"** By P. G. Hoel, S. C. Port and C. J. Stone,, Universal Book Stall, 2003 (Reprint).
4. **"Fundamentals of Mathematical Statistics"**By S.C. Gupta and V. K. Kapoor, Sultan Chand & Sons

Semester 4 - 6

a. **Course:** Professional Grooming & Personality Development

b. **Course Code:** 303193252

c. **Prerequisite:** Knowledge of English language in practical life

d. **Rationale:** Knowledge and application of English, Aptitude, and Management Skills are crucial for better employability as well as professionalism.

e. **Course Learning Objectives:**

CLOBJ 1	Students will be able to demonstrate the ability to communicate clearly and persuasively in oral presentations.
CLOBJ 2	Students will practice active listening techniques to enhance understanding in professional interactions.
CLOBJ 3	Students will write professional emails, memos, and reports with clarity and conciseness.
CLOBJ 4	Students will understand and practice time management strategies effectively.
CLOBJ 5	Students will be able to demonstrate skills in resolving conflicts and negotiating effectively.
CLOBJ 6	Students will use digital communication tools and platforms effectively.

f. **Course Learning Outcomes:**

CLO 1	Identify and develop soft skills required for personal and professional growth.
CLO 2	Develop professional etiquette & desired behavior at the workplace.
CLO 3	Speak and participate effectively in oral organizational communication.
CLO 4	Improve comprehensive skills for reading.
CLO 5	Know how to be assertive in a professional environment.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
0	1	0	1	0	100	0	0	0	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**Self Development and Assessment:** Various self-assessments for personal and professional development skills that are relevant to career development: Change, Grow, Persist, Prioritize, Read, Learn, Listen, Record, Remember, Guess, Think, Communicate, Relate, and Dream.	25%	4
2	**Corporate Etiquette:** Tips and guide to develop personality and gain various etiquettes manners, case studies, and activities. Telephone etiquette Etiquette for foreign business trips Etiquette for small talks Respecting privacy Learning to say 'No'	25%	4
3	**Public Speaking:** It's process of communicating information to an audience and is helpful in career advancement. Effective Public speaking skills includes: Choosing appropriate pattern Selecting appropriate method Art of persuasion Making speeches effective Delivering different types of speeches	20%	4
4	**Reading Skills Activity & Reading Comprehension:** Aims to improve students' comprehensive skills in English Language by getting them involved in reading activity and providing practice for reading comprehension.	15%	2

5	**Listening Skills- Inquiry Based Listening Questions:** Aims to improve students' listening skills in English Language providing them practice of various types of inquiry based listening tracks. Students will listen and will be able to find out details from the conversations.	15%	1

*Continuous Evaluation: It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

i. Reference Books:

1. **Business Correspondence and Report Writing** By SHARMA, R. AND MOHAN, K.
2. **Communication Skills** By Kumar S and Lata P; New Delhi Oxford University Press
3. **Practical English Usage** By MICHAEL SWAN
4. **A Remedial English Grammar for Foreign Student** By F.T. WOOD

Semester 4 - 7

a. Course: Competitive Coding

b. Course Code: 303105259

c. Prerequisite: Computer Programming and Basic Syntaxes

d. Rationale: Competitive coding enhances problem-solving abilities by exposing individuals to a wide range of algorithmic challenges. It fosters critical thinking and quick decision-making skills, crucial for real-world problem-solving in computer science and software development. Engaging in competitive coding also promotes continuous learning and improvement, as participants strive to optimize solutions and compete against global coding communities.

e. Course Learning Objectives:

CLOBJ 1	Analyze and assess time complexity in algorithmic solutions, enabling the ability to make informed judgments during problem-solving processes.
CLOBJ 2	Demonstrate proficiency in the application of various sorting algorithms, employing them effectively to organize and manipulate data structures for problem-solving purposes.
CLOBJ 3	Evaluate problem requirements and make informed decisions on selecting the most suitable data structure to optimize solution efficiency and address specific problem constraints.
CLOBJ 4	Employ diverse problem-solving techniques to effectively tackle a range of challenges, showcasing adaptability and resourcefulness in approaching and resolving different types of problems.

f. Course Learning Outcomes:

CLO 1	Judge time complexity rules during problem solving.
CLO 2	Apply sorting algorithms to data structures to solve problems.
CLO 3	Select the best data structure to solve the given problem.
CLO 4	Solve given problems using different Problem-Solving Techniques.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				T	**P**	**CE**	**Theory**	**P**	
-	-	4	2	-	20	-	-	30	50

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. List of Practicals:

1. Write a program for implementing a MINSTACK which should support operations like push, pop, overflow, underflow, display.
   i. Construct a stack of N-capacity
   ii. Push elements
   iii. Pop elements
   iv. Top element
   v. Retrieve the min element from the stack
2. Write a program to deal with real-world situations where Stack data structure is widely used. Evaluation of expression: Stacks are used to evaluate expressions, especially in languages that use postfix or prefix notation. Operators and operands are pushed onto the stack, and operations are performed based on the LIFO principle.
3. Write a program for finding the Next Greater Element (NGE) from an array.
4. Write a program to design a circular queue (k) which should implement the following functions:
   i. Enqueue
   ii. Dequeue
   iii. Front
   iv. Rear
5. Write a program for an infix expression, and convert it to postfix notation. Use a queue to implement the Shunting Yard Algorithm for expression conversion.
6. Write a program for finding the Product of the three largest Distinct Elements. Use a Priority Queue to efficiently find and remove the largest elements.
7. Write a program to Merge two sorted linked lists.
8. Write a program to find the Merge point of two sorted linked lists.
9. Write a program to Swap Nodes pairwise in a linked list.
10. Write a program for building a function ISVALID to validate a Binary Search Tree (BST).

11. Write a program to Build a BST.
12. Write a program to determine the depth of a given Tree by implementing MAXDEPTH.
13. Write a program to understand and implement Tree traversals, i.e., Pre-Order, Post-Order, In-Order.
14. Write a program to perform Boundary Traversal on a BST.
15. Write a program for Lowest Common Ancestors (LCA) in a BST.
16. Write a program to verify and validate mirrored trees.
17. Write a program for a basic hash function in a programming language of your choice. Demonstrate its usage to store and retrieve key-value pairs.
18. Implement a hash table using separate chaining for collision handling. Perform operations like insertion, deletion, and search on the hash table.
19. Write a program to implement Two Sums using a HashMap.
20. Write a program to implement search, insert, and remove operations in a Trie.
21. Write a program to implement Huffman coding.
22. Write a program to find distinct substrings in a string.
23. Write a program to find the number of words in a Trie.
24. Write a program to view a tree from the left view.
25. Write a program to traverse a tree using Level Order Traversal.

i. Text Books:

1. **Algorithms, Part I and II** By Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. **Introduction to Algorithms** By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.
3. **The Algorithm Design Manual** By Steven S. Skiena, Springer.
4. **Data Structures and Algorithm Analysis in C++** By Mark Allen Weiss, Addison-Wesley.
5. **Fundamental Data Structures** By Sahni, A. and Horowitz, E., McGraw-Hill.

Semester 5-1

a. Course: Design and Analysis of Algorithm

b. Course Code: 303105218

c. Prerequisite: Data structures, Fundamentals of programming

d. Rationale: Analyze the asymptotic performance of algorithms. Write rigorous correctness proofs for algorithms. Demonstrate a familiarity with major algorithms and data structures. Apply important algorithmic design paradigms and methods of analysis. Synthesize efficient algorithms in common engineering design situations.

e. Course Learning Objectives:

CLOBJ 1	Develop the ability to analyze the running time of any given algorithm using asymptotic analysis and prove the correctness of basic algorithms.
CLOBJ 2	Design efficient algorithms for computational problems, using various algorithm design techniques taught in the course.
CLOBJ 3	Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate.
CLOBJ 4	Analyze String matching algorithms.
CLOBJ 5	Explain the complexity classes P, NP, and NP-Complete, and demonstrate the NP-Completeness of a specific problem.

f. Course Learning Outcomes:

CLO 1	Develop the ability to analyze the running time of any given algorithm using asymptotic analysis and prove the correctness of basic algorithms.
CLO 2	Design efficient algorithms for computational problems, using various algorithm design techniques taught in the course.
CLO 3	Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate.
CLO 4	Analyze String matching algorithms.
CLO 5	Explain the complexity classes P, NP, and NP-Complete, and demonstrate the NP-Completeness of specific problems.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	-	4	5	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Topics**	**Weightage (%)**	**Teaching Hours**
1	Introduction and Analysis of Algorithms: Algorithm: Definition, Properties, Types of Algorithms, Writing an Algorithm, Analysis: Parameters, Design Techniques of Algorithms Asymptotic Analysis: Big Oh, Big Omega & Big Theta Notations, Lower Bound, Upper Bound and Tight Bound, Best Case, Worst Case, Average Case, Analyzing control statement, Loop invariant and the correctness of the algorithm, Recurrences-substitution method, recursion tree method, master method. Sorting Techniques with analysis: Bubble Sort, Selection Sort, Insertion Sort.	20%	10
2	Divide & Conquer Algorithms: Structure of divide-and-conquer algorithms, Examples: Binary search, Quick sort, Merge sort, Strassen's Multiplication, Max-Min problem.	20%	6
3	Greedy Algorithms: Introduction, Elements of Greedy Strategy, Minimum Spanning Tree: Kruskal's & Prim's Algorithm, Dijkstra's Algorithm, Knapsack Problem, Activity Selection Problem, Huffman Codes.	20%	8
4	Dynamic Programming: Principle of Optimality, 0/1 Knapsack Problem, Making Change Problem, Chain Matrix Multiplication, Longest Common Subsequence, All Pair Shortest Paths: Warshall's and Floyd's Algorithms.	20%	8
5	Exploring Graphs: An introduction using graphs and games, Undirected Graph, Directed Graph, Traversing Graphs, Depth First Search, Breadth First Search, Topological Sort.	5%	3
6	Backtracking and Branch & Bound: Introduction to Backtracking, Introduction to Branch & Bound, 0/1 Knapsack Problem, N-Queens Problem, Travelling Salesman Problem.	5%	4

7	String Matching & NP Completeness: String Matching: Introduction to String Matching, Naive String Matching, Rabin-Karp Algorithm, Kruth-Morris-Pratt Algorithm, String Matching using Finite Automata. NP Completeness: Introduction to NP Completeness, P Class Problems, NP Class Problems, Hamiltonian Cycle.	10%	6

i. Reference Books:

1. **"Introduction to Algorithms, 4TH Edition"** By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press/McGraw-Hill. (TextBook)
2. **"Fundamentals of Algorithms"** By E. Horowitz et al. (TextBook)
3. **"Algorithm Design, 1ST Edition"** By Jon Kleinberg and Éva Tardos, Pearson.
4. **"Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition"** By Michael T. Goodrich and Roberto Tamassia, Wiley.
5. **"Algorithms—A Creative Approach, 3RD Edition"** By Udi Manber, Addison-Wesley, Reading, MA.

j. List of Practical:

1. Write a program to determine whether the given number is Prime or not.
2. Given a sorted array and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.
3. There are N children standing in a line with some rating value. You want to distribute a minimum number of candies to these children such that:
   - Each child must have at least one candy.
   - The children with higher ratings will have more candies than their neighbours.

