

PERIYAR UNIVERSITY

State University - NAAC 'A++' Grade - NIRF Rank 94
State Public University Rank 40 - SDG Institutions Rank Band: 11-50
Salem - 636 011, Tamil Nadu, India.



DEPARTMENT OF COMPUTER SCIENCE

MASTER OF COMPUTER APPLICATIONS (MCA) (AICTE Approved)

CHOICE BASED CREDIT SYSTEM (CBCS)

OBE REGULATIONS AND SYLLABUS (Effective from the academic year 2026-2027 and thereafter)

CONTENTS

- i. PO and PSO Description
- ii. PG– Template
- iii. Methods of Evaluation &Methods of Assessment
- iv. Semester Index.
- v. Subjects–Core, Elective, Non Major, Skill Enhanced, Ability Enhanced, Extension
Activity, Environment, Professional Competency
 - 1) Course Lesson Box
 - 2) Course Objectives
 - 3) Units
 - 4) Learning Outcome
 - 5) Reference and Text Books
 - 6) Web Sources
 - 7) PO & PSO Mapping Tables

REGULATIONS ON LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK FOR POSTGRADUATE EDUCATION	
Programme	M.C.A.
Programme Code	CSC01
Duration	PG-Two Years
Programme Outcomes(Pos)	PO1:Problem Solving Skill Apply knowledge of Management theories and Human Resource practices to solve business problems through research in Global context. PO2:Decision Making Skill Foster analytical and critical thinking abilities for data-based decision making. PO3:Ethical Value Ability to incorporate quality, ethical and legal value-based perspectives to all organizational activities. PO4:Communication Skill Ability to develop communication, managerial and interpersonal skills. PO5:Individual and Team Leadership Skill Capability to lead themselves and the team to achieve organizational goals. PO6:Employability Skill Inculcate contemporary business practices to enhance employability skills in the competitive environment. PO7:Entrepreneurial Skill Equip with skills and competencies to become an entrepreneur. PO8:Contribution to Society Succeed in career endeavors and contribute significantly to society. PO9:Multicultural competence Possess knowledge of the values and beliefs of multiple cultures and a global perspective. PO10: Moral and ethical awareness /reasoning Ability to embrace moral/ethical values in conducting one's life.

Programme Specific Outcomes (PSOs)	PSO1–Placement To prepare the students who will demonstrate respectful engagement with other’s ideas, behaviors, and beliefs apply diverse frames of reference to decisions and actions. PSO2- Entrepreneur To create effective entrepreneurs by enhancing their critical thinking, problem solving, decision making and leadership skill that will facilitate startups and high potential organizations. PSO3–Research and Development Design and implement HR systems and practices grounded in research that complies with employment laws, leading the organization towards growth and development. PSO4–Contribution to Business World To produce employable, ethical and innovative professionals to sustain in the dynamic business world. PSO5–Contribution to the Society To contribute to the development of the society by collaborating with stakeholders for mutual benefit.
--	---

CURRICULUM STRUCTURE FOR EACH SEMESTER-M.C.A 2026-2027

SEMESTER-I

Course Code	Category	Course Name	Number of Credits	Hours per Week
26UPCSC1C01	Core I	Operating System with Linux Programming	4	4
26UPCSC1C02	Core II	Python Programming	4	4
26UPCSC1C03	Core III	Discrete Mathematics	4	4
26UPCSC1L01	Core IV-Lab	Operating System with Linux Programming Lab	2	4
26UPCSC1L02	Core V-Lab	Python Programming Lab	2	4
26UPCSC1L03	Core VI-Lab	Data Visualization Lab	1	2
Discipline Centric Elective-I	Elective I	Elective Subject-Theory	4	4
Generic Elective- I 26UPCSC1S01	Elective II	Soft Skill Development Lab	1	2
26UPCSC1L04	Core VII-Lab	Internet of Things Lab	1	2
Total			23	30

SEMESTER-II

Course Code	Category	Course Name	Number of Credits	Hours per Week
26UPCSC1C04	Core VIII	Data Structures and Algorithms	4	4
26UPCSC1C05	Core IX	Big Data Analytics	4	4
26UPCSC1C06	Core X	Artificial Intelligence Concepts and Techniques	4	4
26UPCSC1L05	Core XI-Lab	Data Structures and Algorithms Lab	2	4
26UPCSC1L06	Core XII-Lab	Big Data Analytics Lab	2	4
26UPCSC1P01	Core XIII	Mini Project	2	2
Discipline Centric Elective –II	Elective III	Elective Subject - Theory	4	4