   You need to write a program to calculate the minimum candies you must give.
4. There is a new barn with N stalls and C cows. The stalls are located on a straight line at positions x_1, x_N $(0 \leq x_i \leq 1,000,000,000)$. We want to assign the cows to the stalls, such that the minimum distance between any two of them is as large as possible. What is the largest minimum distance?
5. Given an undirected graph with V vertices and E edges, check whether it contains any cycle or not.
6. There are n servers numbered from 0 to n–1 connected by undirected server-to-server connections forming a network where connections$[i] = [a_i, b_i]$ represents a connection between servers a_i and b_i. Any server can reach other servers directly or indirectly through the network. A critical connection is a connection that, if removed, will make some servers unable to reach some other servers. Return all critical connections in the network in any order.

7. Given a grid of size $N \times M$ (N is the number of rows and M is the number of columns in the grid) consisting of '0's (Water) and '1's (Land). Find the number of islands.

8. Given a grid of dimension $N \times M$ where each cell in the grid can have values 0, 1, or 2 which has the following meaning:

   - 0: Empty cell
   - 1: Cells have fresh oranges
   - 2: Cells have rotten oranges

   We have to determine what is the minimum time required to rot all oranges. A rotten orange at index $[i, j]$ can rot other fresh oranges at indexes $[i-1, j]$, $[i+1, j]$, $[i, j-1]$, $[i, j+1]$ (up, down, left and right) in unit time.

9. Given two strings str1 and str2 and below operations that can be performed on str1. Find the minimum number of edits (operations) required to convert 'str1' into 'str2'.

   - Insert
   - Remove
   - Replace

   All of the above operations are of equal cost.

10. The "Minimum Path Sum" problem states that given an $n \times m$ grid consisting of non-negative integers, we need to find a path from top-left to bottom-right, which minimizes the sum of all numbers along the path.

11. Given string num representing a non-negative integer num, and an integer k, return the smallest possible integer after removing k digits from num.

12. There is a robot on an $m \times n$ grid. The robot is initially located at the top-left corner (i.e., grid[0][0]). The robot tries to move to the bottom-right corner (i.e., grid$[m-1][n-1]$). The robot can only move either down or right at any point in time. Given the two integers m and n, return the number of possible unique paths that the robot can take to reach the bottom-right corner.

Semester 5-2

a. Course: Theory of Computation

b. Course Code: 303105306

c. Prerequisite: Calculus, Data Structures and Algorithms

d. Rationale: Formal Language & Automata Theory helps in natural language processing to solve a problem on a model of computation, using an algorithm. It enables to learn in which machine can be made to think.

e. Course Learning Objectives:

CLOBJ 1	Understand formal language basics including alphabets, grammars, and Chomsky hierarchy.
CLOBJ 2	Design and analyze finite automata and regular expressions with their properties and equivalences.
CLOBJ 3	Study context-free and context-sensitive grammars and their corresponding computational models.
CLOBJ 4	Model computation using Turing machines and explore their capabilities and variants.
CLOBJ 5	Explore undecidability and the limits of algorithmic computation through key theoretical concepts.

f. Course Learning Outcome:

CLO 1	Recognize the basic concepts and applications of theory of Computation.
CLO 2	Solve Computational Problems using Regular Languages and Finite Automata.
CLO 3	Solve Computational Problems using Context free Grammar and Push Down Automata.
CLO 4	Design Turing Machine for simple computational Problems.
CLO 5	Analyze various concepts of undecidability and Computable Function.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	0	3	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction:** Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages	5%	2
2	**Regular languages and finite automata:** Regular expressions and languages, deterministic finite automata -(DFA) and equivalence with regular expressions,Moore machines and mealy machines, Conversion from Mealy to Moore and vice versa, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata.	30%	12
3	**Grammars:** Context-free grammars (CFG) and languages (CFL), Chomsky normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic pushdown automata, closure properties of CFLs. , Context-sensitive languages: Context-sensitive grammars (CSG) and languages.	35%	15
4	**Turing machines:** The basic model for Turing machines (TM), Turing-recognizable (recursively enumerable) and Turing- decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.	25%	10

5	**Undecidability:** Church Turing thesis, universal Turing machine, the universal and diagonalization languages	5%	6

i. Reference Books:

1. **Introduction to Automata theory, languages and Computation** By John E. Hopcroft, Rajiv Motwani and Jeffery D. Ullman — Pearson (Text-Book)
2. **Elements of the Theory of Computation** By Harry R.Lewis and Christos H. Papadimitriou — Pearson Education Asia
3. **Introduction to the Theory of Computation** By Michael Sipser — PWS Publishing
4. **Introduction to Languages and the Theory of Computation** By John C. Martin — McGraw Hill
5. **Automata and Computability** By Dexter C. Kozen — Undergraduate Texts in Computer Science, Springer

Semester 5-3

a. Course: Artificial Intelligence

b. Course Code: 303105307

c. Prerequisite: Data structure, Formal Languages and Automata Theory, Mathematics.

d. Rationale: This course provides a broad introduction to Artificial Intelligence. AI techniques for search and knowledge representation also Apply knowledge of AI planning and machine learning techniques to real-world problems.

e. Course Learning Objectives:

CLOBJ 1	Understand the fundamentals, historical development, and major application areas of Artificial Intelligence including intelligent agents and AI problem-solving techniques.
CLOBJ 2	Analyze and implement various search strategies such as A*, hill climbing, constraint satisfaction, and heuristic-based approaches for solving AI problems.
CLOBJ 3	Apply knowledge representation techniques like propositional logic, predicate logic, and frame notation for building intelligent systems.
CLOBJ 4	Evaluate reasoning under uncertainty using Bayesian networks, Markov models, and utility theory to support decision-making in dynamic environments.
CLOBJ 5	Implement fuzzy logic concepts and expert systems for handling imprecise and linguistic data in real-world scenarios.
CLOBJ 6	Utilize Natural Language Processing (NLP) techniques and neural networks for text processing, semantic analysis, and building AI-enabled applications.

f. Course Learning Outcome:

CLO 1	Discuss AI fundamentals, history, and future trends to develop solutions for problem-solving, inference, perception, knowledge representation, and learning tasks.
CLO 2	Utilize knowledge representation methods like propositional logic, predicate logic, and frame notation to effectively represent knowledge within AI systems.
CLO 3	Discover methods for solving AI problems, including diverse search algorithms and techniques like non-monotonic reasoning, probability theory, Bayesian networks, and fuzzy logic for effective decision-making in uncertain scenarios.
CLO 4	Apply Natural Language Processing (NLP), Neural Networks and Expert Systems technologies effectively in real-world scenarios.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction:** Definition of an AI, Major Areas of Artificial Intelligence, AI Techniques, History, AI problems, Production Systems,Problem characteristics, Intelligent Agents, Agent Architecture, AI Application (E-Commerce, & Medicine), AI Representation, Properties of internal representation, Future scope of AI, Issues in the design of search algorithms. Introduction to AI Problems and Applications, Defining Problems as a State Space Search, Problem Characteristics, Production Systems.	15%	7
2	**Search Techniques:** Generate-And-Test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, Means-Ends Analysis. Heuristic search, Hill Climbing, Best first search, mean and end analysis, Constraint Satisfaction, A* and AO* Algorithm, Knowledge Representation: Basic concepts, Knowledge representation Paradigms, Propositional Logic, Inference Rules in Propositional Logic, Knowledge representation using Predicate logic, Predicate Calculus, Predicate and arguments, ISA hierarchy, Frame notation, Resolution, Natural Deduction	20%	8

3	**Knowledge Representation:** Representation and Mappings, Different Approaches, Issues in knowledge representation.Predicate Logic - Representation Simple Facts in Logic, Representing Instance and Isa Relationships, Computable Functions and Predicates, Resolution. Propositional Logic: Representation, Inference, Reasoning Patterns, Resolution, First-order Logic: Representation, Inference, Reasoning Patterns, Resolution	15%	8
4	**Uncertainty:** Non-Monotonic Reasoning, Logics for Non-Monotonic Reasoning, Forward rules, and Backward rules, Justification based Truth Maintenance Systems, Semantic Nets Statistical Reasoning, Probability and Bayes' theorem, Bayesian Network, Markov Networks, Hidden Markov Model, Basis of Utility Theory, Utility Functions.	15%	4
5	**Fuzzy Sets and Fuzzy Logic:** Fuzzy Set Operations, Membership Functions, Fuzzy Logic, Hedges, Fuzzy Proposition and Inference Rules, Fuzzy Systems.	10%	5
6	**Natural Language Processing:** Introduction, Syntactic Processing, Semantic Analysis, Semantic Analysis, Discourse and Pragmatic Processing, Spell Checking.	10%	5
7	**Neural Networks and Expert systems:** Introduction to neural networks and perception-qualitative Analysis, Neural net architecture and applications, Utilization and functionality, the architecture of the expert system, knowledge representation, two case studies on expert systems	18%	5

i. Reference Books:

1. **Artificial Intelligence: A New Synthesis, Harcourt Publishers** By N. J. Nilsson — Harcourt Publishers (TextBook)
2. **Artificial Intelligence** By Elaine Rich and Kevin Knight — TMH (TextBook)
3. **Artificial Intelligence-Structures and Strategies For Complex Problem Solving** By George F. Luger — Pearson Education / PHI
4. **Artificial Intelligence-A Modern Approach** By Stewart Russell and Peter Norvig — Pearson Education/ Prentice Hall of India — 2
5. **Artificial Intelligence – A Practical Approach** By Patterson — Tata McGraw Hill — 3

j. List of Practicals:

1. Develop an AI-based medical diagnosis system using expert systems architecture and knowledge representation techniques.
2. Build an intelligent agent to optimize inventory management in e-Commerce using search algorithms such as hill climbing and best-first search.
3. Implement a constraint satisfaction algorithm to solve scheduling problems in healthcare facilities.
4. Create a personalized learning recommendation system using means-end analysis and heuristic search techniques.
5. Develop a problem solving agent to optimize resource allocation in logistics using the A * and AO * algorithms.
6. Develop a fuzzy logic-based system to predict stock market trends considering uncertain market conditions.
7. Write a program to implement BFS (Water Jug problem or any AI search problem).
   - Write a program to implement DFS (Water Jug problem or any AI search problem).
8. Define a predicate brother(X,Y) that holds iff X and Y are brothers.
   Define a predicate cousin(X,Y) that holds iff X and Y are cousins.
   Define a predicate grandson(X,Y) that holds iff X is a grandson of Y.
   Define a predicate descendent(X,Y) that holds iff X is a descendent of Y.
   Consider the following genealogical tree:
   father(a,b).
   father(a,c).
   father(b,d).
   father(b,e).
   father(c,f).
   Say which answers, and in which order, are generated by your definitions for the following queries in Prolog:
   ?- brother(X,Y).
   ?- cousin(X,Y).
   ?- grandson(X,Y).
   ?- descendent(X,Y).
9. Write a program to implement the Tic-Tac-Toe game using Python.
10. Create a spell-checking application that uses natural language processing (NLP) techniques, including syntactic and semantic analysis.
11. Design a neural network architecture for pattern recognition in medical imaging for disease diagnosis.

Semester 5-4

a. Course: Enterprise Programming using Java

b. Course Code: 303105309

c. Prerequisite: Basic knowledge of software applications.

d. Rationale: This course provides a broad introduction to software engineering. The various process models required to develop software is also being described. Moreover the functional and non-functional requirements are also described.

e. Course Learning Objectives:

CLOBJ 1	Understand the architecture and functioning of JDBC and Maven, and utilize them to establish connectivity and manage dependencies in enterprise applications.
CLOBJ 2	Apply the concepts of Servlets including configuration, session tracking, and CRUD operations to develop dynamic web-based applications.
CLOBJ 3	Demonstrate the use of JSP scripting and directive elements to design server-side logic and perform CRUD operations in enterprise-level applications.
CLOBJ 4	Implement Hibernate ORM concepts such as annotations, transaction management, and HQL/Criteria queries for efficient database handling.
CLOBJ 5	Design modular enterprise applications using Spring Framework, incorporating Dependency Injection, Autowiring, and MVC patterns.
CLOBJ 6	Develop and deploy scalable web applications using Spring Boot, with features such as AOP, RESTful services, and application context configuration.

f. Course Learning Outcome:

CLO 1	Analyze the structure and operations of JDBC, and apply this knowledge to connect and interact with Oracle and MySQL databases.
CLO 2	Perform the concepts of Servlet Configuration and Context, and apply these in practical scenarios.
CLO 3	Apply their knowledge to perform CRUD operations using JSP and Hibernate and evaluate the results for correctness and efficiency.
CLO 4	Design and create a web application using Spring Boot.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
2	0	2	3	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours(%)
1	**Foundation of Enterprise Programming:** JDBC, JDBC architecture, JDBC with Oracle, MySql, Maven: integration with eclipse, POM.xml	10%	3
2	**Servlets:** Basics of Web, Servlet Lifecycle, Servlets API, HTTP Servlets with XML and annotation, Servlets Configuration, Servlets Context, Servlets Collaboration, Session Tracking, CRUD operations	15%	4
3	**Java Server Programming:** Scripting elements, Directive elements, CRUD operations.	15%	4
4	**Hibernate (ORM):** Architecture, JPA, Generator class, Dialects, Mapping, Annotations, Transaction Management, HQL, HCQL, CRUD operations.	20%	6
5	**Spring:** Architecture, Modules, Dependency Injection, Autowire, Application Context, annotation-based configuration, MVC CRUD operations	20%	7
6	**Spring Boot:** Dependency Injection, Web App using spring boot, Spring boot AOP, spring boot Database, Spring Rest	20%	6

i. Reference Books:

1. **Java Enterprise in a Nutshell** by Jim Farley, William Crawford, and David Flanagan (TextBook)
2. **Java EE 8 Design Patterns and Best Practices** by Rhuan Rocha
3. **Java EE and HTML5 Enterprise Application Development** by John Brock, Arun Gupta, and Geertjan Wielenga
4. **Java 8 Programming Black Book**

j. List of Practicals:

1. Write a program to insert and retrieve the data from database using JDBC.
2. Write a program to demonstrate the use of Prepared Statement and Result Set interface.
3. Servlet Programming Servlet Execution on tomcat A servlet program to print hello world A servlet program to display request details A servlet program to handle user form A servlet program to create a cookie A servlet program to display cookie A servlet program to do session tracking Write a program to implement chat Server using Server Socket and Socket class. Write a Servlet program to send username and password using HTML forms and authenticate the user
4. JSP Programming JSP program to display hello world. JSP program to demonstrate arithmetic operations JSP program to demonstrate jsp: forward action tag JSP program to request implicit object Developing a web application to insert record into Oracle Database using JSP and JDBC
5. Create application to store the data in database to perform Hibernate CRUD operations.
6. Create a application store the data in database to perform Spring CRUD operations.
7. Create a web application to store the data in database with spring boot.

Semester 5 - 5

a. **Course:** Professionalism & Corporate Ethics

b. **Course Code:** 303193304

c. **Prerequisite:** Basic knowledge of SWOT analysis and understanding of the fundamentals of communication are essential.

d. **Rationale:** Soft skills and ethics are essential for career growth.

e. **Course Learning Objectives:**

CLOBJ 1	Student will define and articulate the principles of professionalism in a corporate context.
CLOBJ 2	Student will be able to develop the ability to analyse ethical dilemmas and make informed decisions.
CLOBJ 3	Student will apply ethical decision-making models to real-world business scenarios
CLOBJ 4	Student will evaluate the impact of corporate activities on various stakeholders, including the community and the environment.
CLOBJ 5	Student will be able to understand and practice proper business etiquette in various communication channels.
CLOBJ 6	Student will be able to develop skills in resolving conflicts ethically and professionally.

f. **Course Learning Outcomes:**

CLO 1	Identity and develop soft skills required for personal and professional growth.
CLO 2	Develop professional etiquette & desired behaviour at the workplace
CLO 3	Speak and participate effectively in oral organizational communication
CLO 4	Improve comprehensive skills for reading.
CLO 5	Know how to be assertive in professional environment

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
0	1	0	1	0	100	0	0	0	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	**Ethics in Engineering** Scope of engineering ethics Accepting & sharing responsibility Responsible professionals and ethical corporations Resolving ethical dilemmas Case studies	20%	5
2	**Group Discussion** Communication core Definition, types, process, guidelines Mock round -1	15%	3
3	**Introduction to B-School Tests** Students will be able to solve verbal questions from the following exams. In these sessions students will learn to distinguish between national & international level of Management exam like GMAT & CAT	15%	2
4	**Listening Skills- Advanced Level** Demonstrate ability to listen more than two minutes of audio clips & solve questions based on it.	10%	1
5	**Preparing Brochures** Students will learn how to establish the purpose of writing & determine audience they are writing for.	15%	2
6	**Agenda & Minutes of Meeting** Students will be able to explain what an agenda & minutes of meeting are and why they are useful.	15%	1
7	**Reading Comprehension; Intermediate level** Students will develop their ability to skim for main idea(s). They will able to make use of contextual clues to infer meaning of unfamiliar words from context and will be able to solve questions based on it.	10%	1

i. Reference Books:

1. **Business Correspondence and Report Writing** By SHARMA, R. AND MOHAN, K.
2. **Ethics in Engineering Practice and Research** By Caroline Whitbeck, Cambridge University Press
3. **Technical Communication : Principles And Practice** By Sangeetha Sharma, Meenakshi Raman; Oxford University Press
4. **How to prepare for verbal ability and reading comprehension for the CAT** By Arun Sharma, Meenakshi Upadhyay, TATA McGRAW HILL'S

Semester 5-6

a. Course: Quant and Reasoning

b. Course Code: 303105311

c. Prerequisite: Good fundamentals in calculations and ability to think logically.

d. Rationale: The course focuses on building core aptitude and reasoning skills. These include analytical thinking, problem-solving, and logical decision-making abilities—vital for engineers in real-world scenarios such as software development, project analysis, and competitive assessments.

e. Course Learning Objectives:

CLOBJ 1	Understand core mathematical concepts related to number systems, averages, ratios, and profit/loss.
CLOBJ 2	Apply logical reasoning techniques to problems involving directions, seating arrangements, syllogisms, clocks, and calendars.
CLOBJ 3	Develop speed and accuracy in solving aptitude problems through practice and structured approaches.
CLOBJ 4	Solve real-world problems using quantitative methods like permutations, combinations, and probability.
CLOBJ 5	Prepare for competitive exams and placement tests by strengthening analytical and reasoning abilities.

f. Course Learning Outcome:

CLO 1	Apply logic and critical thinking skills to analyze information and draw logical conclusions.
CLO 2	Solve complex problems by breaking them into manageable parts and creating effective solutions.
CLO 3	Demonstrate the ability to approach problem-solving from different perspectives.
CLO 4	Master foundational quantitative techniques for competitive and academic aptitude tests.
CLO 5	Evaluate and solve diverse analytical puzzles and reasoning scenarios confidently.

g. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**UNIT-1:** Number system, LCM & HCF simplifications and approximations	9%	4
2	**UNIT-2:** Averages, progressions	9%	4
3	**UNIT-3:** Ratio and proportion, Problems on Ages, Percentages	12%	5
4	**UNIT-4:** Profit & loss, partnerships, S.I & C.I	12%	5
5	**UNIT-5:** Time & work, pipes and cisterns, Time speed and distance, Problems on train crossings, Boats & streams	18%	8
6	**UNIT-6:** Permutations & combinations, probability	11%	5
7	**UNIT-7:** Directions, seating arrangements	4%	2
8	**UNIT-8:** Clocks, calendars	6%	3
9	**UNIT-9:** Cubes & Dice, syllogisms	9%	4
10	**UNIT-10:** Blood Relations	5%	2
11	**UNIT-11:** Series, Analogy, odd man out, coding and Decoding	5%	3

h. Reference Books:

1. **Quantitative Aptitude for CAT** By Arun Sharma (TextBook)
2. **Logical Reasoning for CAT** By Arun Sharma
3. **Quantitative Aptitude** By Abhijit Guha

Semester 5-7

a. Course: Azure Fundamentals

b. Course Code: 303105302

c. Prerequisite: Basic understanding of computer concepts and basic programming

d. Rationale: This course provides a broad introduction to Azure cloud, infrastructure, services, security and compliance, as well as billing, pricing, and support plans.

e. Course Learning Objectives:

CLOBJ 1	Describe fundamental cloud computing principles, including cloud deployment models and service models (IaaS, PaaS, SaaS).
CLOBJ 2	Identify and utilize various Azure services such as Virtual Machines, App Services, Storage, Functions, and SQL Database for business and technical scenarios.
CLOBJ 3	Apply security best practices in Azure, manage user identities using Azure Active Directory, and ensure compliance with security frameworks.
CLOBJ 4	Evaluate different Azure pricing models, optimize cloud spending, and select appropriate support plans based on business requirements.
CLOBJ 5	Analyze Azure Service Level Agreements (SLAs) and understand Azure service lifecycles, including updates, maintenance, and deprecation policies.

f. Course Learning Outcome:

CLO 1	Describe fundamental cloud computing principles and cloud deployment models.
CLO 2	Identify and use various Azure services such as Virtual Machines, App Services, Storage, Functions, and SQL Database for different business and technical scenarios.
CLO 3	Apply security best practices in Azure and manage user identities with Azure Active Directory.
CLO 4	Evaluate different Azure pricing models, optimize cloud spending, and choose appropriate support plans based on business needs.
CLO 5	Analyze Azure Service Level Agreements (SLAs) and ensure smooth transitions during the Azure service lifecycle.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
2	0	0	2	20	20	-	60	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours(%)
1	**Cloud Concepts:** Understanding cloud computing principles, such as the different types of cloud models (public, private, hybrid),infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS).	15%	6
2	**Azure Services:** Familiarity with the various Azure services and their common use cases. This includes services like Azure Virtual Machines, Azure App Services, Azure Storage, Azure Functions, Azure SQL Database, and more	20%	7
3	**Security, Privacy, Compliance, and Trust:** Knowledge of Azure security features, identity and access management, Azure Active Directory, data protection, compliance frameworks, and Azure governance methodologies.	25%	5
4	**Azure Pricing and Support:** Understanding Azure subscription options, cost management, pricing models, and the different support options available to Azure customers	15%	5
5	**Azure SLA and Service Lifecycles:** Familiarity with Azure Service Level Agreements (SLAs) and the Azure service lifecycle, including planned maintenance, updates, and deprecation policies.	25%	7

i. Reference Books:

1. **Microsoft Azure Fundamentals: Understanding Azure** By Michael Collier and Robin Shahan - 3rd Edition (TextBook)
2. **Azure for Architects: Implementing cloud design, DevOps, containers, IoT, and serverless solutions on your public cloud** By Ritesh Modi - 2nd Edition
3. **Exam Ref AZ-900 Microsoft Azure Fundamentals** By Jim Cheshire - 2nd Edition

Semester 6-1

a. **Course:** Machine Learning

b. **Course Code:** 303105353

c. **Prerequisite:** Data structure, automata, and languages, Mathematics, Python.

d. **Rationale:**This course provides a broad introduction to Artificial Intelligence. AI techniques for search and knowledge representation also apply knowledge of AI planning and machine learning techniques to real-world problems.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the foundational concepts of machine learning including learning paradigms, probability theory, and the preprocessing of real-world data.
CLOBJ 2	Apply supervised learning algorithms such as Linear/Multilinear Regression, Naïve Bayes, Decision Trees, K-NN, Logistic Regression, Perceptrons, and SVM for classification and prediction tasks.
CLOBJ 3	Analyze and implement unsupervised learning techniques including K-Means, Hierarchical Clustering, PCA, Expectation Maximization, and Self-Organizing Maps.
CLOBJ 4	Evaluate the performance of machine learning models using appropriate metrics such as ROC curves, significance testing, and error analysis.
CLOBJ 5	Utilize ensemble learning techniques such as Bagging, Boosting, Random Forests, AdaBoost, and XGBoost for improving model accuracy and robustness.

f. **Course Learning Outcome:**

CLO 1	Discover the basic issues and challenges in Machine Learning including data and model selection and its complexity.
CLO 2	Understand the underlying mathematical relations within and across Machine Learning algorithms.
CLO 3	Assess the different Supervised Learning algorithms using a suitable Dataset.
CLO 4	Evaluate the different unsupervised Learning algorithms using a suitable Dataset.
CLO 5	Design and implement different machine learning algorithms in a range of real-world applications.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction:** Introduction to Machine Learning, Learning Paradigms, PAC learning, Basics of Probability, Version Spaces, Machine Learning in Practice: Data collection, Preprocessing (Missing values, Normalization, Adopting to chosen algorithm etc.,), Outlier Analysis (Z-Score), Model selection & evaluation, Optimization of tuning parameters, Setting the environment, Visualization of results.	20%	9
2	**Supervised Learning - 1:** Linear and Non-Linear examples, Multi–Class & Multi-Label classification, Linear Regression, Multilinear Regression, Naïve Bayes Classifier, Decision Trees, ID3, CART, Error bounds	20%	8
3	**Supervised Learning - 2:** K-NN classifier, Logistic regression, Perceptrons, Single layer & Multi-layer, Support Vector Machines, Linear & Non-linear, Semi Supervised Learning	20%	9
4	**Unsupervised Learning:** Clustering basics (Partitioned, Hierarchical and Density based), K-Means clustering, K-Mode clustering, Self organizing maps, Expectation maximization, Principal Component Analysis, Reinforcement Learning	20%	8
5	**Evaluation Metrics:** ROC Curves, Evaluation Metrics, Significance tests, Error correction in Perceptrons.	10%	6
6	**Ensemble Learning:** Bagging and Boosting, Random forests, Adaboost, XG boost inclusive.	6%	5

i. Reference Books:

1. **Real-World Machine Learning** By Henrik Brink, Joseph Richards, Mark Fetherolf — DreamTech (Textbook)
2. **Pattern Recognition and Machine Learning** By Christopher M. Bishop
3. **Elements of Statistical Learning** By Hastie, Tibshirani, and Friedman — Soft Computing for Problem Solving, AISC , Springer
4. **Data Mining: Tools and Techniques** By Jiawei Han and Michelline Kamber
5. **Data Mining: A practical Machine Learning Tools and techniques** By I H Witten, Eibe Frank, Mark A Hall — Elsevier

j. List of Practicals:

1. Dealing with Data using Numpy, Pandas, Statistics library.
2. Data Analysis & Visualization on Diwali Sales Dataset.
3. Implement linear regression and logistic regression.
4. Implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering a few test data sets.
5. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task.
6. Decision tree-based ID3 algorithm.
7. Write a program to implement the K-Nearest Neighbor algorithm to classify the iris data set.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm.
9. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set.
10. Compare the various supervised learning algorithm by using appropriate dataset. (Linear Regression, Support Vector Machine, Decision Tree)
11. Compare the various Unsupervised learning algorithm by using the appropriate datasets. (K Means Clustering, K Mode)
12. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets

Semester 6-2

a. **Course:** Compiler Design

b. **Course Code:** 3303105349

c. **Prerequisite:** Algorithms, Data Structures, Assembly Language Program, Theory of Computation, C/C++ Programming Skills

d. **Rationale:** Compiler Design is a fundamental subject of Computer Engineering. Compiler design principles provide an in-depth view of translation, optimization and compilation of the entire source program. It also focuses on various designs of compiler and structuring of various phases of compiler. It is inevitable to grasp the knowledge of various types of grammar, lexical analysis, acc, FSM(Finite State Machines) and correlative concepts of languages.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the structure and functionality of a compiler, including lexical analysis using tools like LEX and the role of finite state machines in token recognition.
CLOBJ 2	Apply context-free grammars to syntax analysis and implement top-down and bottom-up parsing techniques such as LL(1), LR(0), SLR(1), and LALR(1), including use of tools like YACC.
CLOBJ 3	Design and evaluate syntax-directed definitions and attribute grammars (S- and L-attributed) for semantic analysis of programming constructs.
CLOBJ 4	Generate intermediate representations such as quadruples, triples, trees, and flow graphs, and translate various control and data flow statements using backpatching techniques.
CLOBJ 5	Analyze runtime environments including stack-based memory allocation, activation records, and non-local data access for procedure handling.