-	Open Elective-I	Online Course (SWAYAM/NPTEL/MOOC, etc.)	2	1
Generic Elective- II 26UPPGC1H01	Elective-IV	Fundamentals of Human Rights	1	1
26UPCSC1L07	Core XIV	Cloud Computing with Prompt Engineering Lab	1	2
26UPCSC1X01	Core XIV	Extension Activity	1	-
Total			27	30

SEMESTER-III

Course Code	Category	Course Name	Number of Credits	Hours per Week
26UPCSC1C07	Core XV	Advanced Java Programming	4	4
26UPCSC1C08	Core XVI	Web Technologies	4	4
26UPCSC1C09	Core XVII	Advanced Machine Learning Techniques	4	4
26UPCSC1L08	Core XVIII -Lab	Advanced Java Programming Lab	2	3
26UPCSC1L09	Core XIX-Lab	Web Technologies Lab	2	3
26UPCSC1L10	Core XX -Lab	Advanced Machine Learning with AI-Assisted Coding Lab	2	4
Discipline Centric Elective V	Elective V	Elective Subject - Theory	4	4
-	Open Elective-II	Supportive Course	2	3
26UPCSC1I01	Core XXI	Internship	2	-
Generic Elective- II 26UPGEN1V01	Elective-VI	Peace Education	2	1
Total			28	30

SEMESTER-IV

Course Code	Category	Course Name	Number of Credits	Hours per Week
26UPCSC1P02	Core XXII	Major Project with Viva Voce	16	-
Total			16	-
Overall Credits			94	-

LIST OF ELECTIVE SUBJECTS

GENERIC ELECTIVE SUBJECTS

26UPCSC1S01 - Soft Skill Development Lab

DISCIPLINE CENTRIC ELECTIVE SUBJECTS

VERTICAL I: Artificial Intelligence and Data Science (AIDS)

26UPCSC1E01 - Generative AI
26UPCSC1E02 - Advanced Large Language Models
26UPCSC1E03 - Agentic AI

VERTICAL II: Full Stack Development for IT

26UPCSC1E04 - App Development
26UPCSC1E05 - UI and UX Design
26UPCSC1E06 - Software Testing and Automation

VERTICAL III: Cloud Computing and Data Centre Technologies

26UPCSC1E07 - Cloud Computing
26UPCSC1E08 - Virtualization
26UPCSC1E09 - Cloud Security

VERTICAL IV: Cyber Security and Data Privacy

26UPCSC1E10 - Network Security and Ethical Hacking
26UPCSC1E11 - Digital and Mobile Forensics
26UPCSC1E12 - Modern Cryptography and Blockchain Technologies

Regulation on Discipline-Centric Elective Verticals

Students shall select **one vertical** from the prescribed Discipline-Centric Elective Subject Verticals and complete all the courses offered under the chosen vertical. The selection of the vertical shall be **final**, and students shall **not be permitted to change** to another vertical at any stage during the course of study.

NONMAJOR ELECTIVE -II

26UPCSC1N01	-	Advanced Microsoft Office Lab
26UPCSC1N02	-	Bio Python Programming Lab

VALUE ADDED / ADD ON COURSES

26UPCSCVA01	-	MS-Office (Excel)
26UPCSCVA02	-	R Programming
26UPCSCVA03	-	Bio-python
26UPCSCVA04	-	Large Language Models (LLMs)
26UPCSCVA05	-	Basics of Computer Hardware & Introduction to Networking

DISTRIBUTION OF CREDITS M.C.A. (R2026)

Type of Courses	Component	Credits per Course	No. of Courses	Total Credits
Core	Theory Courses	4	6	24
	Theory Courses (Advanced / Core)	4	3	12
	Lab Courses	2	7	14
	Lab Courses	1	4	4
	Mini Project	2	1	2
	Major Project with Viva Voce	16	1	16
Elective	Discipline-Centric – Theory	4	3	12
	Generic – Soft Skill Development Lab	1	1	1
	Generic – Fundamentals of Human Rights	1	1	1
	Generic – Peace Education	2	1	2
Open Elective	Online Course (SWAYAM/NPTEL/MOOC)	2	1	2
	Support Course	2	1	2
Internship	Internship / Industrial Training	2	1	2
	Total Credits			94

**Choice Based Credit System (CBCS), Learning Outcomes Based Curriculum Framework (LOCF) Guideline Based Credits and Hours Distribution System
For M. C. A. Courses including Lab Hours**

I. Semester-wise Credit Distribution

Semester	Core Theory	Core Lab	Projects	Electives	Open Elective	Internship	Total Credits
Semester I	12	6	–	4	–	–	22
Semester II	12	5	2	4	2	–	25
Semester III	12	6	–	4	2	2	26
Semester IV	–	–	16	–	–	–	16
Total	36	17	18	12	4	2	94