f. **Course Learning Outcome:**

CLO 1	Understand the basic concepts; ability to apply automata theory and knowledge on formal languages.
CLO 2	Ability to identify and select suitable parsing strategies for a compiler for various cases. Knowledge in alternative methods (top-down or bottom-up, etc.).
CLO 3	Understand backend of compiler: intermediate code, Code optimization Techniques and Error Recovery mechanisms.
CLO 4	Understand issues of run time environments and scheduling for instruction level parallelism.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Overview of compilation :** The structure of a compiler and applications of compiler technology; Lexical analysis - The role of a lexical analyzer, specification of tokens, recognition of tokens, hand-written lexical analyzers, LEX, examples of LEX programs.	10%	8
2	**Introduction to syntax analysis:** Role of a parser, use of context-free grammars (CFG) in the specification of the syntax of programming languages, techniques for writing grammars for programming languages (removal left recursion, etc.), non-context-free constructs in programming languages, parse trees and ambiguity, examples of programming language grammars.	10%	7
3	**Top-down parsing:** FIRST & FOLLOW sets, LL(1) conditions, predictive parsing, recursive descent parsing, error recovery. LR-parsing - Handle pruning, shift-reduce parsing, viable prefixes, valid items, LR(0) automaton, LR-parsing algorithm, SLR(1), LR(1), and LALR(1) parsing. YACC, error recovery with YACC and examples of YACC specifications.	20%	7
4	**Syntax-directed definitions (attribute grammars):** Synthesized and inherited attributes, examples of SDDs, evaluation orders for attributes of an SDD, dependency graphs. S-ttributed and L-attributed SDDs and their implementation using LR-parsers and recursive-descent parsers respectively.	15%	6

5	**Semantic analysis:** Symbol tables and their data structures. Representation of "scope". Semantic analysis of expressions, assignment and control-flow statements, declarations of variables and functions, function calls, etc., using S- and L-attributed SDDs (treatment of arrays and structures included). Semantic error recovery.	15%	6
6	**Intermediate code generation:** Different intermediate representations –quadruples, triples, trees, flow graphs, SSA forms, and their uses. Translation of expressions (including array references with subscripts) and assignment statements. Translation of control-flow statements – it- then-else, while-do, and switch. Short-circuit code and control-flow translation of Boolean expressions. Back patching. Examples to illustrate intermediate code generation for all constructs.	15%	6
7	**Run-time environments:** Stack allocation of space and activation records. Access to non-local data on the stack in the case of procedures with and without nesting of procedures.	10%	3
8	**Introduction to machine code generation and optimization:** Simple machine code generation, examples of machine-independent code optimizations.	5%	2

i. Reference Books:

1. **Compilers: Principles, Techniques and Tools** By Aho, Lam, Sethi, and Ullman — Pearson — Second, Pub. Year 2014

j. List of Practicals:

1. Program to implement Lexical Analyzer.
2. Program to count digits, vowels and symbols in C.
3. Program to check validation of User Name and Password in C.
4. Program to implement Predictive Parsing LL (1) in C.
5. Program to implement Recursive Descent Parsing in C.
6. Program to implement Operator Precedence Parsing in C.
7. Program to implement LALR Parsing in C.
8. To Study about Lexical Analyzer Generator (LEX) and Flex (Fast LexicalAnalyzer)
9. Implement following programs using Lex.
    1. Create a Lexer to take input from text file and count no of characters, no. of lines & no. of words.

2. Write a Lex program to count number of vowels and consonants in a given input string.

10. Implement following programs using Lex.

1. Write a Lex program to print all numbers from the given file.
2. Write a Lex program to print all HTML tags in the file.
3. Write a Lex program which adds line numbers to the given file and displays the same onto the standard output.

Semester 6-3

a. **Course:** High Performance Computing

b. **Course Code:** 303105355

c. **Prerequisite:** Parallel Computing, Computer Architecture , 303105210 - Computer Organization and Microprocessor

d. **Rationale:**HPC is more than just for achieving high performance - it is a compelling vision for how computation can seamlessly scale from a single processor to virtually limitless computing power. The market demands general-purpose processors that deliver high single threaded performance as well as multi-core throughput for a wide variety of workloads on client, server, and high-performance computing (HPC) systems.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the fundamentals of parallel computing, levels of parallelism, and various parallel computing models such as SIMD, MIMD, and SPMD.
CLOBJ 2	Design efficient parallel algorithms using decomposition, task mapping, load balancing, and interaction overhead minimization techniques.
CLOBJ 3	Implement parallel programs using the message-passing paradigm, focusing on concepts like MPI operations, communication topologies, and collective computations.
CLOBJ 4	Apply multithreading concepts using POSIX threads and OpenMP for synchronization, job partitioning, and thread management in parallel environments.
CLOBJ 5	Utilize GPGPU and CUDA programming to develop high-performance applications using the GPU memory hierarchy and DGX architecture.

f. **Course Learning Outcome:**

CLO 1	To understand the basic parallel programming concepts & GPU.
CLO 2	To comprehend the various High Performance Computing Paradigms.
CLO 3	To design parallel algorithms.
CLO 4	To demonstrate the Message passing and synchronization concepts.
CLO 5	To Perform basic CUDA Programming

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction to Parallel Computing:** Motivating Parallelism, Scope of Parallel Computing, Organization and Contents of the Text, Parallel Programming Platforms: Implicit Parallelism: Trends in Microprocessor & Architectures, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines Levels of parallelism (instruction, transaction, task, thread, memory, function) Models (SIMD, MIMD, SIMT, SPMD, Dataow Models, Demand-driven Computation) Architectures: N-wide superscalar architectures, multi-core, multithreaded.	20%	10
2	**Parallel Algorithm Design:** Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models.	15%	8
3	**Programming Using the Message-Passing Paradigm:** Principles of Message- Passing Programming, The Building Blocks: Send and Receive Operations, MPI: the Message Passing Interface, Topology and Embedding, Overlapping Communication with Computation, Collective communication, and Computation Operations	15%	8

4	**Synchronization:** Scheduling, Job Allocation, Job Partitioning, Dependency Analysis Mapping Parallel Algorithms onto Parallel Architectures, Thread Basics, The POSIX Thread API, Thread Basics: Creation and Termination, Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs, Tips for Designing Asynchronous Programs, OpenMP: a Standard for Directive Based Parallel Programming	20%	9
5	**Parallel Programming Using GPGPU:** An Overview of GPGPU, DGX architecture, An Overview of GPGPU Programming, An Overview of GPGPU Memory Hierarchy Features, CUDA Programming	15%	5
6	**Performance measures:** Speedup, efficiency, and scalability. Abstract performance metrics (work, critical paths), Amdahl's Law, abstract vs. Real performance (granularity, scalability)	15%	5

i. Reference Books:

1. **Introduction to Parallel Computing** By Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar — Pearson Publication (Textbook)
2. **Advanced Computer Architecture: Parallelism, Scalability, Programmability** By Kai Hwang, Naresh Jotwani, McGraw Hill,Second Edition, 2010
3. **CUDA by Example An Introduction to General Purpose GPU Programming** By Edward Kandrot and Jason Sanders, Addison- Wesley Professional, 2010

j. List of Practicals:

1. Study the facilities provided by Google Colab.
2. Demonstrate basic Linux Commands.
3. Using Divide and Conquer Strategies design a class for Concurrent Quick Sort using C++.
4. Demonstrate MPI functions through a simple program.
5. Write a program to check task distribution using Gprof.
6. Write a simple CUDA program to print "Hello World!"
7. Write a simple CUDA program to add two numbers.
8. Write a CUDA program to add two arrays.
9. Analyze the code using Nvidia-Profilers.
10. Demonstration of OpenMP and pthread functions.

Semester 6-4

a. **Course:** Cyber Security

b. **Course Code:** 303105341

c. **Prerequisite:** Fundamental of Programming, Computer Network

d. **Rationale:** Cyber security is the application of technologies, processes, and controls to protect systems, networks, programs,devices, and data from cyber-attacks. It aims to reduce the risk of cyber-attacks and protect against the unauthorized exploitation of systems, networks, and technologies.

e. **Course Learning Objectives:**

CLOBJ 1	Understand the fundamental concepts of information security, including threats, risk analysis, and the importance of securing information systems and networks.
CLOBJ 2	Apply vulnerability scanning techniques using tools like Nmap, Netcat, and Wireshark to identify system and network vulnerabilities.
CLOBJ 3	Analyze the role and configuration of network defense tools such as firewalls, intrusion detection systems, and VPNs in securing digital environments.
CLOBJ 4	Explain various types of cybercrimes, associated legal frameworks (e.g., IT Act 2000), and the process of digital forensic investigation.
CLOBJ 5	Investigate and evaluate cybersecurity attacks including malware, SQL injection, DoS/DDoS, buffer overflow, and wireless attacks, and apply countermeasures effectively.

f. **Course Learning Outcome:**

CLO 1	Explain the features and characteristics of the Linux Operating System and Windows Operating System.
CLO 2	Apply network monitoring tools to identify attacks against network protocols and services.
CLO 3	Apply various methods to prevent malicious access to computer networks, hosts, and data.
CLO 4	Explain how to investigate endpoint vulnerabilities and attacks.
CLO 5	Analyze network intrusion data to verify potential exploits.
CLO 6	Apply incident response models to manage network security incidents

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Information Security:** Introduction to information system, Types of information Systems, Development of Information Systems, Introduction to Information Security, Need for Information Security, Threats to Information Systems, Information Assurance, Cyber Security and Security Risk Analysis.	15%	8
2	**Systems Vulnerability Scanning:** Overview of vulnerability scanning, Open Port/Service Identification, Banner/ Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples. Networks Vulnerability Scanning - Netcat, Understanding Port and Services tools, Network Reconnaissance–Nmap. Network Sniffers and Injection tools–Wireshark.	25%	13
3	**Network Defense tools:** Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation(NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System	20%	11

4	**Introduction to Cyber Crime and law:** Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics ,Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian ITACT 2000.	20%	7
5	**Introduction to Cyber Crime Investigation:** Firewalls and Packet Filters, password Cracking, Key loggers and Spyware, Virus And Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks.	20%	6

i. Reference Books:

1. **Cryptography and Network Security** By William Stallings — Pearson Education (TextBook)
2. **Cryptography and Network Security** By V.K. Jain — Khanna Publishing House (TextBook)
3. **Information and Cyber Security** By Gupta Sarika — Khanna Publishing House (TextBook)
4. **Cryptography and Network Security** By Atul Kahate, TMH (TextBook)
5. **Cryptography and Information Security** By V.K. Pachghare — PHI Learning (TextBook)
6. **Anti-Hacker Tool Kit** By Mike Shema — McGrawHill
7. **Cyber Security understanding Cyber Crimes, Computer forensics and Legal Perspectives** By Nina Godbole and Sunit Belapure ,WILEY

j. List of Practicals:

1. Implementation to gather information from any PC's connected to the LAN using whois, port scanners, network scanning, Angry IP scanners etc.
2. Experiments with open source firewall/proxy packages like iptables, squid etc.
3. Implementation of Steganography.
4. Implementation of MITM- attack using wireshark / network sniffers.
5. Implementation of Windows security using firewalls and other tools.
6. Implementation to identify web vulnerabilities, using OWASP project.
7. Implementation of IT Audit, malware analysis and vulnerability assessment and generate the report.
8. Implementation of OS hardening and RAM dump analysis to collect the Artifacts and other Information.