II. Category-wise Credit Distribution

Type of Courses	Component	Credits per Course	No. of Courses	Total Credits
Core	Theory Courses	4	9	36
	Lab Courses	2	7	14
	Lab Courses	1	4	4
	Mini Project	2	1	2
	Major Project with Viva Voce	16	1	16
Elective	Discipline-Centric – Theory Courses	4	3	12
	Generic – Soft Skill Development Lab	1	1	1
	Generic – Fundamentals of Human Rights	1	1	1
	Generic – Peace Education	2	1	2
Open Elective	Online Course (SWAYAM/NPTEL/MOOC)	2	1	2
	Supportive Course	2	1	2
Internship	Internship / Industrial Training	2	1	2
	Total Credits			94

III. Credit Distribution by LOCF Component

LOCF Dimension	Academic Component	Credits	Percentage (%)	Academic Purpose
Knowledge Foundation	Core Theory Courses	36	38.30%	Conceptual depth in Computing, Mathematics, AI, Systems, and Algorithms
Skill & Technical Competency	Core Laboratory Courses	18	19.15%	Practical proficiency, programming expertise, tool-based learning
Research & Professional Competency	Mini Project (2) + Major Project (16)	18	19.15%	Research capability, innovation, and problem-solving
Disciplinary Breadth	Discipline-Centric Electives	12	12.77%	Advanced and specialized domain knowledge
Interdisciplinary & Value-Based Learning	Generic Electives	4	4.26%	Ethics, communication, and social awareness
Interdisciplinary Exposure	Open Electives	4	4.26%	Cross-disciplinary flexibility
Experiential Learning	Internship	2	2.13%	Industry readiness and real-world exposure
	Total	94	100%	

IV LOCF Outcome Mapping Perspective

LOCF Focus Area	Programme Implementation Mechanism
Disciplinary Knowledge	9 Core Theory Courses covering computing fundamentals, Artificial Intelligence, data systems, algorithms, and mathematical foundations
Practical & Analytical Skills	11 Laboratory Courses (7 Core Labs + 4 Skill Labs) emphasizing programming, data analysis, cloud computing, and AI-assisted coding practices
Research & Innovation	Mini Project (2 Credits) and Capstone Major Project (16 Credits) focusing on real-world problem-solving, research methodology, and innovation
Professional Skills	Mini Project, Internship/Industrial Training, and Soft Skill Development Lab fostering teamwork, communication, leadership, and project management
Ethical & Social Responsibility	Courses such as Fundamentals of Human Rights and Peace Education addressing ethics, inclusivity, sustainability, and societal impact of technology
Lifelong Learning	Integration of SWAYAM/NPTEL/MOOC courses enabling self-directed, continuous, and flexible learning pathways
Industry Orientation	Internship combined with application-oriented labs (Cloud Computing, AI, ML) aligned with real-world industrial practices and tools

V. Consolidated LOCF Classification Matrix

Sem	Course Name	Credits	KF	SD	E	Primary Orientation
I	Operating System with Linux Programming	4	✓	✓	✓	System Programming
I	Python Programming	4	✓	✓	✓	Application Development
I	Discrete Mathematics	4	✓	–	–	Mathematical Foundation
I	Operating System with Linux Lab	2	–	✓	✓	Practical Implementation
I	Python Programming Lab	2	–	✓	✓	Coding Skill
I	Data Visualization Lab	1	–	✓	✓	Analytical Skill
I	Discipline-Centric Elective I	4	✓	✓	✓	Specialization
I	Soft Skill Development Lab	1	–	✓	✓	Communication Skills
I	Internet of Things Lab	1	–	✓	✓	Embedded & IoT Systems
II	Data Structures and Algorithms	4	✓	✓	✓	Core Technical Competency
II	Big Data Analytics	4	✓	✓	✓	Industry Analytics
II	Artificial Intelligence Concepts and Techniques	4	✓	✓	✓	Intelligent Systems
II	Data Structures and Algorithms Lab	2	–	✓	✓	Implementation Skill
II	Big Data Analytics Lab	2	–	✓	✓	Tool-Based Analytics
II	Mini Project	2	–	✓	✓	Problem Solving
II	Discipline-Centric Elective II	4	✓	✓	✓	Domain Expertise
II	Online MOOC	2	✓	✓	✓	Lifelong Learning
II	Fundamentals of Human Rights	1	✓	–	✓	Ethical Awareness
II	Cloud Computing with Prompt Engineering Lab	1	–	✓	✓	Generative AI & Cloud Skills
III	Advanced Java Programming	4	✓	✓	✓	Enterprise Development
III	Web Technologies	4	✓	✓	✓	Full Stack Development
III	Advanced Machine Learning Techniques	4	✓	✓	✓	AI/ML Competency
III	Advanced Java Programming Lab	2	–	✓	✓	Coding Skill
III	Web Technologies Lab	2	–	✓	✓	Deployment Skill
III	Advanced Machine Learning with AI-Assisted Coding Lab	2	–	✓	✓	AI-Integrated Model Development
III	Discipline-Centric Elective III	4	✓	✓	✓	Advanced Specialization
III	Support Course	2	✓	✓	✓	Interdisciplinary Learning
III	Internship / Industrial Training	2	–	✓	✓	Industry Exposure