9. Implementation of Mobile Audit and generate the report of the existing Artifacts.
10. Implementation of Cyber Forensics tools for Disk Imaging, Data acquisition, Data extraction and Data Analysis and recovery.

Semester 6-5

a. Course: MEA(R)N Stack Web Development

b. Course Code: 303105385

c. Prerequisite: Database Management System, SQL, Basics of JavaScript and Web Development

d. Rationale:

(a) Understanding the fundamentals of JavaScript programming and web development

(b) Acquiring knowledge about how to store and retrieve data using MongoDB

(c) Acquiring knowledge about how to handle server-side logic and develop APIs using Node.js, a server-side JavaScript runtime

(d) Learning how to create web apps with Express.js

(e) Acquiring knowledge about how to connect to APIs and create dynamic user interfaces using AngularJS, a potent front-end JavaScript framework

(f) Building a full-stack web application from scratch using the MEAN stack

(g) Understanding best practices for deploying, testing, and maintaining MEAN stack applications

e. Course Learning Objectives:

CLOBJ 1	Set up and configure the MEAN stack development environment by installing and managing MongoDB, Express.js, Angular, and Node.js.
CLOBJ 2	Develop and manage databases using MongoDB by performing CRUD operations, creating indexes, and designing schemas for structured data storage.
CLOBJ 3	Build server-side web applications using Node.js and Express.js, including API development, authentication, middleware handling, and security implementation.
CLOBJ 4	Create dynamic front-end applications using Angular, with a focus on components, routing, data binding, forms, and HTTP communication.
CLOBJ 5	Integrate, deploy, and optimize full-stack MEAN applications by connecting front-end and back-end systems, implementing real-time data features, and using cloud deployment practices.

f. Course Learning Outcome:

CLO 1	Set up and configure the MEAN stack by installing and managing MongoDB, Express.js, Angular, and Node.js for web development.
CLO 2	Perform database operations by implementing CRUD functions, indexing, and schema design in MongoDB.
CLO 3	Develop server-side applications using Node.js and Express.js with authentication, security, and middleware handling.
CLO 4	Build dynamic front-end applications using Angular components, data binding, form validation, and HTTP communication.
CLO 5	Deploy and optimize MEAN stack applications by implementing security measures, performance enhancements, and continuous integration practices.

g. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	**Topics**	**Weightage (%)**	**Teaching Hours**
1	**Introduction to Web Development and the MEAN Stack:** Overview of web development, Introduction to the MEAN stack, Setting up the development environment	4%	2
2	**MongoDB:** Introduction to NoSQL databases, Installation and configuration of MongoDB, CRUD operations in MongoDB, Indexing and querying in MongoDB, Schema design and data modeling	20%	10

3	**Node.JS & Express JS:** Introduction to Node.js and Express.js, Middleware and routing, Authentication and security with Passport.js, Error handling and logging	20%	10
4	**Angular:** Introduction to Angular, Setting up an Angular application, Components, modules, and services, Data binding and templates, Forms and validation, Routing and navigation, HTTP and observables, Building a complete frontend for the MEAN stack application	30%	13
5	**Integration:** Integrating the Angular frontend with the Express.js API, Authentication and user management integration, Handling real-time data with WebSockets, Error handling and testing	10%	3
6	**Deployment and Best Practices:** Preparing the application for deployment, Hosting and server setup options, Security best practices, Performance optimization and testing, Version control and continuous integration	6%	3
7	**Final Project:** Project	10%	4

i. Reference Books:

1. **MEAN Web Development** By Amos Q. Haviv — Packt Publishing (Textbook)
2. **Learning Node.js: A Hands-On Guide to Building Web Applications in JavaScript** By Marc Wandschneider — Addison-Wesley Professional
3. **AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps** By Shyam Seshadri and Brad Green — O'Reilly Media
4. **MongoDB: The Definitive Guide: Powerful and Scalable Data Storage** By Shannon Bradshaw, Kristina Chodorow, and Eoin Brazil — O'Reilly Media

j. List of Practicals:

1. Introduction to MEAN stack, Setting up the development environment, Overview of MongoDB, Express.js, Angular, and Node.js.
2. Creating and configuring MongoDB, Creating and configuring Express.js, Building RESTful APIs with Express.js.
3. Introduction to Angular, Building basic UI components with Angular, Creating a Single-Page Application (SPA) with Angular.
4. Introduction to Node.js, Creating and configuring Node.js, Building server-side applications with Node.js.
5. Integrating all components to build a full-stack application, Testing and debugging the application, Deploying the application on a cloud platform.

Semester 6-6

a. Course: Employability Skills

b. Course Code: 303193353

c. Prerequisite: Basic knowledge of English communication and soft skills fundamentals.

d. Rationale: To enhance students' English proficiency, interpersonal communication, and job readiness through resume building, mock interviews, group discussions, and IELTS training.

e. Course Learning Objectives:

CLOBJ 1	Enhance English communication skills through IELTS-based listening, speaking, reading, and writing modules.
CLOBJ 2	Develop a professional resume and cover letter tailored to job applications, higher education, or scholarships.
CLOBJ 3	Strengthen interpersonal and critical thinking skills through mock group discussions on current and technical topics.
CLOBJ 4	Prepare for personal and case interviews by understanding employer expectations and frequently asked questions.
CLOBJ 5	Improve employability and global readiness through simulated workplace communication and assessment activities.

f. Course Learning Outcomes:

CLO 1	Demonstrate proficiency in IELTS-based communication skills applicable in global academic and professional settings.
CLO 2	Create an effective resume and cover letter that align with specific job profiles or academic pursuits.
CLO 3	Participate actively and confidently in structured group discussions, contributing valuable perspectives.
CLO 4	Exhibit readiness for interviews by articulating thoughts clearly, addressing employer expectations, and handling case scenarios.
CLO 5	Apply employability strategies and soft skills learned through mock assessments to real-world recruitment processes.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
-	1	-	1	-	100	-	-	-	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours(%)
1	IELTS Mock Test To develop students English Learning and improve their employment prospects. To create opportunity for students to study around the globe & give them Practice on : Listening Speaking Reading Writing	25	5
2	Resume Building Cover letter & Resume Writing Students will create a functional resume along with cover letter that they will be able to use when applying for a job, college or a scholarship.	25	2
3	Advanced Group Discussion: Mock Round To provide students with an avenue to train themselves in various interpersonal skills. To prepare students for the Group Discussion after the written test for employment or for admission to educational institutes. To generate new ideas or new approaches for solving a problem. To reach a solution on an issue of concern.	25	4
4	Personal Interview: Mock Round Preparing For The Interview Review Question Employer's Expectation Case Interview	25	4

i. Reference Books:

1. **Business Correspondence and Report Writing** by R. Sharma and K. Mohan
2. **Communication Skills and Soft Skills** by Suresh Kumar, Pearson Publication, 2010

Semester 7-1

a. Course: Deep Learning With NLP

b. Course Code: 303105479

c. Prerequisite: Basics of Python & Machine Learning

d. Rationale: Natural language processing (NLP) is a crucial part of artificial intelligence (AI). This course will introduce about the core concepts related to Natural Language Processing and its applications. A broad introduction about incorporation of Deep Learning techniques in Natural Language Processing is considered.

e. Course Learning Objectives:

CLOBJ 1	Understand the fundamental concepts of Natural Language Processing, including language structure, common processing tasks, and real-world applications.
CLOBJ 2	Explain the architecture and training processes of various Deep Learning models such as CNN, RNN, and LSTM relevant to Natural Language Processing.
CLOBJ 3	Implement and evaluate word embedding techniques including Word2Vec, CBOW, Skip-Gram, and GloVe for effective language representation.
CLOBJ 4	Design and apply Recurrent Neural Network-based models for sequence modeling tasks in NLP, including Sequence-to-Sequence models.
CLOBJ 5	Develop and analyze chatbot systems and sentiment analysis applications using deep learning techniques and modern AI tools.

f. Course Learning Outcome:

CLO 1	Understand the concepts of Natural Language Processing.
CLO 2	Understand the concepts of Deep Learning
CLO 3	Apply several models of Deep Learning in Natural Language Processing.
CLO 4	Analyze the importance of Recurrent Neural Network & LSTM in NLP.
CLO 5	Understand about recent developments in the field of Chatbots.
CLO 6	Classify and Implement Sentimental Analysis techniques.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction to Natural Language Processing:** The First Wave: Rationalism, The Second Wave: Empiricism, The Third Wave: Deep Learning, what makes Natural Language Processing Difficult? What do we want to achieve through Natural Language Processing? Common Terms Associated with Language Processing, Introduction to basics of NLP operations: Word Level Analysis, Syntactic Analysis, Semantic Analysis, Word Sense Disambiguation, Discourse Processing, Part of Speech (PoS) Tagging, Natural Language Inception, Information Retrieval, Applications of NLP.	15%	8
2	**Introduction to Deep Learning:** Introduction to the Deep Learning, Basic Structure of Neural Network, Types of Activation Function, Types of Neural Network: Feedforward Neural Network, Convolutional Neural Network, Recurrent Neural Network, Encoder Decoder Networks, Recursive Neural Network, Multilayer Neural Network, Stochastic Gradient Descent, Backpropagation Algorithm.	15%	8
3	**Word Vector Representations:** Introduction to Word Embedding: Neural Language Model, Word2Vec: Skip-Gram Model, Model Components: Architecture, Hidden Layer, Output Layer, CBOW Model, Subsampling Frequent Words: Negative Sampling. Efficient Softmax approximations, Sampling-based approximations for Softmax, Glove, Cross-lingual word embedding models	20%	9