III	Peace Education	2	✓	–	✓	Value Orientation
IV	Major Project with Viva Voce	16	–	✓	✓	Research & Innovation

Note: KF – Knowledge Foundation, SD – Skill Development, E – Employability Enhancement

VI. Consolidated LOCF Classification Matrix (Value-Added Courses)

Course Name	Credits	KF	SD	E	Primary Orientation	Skills Acquired	Job Roles / Career Paths
MS-Office (Excel)	1	✓	✓	–	Data Handling & Office Automation	Data entry, spreadsheet analysis, data visualization, reporting	Data Entry Operator, Office Assistant, MIS Executive
R Programming	1	✓	✓	✓	Statistical Computing & Data Analytics	Statistical modeling, data analysis, visualization, scripting	Data Analyst, Junior Data Scientist, Research Assistant
Bio-Python	1	✓	✓	✓	Computational Biology & Scientific Programming	Python programming, bioinformatics analysis, data processing	Bioinformatics Analyst, Research Associate, Data Analyst
Large Language Models (LLMs)	1	✓	✓	✓	AI & Generative Models	Prompt engineering, NLP, model fine-tuning basics, AI application development	AI Engineer, NLP Engineer, Generative AI Specialist
Basics of Computer Hardware & Introduction to Networking	1	✓	✓	–	System Fundamentals & Networking	Hardware troubleshooting, network configuration, system maintenance	Hardware Technician, Network Support Engineer, System Administrator

Note: KF – Knowledge Foundation, SD – Skill Development, E – Employability Enhancement

METHODS OF EVALUATION		
Internal Evaluation	Continuous Internal Assessment Test	25 Marks
	Assignments/Snap Test/Quiz	
	Seminars	
	Attendance and Class Participation	
External Evaluation	End Semester Examination	75 Marks
Total		100 Marks

METHODS OF ASSESSMENT	
Remembering (K1)	- The lowest level of questions require students to recall information from the course content - Knowledge questions usually require students to identify Information in the text book.
Understanding (K2)	- Understanding off acts and ideas by comprehending organizing, comparing, translating, interpolating and interpreting in their own words. - The questions go beyond imply recall and require Students to combine together.
Application (K3)	- Students have to solve problems by using/applying a concept learned in the class room. - Students must use their knowledge to determine an exact response.
Analyze(K4)	- Analyzing the question is one that asks the students to break down something in to its component parts. - Analyzing requires students to identify reasons causes Or motives and reach conclusions or generalizations.
Evaluate(K5)	- Evaluation requires an individual to make judgment to something. - Questions to be asked to judge the value of an idea, a character, a work of art, or a solution to a problem. - Students are engaged in decision-making and problem– solving. - Evaluation questions do not have single right answers.
Create(K6)	- The questions of this category challenge students to get engaged in creative and original thinking. - Developing original ideas and problem solving skills.

PROGRAMME OUTCOMES (PO) – PROGRAMME SPECIFIC OUTCOMES (PSO) MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
PSO1	H	H	H	H	H	H	H	H	H	H
PSO2	H	H	H	H	H	H	H	H	H	H
PSO3	H	H	H	H	H	H	H	H	H	H
PSO4	H	H	H	H	H	H	H	H	H	H
PSO5	H	H	H	H	H	H	H	H	H	H

LEVEL OF CORRELATION BETWEEN PO'S AND PSO'S

(Suggested by UGC as per Six Sigma Tool – Cause and Effect Matrix)

Assign the value: 1–Low; 2–Medium; 3–High; 0–No Correlation

CANDIDATE ELIGIBILITY FOR M.C.A. PROGRAMME ADMISSION

Candidates who have passed any one of the following or an equivalent qualification are eligible to apply:

- Candidates must have passed any recognized undergraduate degree (e.g., B.E./B.Tech./B.Sc./B.Com./B.A./B.Voc./BCA or equivalent), preferably with Mathematics at the 10+2 level or at the graduation level.
- Candidates must have secured a minimum of 50% marks in the qualifying examination (45% for candidates belonging to reserved categories).
- Candidates without a Mathematics background shall undergo a compulsory bridge course in Mathematics, along with additional bridge courses in Computer Science subjects, as prescribed by the respective University/Institution in accordance with its norms.