4	**Recurrent Neural Network for NLP:** What is RNN? Recurrence, Difference between Feedforward and Recurrent Neural Networks, Basics of Recurrent Neural Networks, Natural Language Processing & Recurrent Neural Networks, RNN's Mechanism, Training RNN, Meta meaning of Hidden State of RNN. Tuning RNN, Long Short-Term Memory Networks, Sequence to Sequence Models, Advance Sequence to Sequence Models.	20%	8
5	**Chatbot Development:** Introduction to the Chatbot, Origin of Chatbot, How does a chatbot work? Why are chatbots such a big opportunity, Conversational bot, Chatbot: Automatic Text Generation procedure, Introduction to ChatGPT AI Chatbot	10%	4
6	**Deep Learning in Sentiment Analysis:** Introduction, Sentiment- Specific Word Embedding, Sentence Level Sentiment Classification: Convolution Neural Network, Recurrent Neural Network, Recursive Neural Network, Integration of External Resources, Document Level Sentiment Classification, Fine-Grained Sentiment Analysis, Opinion Mining, Targeted Sentiment Analysis, Aspect Level Sentiment Analysis, Stance Detection, Sarcasm Recognition	20%	8

i. Reference Books:

1. **Deep Learning for Natural Language Processing: Creating Neural Networks with Python** By P. Goyal, S. Pandey and K. Jain by Apress. (TextBook)
2. **Deep Learning in Natural Language Processing** By Li Deng, Yang Liu by Springer.
3. **Natural Language Processing: Develop Deep Learning Models for Natural Language in Python** By Jason Brownlee
4. **Handbook of Natural Language Processing** By Nitin Indurkhya Fred J. Damerau, CRC Press.

j. List of Practicals:

1. Implementation of preprocessing of Text with NLTK (Tokenization, Stemming, Lemmatization and removal of stop words in NLP
2. Implementation to Convert the text to word count vectors with ScikitLearn (CountVectorizer).
3. Implementation to Convert the text to word frequency vectors with ScikitLearn (TfidfVectorizer).
4. Implementation to Convert the text to unique integers with ScikitLearn (HashingVectorizer).
5. Use the Keras deep learning library and split words with (text_to_word_sequence).

6. Use the Keras deep learning library and write a code for encoding with (one_hot).
7. Use the Keras deep learning library and write a code for Hash Encoding with (hashing_trick).
8. Use the Keras deep learning library give a demo of Tokenizer API.
9. Perform an experiment to do sentimental analysis on any of the dataset (like twitter tweets, movie review etc.).
10. Perform an experiment using Gensim python library for Word2Vec Embedding.

Semester 7-2

a. Course: Pattern Recognition

b. Course Code: 303105481

c. Prerequisite: Basic Knowledge of data mining and mathematics

d. Rationale: The pattern recognition is used in many application areas like image analysis, speed and audio recognition, biometrics, bioinformatics, data mining and information retrieval. The syllabus introduces the concepts and principles of pattern recognition and covers the classifiers including linear and nonlinear, clustering, feature selection and generation, template matching and context dependent classification.

e. Course Learning Objectives:

CLOBJ 1	Understand the fundamental concepts and design principles of pattern recognition systems, including statistical and probabilistic approaches.
CLOBJ 2	Apply techniques for feature extraction and dimensionality reduction such as PCA and Fisher Discriminant Analysis in real-world datasets.
CLOBJ 3	Implement various classification models including decision trees, k-NN, perceptron, SVMs, and ensemble methods like Bagging and AdaBoost.
CLOBJ 4	Analyze the working of artificial neural networks and kernel-based methods for classification and regression tasks.
CLOBJ 5	Evaluate the applications of pattern recognition techniques in domains such as speech recognition, character recognition, and scene analysis.

f. Course Learning Outcome:

CLO 1	Summarize the various techniques involved in pattern recognition.
CLO 2	Categorize the various pattern recognition techniques into supervised and unsupervised.
CLO 3	Illustrate the artificial neural network-based pattern recognition.
CLO 4	Implement the applications of pattern recognition in various applications.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction:** Basics of pattern recognition system, various applications, Machine Perception, classification of pattern recognition systems. Design of Pattern recognition system, Pattern recognition Life Cycle. Statistical Pattern Recognition: Review of probability theory, Gaussian distribution, Bayes decision theory and Classifiers, Optimal solutions for minimum error and minimum risk criteria, Normal density and discriminant functions, Decision surfaces, Parameter estimation methods: Maximum-Likelihood estimation, Expectation-maximization method, Bayesian parameter estimation.	15%	9
2	**Concept of feature extraction and dimensionality:** Curse of dimensionality, Dimension reduction methods - Fisher discriminant analysis, Principal component analysis, Hidden Markov Models (HMM) basic concepts, Gaussian mixture models.	10%	5
3	**Non-Parameter methods:** Non-parametric techniques for density estimation - Parzen-window method, K-Nearest Neighbour method, non-metric methods for pattern classification: non-numeric data or 15% nominal data Decision trees: Concept of construction, splitting of nodes, choosing of attributes, overfitting, pruning	15%	5
4	**Non-metric methods for pattern classification:** Non-numeric data or 15% nominal data Decision trees: Concept of construction, splitting of nodes, choosing of attributes, overfitting, pruning, Linear Discriminant based algorithm: Perceptron, Support Vector Machines	10%	5

5	**Artificial Neural Networks for Classification and regression:** Multilayer perceptron, Back propagation algorithm, Representational abilities of feedforward networks, Artificial Neural networks	15%	6
6	**Classifier Ensembles:** Bagging/AdaBoost, Risk minimization view of AdaBoost, Unsupervised learning: Clustering-Criterion functions for clustering, Algorithm for clustering: K-mean and Hierarchical methods cluster validation	10%	5
7	**Support Vector Machines and Kernel based methods:** Support Vector Machines – Introduction, obtaining the optimal hyperplane - SVM formulation with slack variables, nonlinear SVM classifiers, Kernel Functions for nonlinear SVMs; Mercer and positive definite Kernels, Support Vector Regression and & epsilon-insensitive Loss function, examples of SVM learning	15%	7
8	**Applications of Pattern Recognition:** Speech and speaker recognition, Character recognition, Scene analysis.	10%	3

i. Reference Books:

1. **Pattern Recognition Principles** By J.I. Tou & R.C. Gonzalez, Addition-Wesley. (TextBook)
2. **Pattern Recognition - Statistical, Structural and Neural Approaches** By R. Schalkoff, John Wiley, 1992.
3. **Pattern Recognition - A Statistical Approach** By P.A. Devijer & J. Kittler, Prentice-Hall.
4. **Pattern recognition and machine learning** By Christopher. M. Bishop, Springer, 2006.
5. **Pattern Classification** By R.O. Duda, P.E. Hart and D.G. Stork, John Wiley, 2002.
6. **Neural Networks and Pattern Recognition** By C.M. Bishop , Oxford University Press (Indian Edition), 2003

j. List of Practicals:

1. Implementation of Gradient Descent
2. Implementation of Linear Regression using Gradient Descent
3. Comparison of Classification Accuracy of SVM for given dataset
4. Generate your own feature set by combining existing set of features, or defining new ones. Feature Representation
5. Generate samples of a normal distribution with specific parameters with respect to Mean and Covariance

6. Implement Linear Perceptron Learning algorithm
7. Build IRIS flower classification in python using pattern recognition models

Semester 7-3

a. Course: Cyber Physical Systems

b. Course Code:

c. Prerequisite:

d. Rationale:

e. Course Learning Objectives:

CLOBJ 1	Understand the foundational concepts of the Internet of Things (IoT), including embedded systems, M2M communication, cloud computing, and fog computing.
CLOBJ 2	Analyze the architecture of IoT systems, including layered designs, communication protocols, and integration of microcontroller platforms such as Arduino, Raspberry Pi, and NodeMCU.
CLOBJ 3	Implement IoT solutions using industry-relevant protocols such as Zigbee, Bluetooth, MQTT, CoAP, and 6LowPAN for effective communication in networked environments.
CLOBJ 4	Develop and program IoT hardware using C, MicroPython, and Python to perform tasks such as sensor integration, data acquisition, cloud communication, and GUI design.
CLOBJ 5	Evaluate real-world applications of Cyber Physical Systems in domains like smart cities, healthcare, industrial automation, and recognize the associated design challenges.
CLOBJ 6	Identify and address open challenges in IoT including security, interoperability, data integration, and energy-efficient computation for large-scale deployments.

f. Course Learning Outcome:

CLO 1	Understand the application areas of IOT.
CLO 2	Realize the revolution of Internet in Mobile Devices, Cloud & Sensor Networks.
CLO 3	Understand building blocks of Internet of Things and characteristics.
CLO 4	Apply and implement various communication technologies of IoT in solving real-life problems.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	0	3	20	20	0	60	0	100

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Introduction to Internet of Things(IoT):** Embedded system – The core part of IoT, Application of Embedded Systems, Evolution in Computing, Machine to Machine(M2M) and IoT, What Is the Internet of Things (IoT)?, Components of IoT, Introduction to Cloud Computing, Introduction to Big data platforms for IoT, Introduction to Cloud and Fog ComputingEmbedded system – The core part of IoT, Application of Embedded Systems, Evolution in Computing, Machine to Machine(M2M) and IoT, What Is the Internet of Things (IoT)?, Components of IoT, Introduction to Cloud Computing, Introduction to Big data platforms for IoT, Introduction to Cloud and Fog Computing	10%	4
2	**IoT Architecture:** IoT Architecture layers, Three- and Five-Layer Architectures, Cloud and Fog Based Architectures **Breaking Down the IoT Stack:** Devices – Arduino, Raspberry Pi, NodeMCU(ESP8266) Sensors & Actuators - Buzzers, Relays, DC Motors, Stepper Motors, Servo Motors, Digital Sensors, Analog Sensors, Serial Communications with UART, Pulse width modulation, I2C, SPI	20%	8
3	**IoT Protocol Stack:** Zigbee, Zwave, RFID, NFC, Smart and Classic Bluetooth, Thread, CoAP, AMQP, DDS, MQTT, WiFi, LiFi , SMQTT, CoRE, 6LowPAN, 6TiSCH, RPL, CORPL, CARP	20%	9

4	**Hardware and Software:** **Arduino Board and C programming:** Arduino Platform, Arduino IDE, Compiling Code, Arduino Schematics, Arduino Basic Setup & Interface, Examples: Blink LED, Serial Print, ADC, pulse width modulation. **ESP8266 and Micropython/C:** ESP8266 platform, ESPlorer IDE, compiling code, Examples: Blink LED, Serial Print, ADC, pulse width modulation, Using ESP as Station, Uploading data on cloud, MQTT publish/subscribe, Communicating through android app **Raspberry Pi and Python:** About the board, Raspberry Pi Interfaces, Raspberry Pi vs. Arduino, Operating System Benefits, Raspberry Pi Setup, Introduction to Linux and Python, GPIO Access, Pulse width modulation, Blink LED, GUI using Tkinter, Network Programs, Client – server programs, Using Twitter API, Camera module, Servo control	30%	12
5	**IoT Case Studies:** IBM Watson, AmazonGo and SCiO. **Applications:** Efficient Waste Management in Smart Cities, A Smart Home Scenario, Shopping, Smart Healthcare systems, smart cities, IoT in industry.	5%	5
6	**Open Challenges in IoT:** Security, Scalability in Networking, Dynamic Topologies, Mobility, Reliability, Device Diversity and Interoperability, Integration of data from multiple Sources, Energy Efficiency, Bandwidth Management, Modeling and Analysis, Interfacing, Storage and computation to handle Exponential growth of data volume from Sensors, Complexity Management	15%	7

i. Reference Books:

1. **Internet of Things (A Hands-on-Approach)** By Vijay Madisetti and Arshdeep Bahga, — VPT (TextBook)
2. **Designing the Internet of Things** By A. McEwen, H. Cassimally — Wiley — 2013
3. **Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems** By Dr. Ovidiu Vermesan, Dr. Peter Friess — River Publishers
4. **Interconnecting Smart Objects with IP: The Next Internet** By Jean-Philippe Vasseur, Adam Dunkels — Morgan Kuffmann Publishers
5. **Getting Started with the Internet of Things** By Cuno Pfister, — O'Reilly Media

Semester 7-4

a. **Course:** Software Testing & Quality Assurance

b. **Course Code:** 303105377

c. **Prerequisite:**Software Engineering Basics, Basics of Java Programming

d. **Rationale:**It equips students with essential skills to ensure the reliability, functionality, and quality of software systems. It emphasizes on testing techniques, QA processes, and industry tools to prepare students for real-world software development and maintenance.

e. **Course Learning Objectives:**

CLOBJ 1	To study pioneer of Software Development Life Cycle, Development models and Agile Software development.
CLOBJ 2	To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
CLOBJ 3	To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
CLOBJ 4	To learn the process of improve the quality of software work products.
CLOBJ 5	To gain the techniques and skills on how to use modern software testing tools to support software testing projects.
CLOBJ 6	To expose Software Process Improvement and Reengineering.

f. **Course Learning Outcome:**

CLO 1	Prepare SRS (Software Requirement Specification) document and SPMP (Software Project Management Plan) document.
CLO 2	Apply the concept of Functional Oriented and Object Oriented Approach for Software Design.
CLO 3	Recognize how to ensure the quality of software product, different quality standards and software review techniques.
CLO 4	Apply various testing techniques and test plan in.
CLO 5	Able to understand modern Agile Development and Service Oriented Architecture Concept of Industry.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Software Testing:** Testing, Verification and Validation, Test Strategies for Conventional and Object-OrientedSoftware, Unit Testing, Integration Testing, Validation Testing, Alpha and Beta Testing,SystemTesting ,Recovery Testing, Security Testing, Stress Testing, Performance Testing, Metrics for Source Code, Metrics for Testing, Debugging Process, Debugging Strategies	15%	8
2	**Testing Techniques:** Software Testing Fundamentals, Black Box and White Box Testing, Basis Path Testing,Flow Graph Notation, Independent Program Paths, Graph Matrices, Control Structure Testing,Condition Testing, Data Flow Testing, Loop Testing, Graph Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis.	15%	8
3	**Object Oriented Testing Methods:** Applicability of Conventional Test Case Design Methods, Issues in Object Oriented Testing, Fault-Based Testing, Scenario-Based Testing, Random Testing and Partition Testing for Classes, Inter Class Test Case Design	20%	8
4	**Testing Process and Specialized Systems Testing:** Test Plan Development, Requirement Phase,Design Phase and Program Phase Testing, Testing Client/Server Systems, Testing Web based Systems, Testing Off the-Shelf Software, Testing in Multi-platform Environment, Testing for Real Time Systems, Testing Security.	15%	7

5	**Software Quality Assurance Concepts and Standards:** Quality Concepts, Quality Control, Qualit yAssurance, SQA Activities, Software Reviews, Formal Technical Reviews, Review Guidelines, Software Reliability, Software Safety, Quality Assurance Standards, ISO9000, ISO9001:2000, ISO9126 QualityFactors, CMM, TQM, Six Sigma, SPICE, Software Quality Assurance Metrics	20%	7
6	**Management and Change Management:** Risk Software Risks, Risk Identification, Risk Projection, Risk Refinement, The RMMM Plan, Software Configuration Management, Baselines, Software Configuration Items, SCM Process: Version Control, Change Control, Configuration Audit,Configuration Management for Web Engineering.	15%	7

i. Reference Books:

1. **Software Engineering** By R. Pressmen, 6th (TextBook)
2. **Software Testing** By Yogesh Singh, Cambridge University Press, 2012
3. **Effective Methods for Software Testing** By William Perry

j. List of Practicals:

1. Design test cases using Boundary value analysis
2. Design test cases using Equivalence class partitioning
3. Design independent paths by calculating cyclometic complexity using date problem
4. Design test cases using Decision table
5. Design independent paths by taking DD path using date problem
6. Understand The Automation Testing Approach (Theory Concept)
7. Using Selenium IDE, Write a test suite containing minimum 4 test cases
8. Install Selenium server and demonstrate it using a script in Java/PHP
9. Write and test a program to login a specific web page.
10. Write and test a program to provide total number of objects present / available on the page.
11. Write and test a program to update 10 student records into table in Excel file.

Semester 7-5

a. Course: Image Processing

b. Course Code: 303105381

c. Prerequisite: Knowledge of Fourier Transform and Digital Signal Processing

d. Rationale: This is fundamental course of computer vision. This course will strengthen fundamental knowledge about digital image processing techniques. Digital image processing is used in almost all engineering fields and wide range of applications in industrial automation, medical, agriculture, security, entertainment, education and many more.

e. Course Learning Objectives:

CLOBJ 1	Understand the fundamental concepts of digital image processing, including image acquisition, sampling, quantization, and visual perception.
CLOBJ 2	Explore various image enhancement techniques in both spatial and frequency domains to improve image quality.
CLOBJ 3	Analyze and implement filtering techniques for image restoration and noise reduction.
CLOBJ 4	Apply principles of image compression using both lossless and lossy methods and understand related standards.
CLOBJ 5	Utilize morphological operations and segmentation techniques for object detection and image analysis.
CLOBJ 6	Develop basic object recognition systems using decision-theoretic and structural pattern recognition methods.

f. Course Learning Outcome:

CLO 1	Ability to develop Orthographic and Contour maps using aerial photographs and Remote sensing Images using standard algorithms to meet design specifications.
CLO 2	Ability to develop maps using Total Station, GIS, GPS and Scanners.
CLO 3	Ability to create GIS application referencing Spatial features with Attribute data.
CLO 4	Have an understanding of the underlying mechanisms of image compression, and the ability to design systems.

g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	2	4	20	20	20	60	30	150

L- Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

h. Course Content:

Sr. No.	Topics	Weightage (%)	Teaching Hours
1	**Digital Image Fundamentals:** What Is Digital Image Processing?, The Origins of Digital Image Processing, Examples of Fields that Use Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System. Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations.	12%	5
2	**Image enhancement and filtering in spatial domain:** Background, Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters.	16%	7
3	**Image filtering in the frequency domain:** Background, Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering.	16%	7
4	**Image restoration:** A Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error(Wiener) Filtering	16%	7

5	**Image Compression:** Fundamentals, Image Compression Models, Elements of Information Theory, Error-Free Compression, Lossy Compression, Image Compression Standards.	10%	4
6	**Morphological Image Processing:** Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms.	10%	5
7	**Image Segmentation:** Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation	10%	5
8	**Object Recognition:** Patterns and Pattern Classes, Recognition Based on Decision-Theoretic Methods, Structural Methods	10%	5

i. Reference Books:

1. **Digital Image Processing** By Rafael C. Gonzalez and Richard E. Woods, Pearson Education. (TextBook)
2. **Fundamentals of Digital Image Processing** By A. K. Jain, Pearson Education
3. **Digital Image Processing** By Pratt William, John Wiley & Sons
4. **Digital Image Processing Using MATLAB** By Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, Tata McGraw Hill

j. List of Practicals:

1. To study and implement basic commands of MATLAB required for digital image processing techniques, and various image file formats.
2. To study the effect of down sampling and quantization techniques on the grayscale image.
3. Write MATLAB code to enhance the visual quality of the image using point processing techniques, namely a) Image negative, b) Logarithmic transformation, and c) Power law transformation and Gray level slicing technique.
4. Write MATLAB code to display the histogram of the image and enhance the visual quality of the image with the help of histogram equalization technique.
5. Write MATLAB code to perform the bit-plane slicing method on grayscale image.
6. Write MATLAB code to demonstrate that the convolution in spatial domain is equivalent to multiplication in the frequency domain.
7. Write MATLAB code to restore grayscale image from noisy image with the help of image restoration techniques.
8. Write MATLAB code to perform Min, Median & Max Filtering on grayscale image.

9. Write MATLAB code to blur the image using Ideal low pass, Butterworth low pass and Gaussian Low pass filter.
10. Write MATLAB code to blur the image using Ideal High pass, Butterworth High pass and Gaussian High pass filter.