Bridge Course on Basics in Computer Science

Course duration : 30 Hours

Course Modules:

- Module 1 : Computer Organization and Architecture
- Module 2 : Data Structures and Algorithms
- Module 3 : Problem Solving Techniques
- Module 4 : Operating Systems

Module 5	:	Object oriented programming
Module 6	:	Database Management System
Module 7	:	Software Engineering
Module 8	:	Computer Networks

DURATION OF THE PROGRAMME AND MEDIUM

The programme shall be of two years duration spread over four semesters under choice based credit system. The Maximum duration to complete the course shall be two academic years after normal completion of the programme. The medium of instruction/study is English.

SCHEME OF EXAMINATION FOR EACH SEMESTER

SEMESTER-I

Course Code	Course Name	Credits	Hours/ Week	T	P	CIA	ESE	Total	Duration
26UPCSC1C01	Operating System with Linux Programming	4	4	4	–	25	75	100	3 Hours
26UPCSC1C02	Python Programming	4	4	4	–	25	75	100	3 Hours
26UPCSC1C03	Discrete Mathematics	4	4	4	–	25	75	100	3 Hours
26UPCSC1L01	OS with Linux Lab	2	4	–	4	40	60	100	3 Hours
26UPCSC1L02	Python Programming Lab	2	4	–	4	40	60	100	3 Hours
26UPCSC1L03	Data Visualization Lab	1	2	–	2	40	60	100	3 Hours
-	Discipline-Centric Elective I	4	4	4	–	25	75	100	3 Hours
26UPCSC1S01	Soft Skill Development Lab	1	2	–	2	40	60	100	3 Hours
26UPCSC1L04	Internet of Things Lab	1	2	–	2	40	60	100	3 Hours
Total		23	30					900	27 Hours

SEMESTER-II

Course Code	Course Name	Credits	Hours /Week	T	P	CIA	ESE	Total	Duration
26UPCSC1C04	Data Structures and Algorithms	4	4	4	–	25	75	100	3 Hours
26UPCSC1C05	Big Data Analytics	4	4	4	–	25	75	100	3 Hours
26UPCSC1C06	Artificial Intelligence Concepts and Techniques	4	4	4	–	25	75	100	3 Hours
26UPCSC1L05	Data Structures and Algorithms Lab	2	4	–	4	40	60	100	3 Hours
26UPCSC1L06	Big Data Analytics Lab	2	4	–	4	40	60	100	3 Hours
26UPCSC1P01	Mini Project	2	2	–	2	40	60	100	3 Hours
-	Discipline-Centric Elective II	4	4	4	–	25	75	100	3 Hours
-	Open Elective I (MOOC)	2	1	–	–	–	–	100*	-
26UPPGC1H01	Fundamentals of Human Rights	1	1	1	–	25	75	100	3 Hours
26UPCSC1L07	Cloud Computing and Prompt Engineering Lab	1	2	–	2	40	60	100	3 Hours
26UPCSC1X01	Extension Activity	1	–	–	–	40	60	100	-
Total		27						900	27 Hours

*MOOC: evaluation based on external certification

SEMESTER-III

Course Code	Course Name	Credits	Hours /Week	T	P	CIA	ESE	Total	Duration
26UPCSC1C07	Advanced Java Programming	4	4	4	–	25	75	100	3 Hours
26UPCSC1C08	Web Technologies	4	4	4	–	25	75	100	3 Hours
26UPCSC1C09	Advanced Machine Learning Techniques	4	4	4	–	25	75	100	3 Hours
26UPCSC1L08	Advanced Java Lab	2	3	–	3	40	60	100	3 Hours
26UPCSC1L09	Web Technologies Lab	2	3	–	3	40	60	100	3 Hours
26UPCSC1L10	Advanced Machine Learning with AI-Assisted Coding Lab	2	4	–	4	40	60	100	3 Hours
-	Discipline-Centric Elective III	4	4	4	–	25	75	100	3 Hours
-	Open Elective II	2	3	2	–	25	75	100	3 Hours
26UPCSC1I01	Internship	2	–	–	–	40	60	100	3 Hours
26UPGEN1V01	Peace Education	2	1	1	–	25	75	100	3 Hours
Total		28	30					1000	30 Hours

SEMESTER-IV

Course Code	Course Name	Credits	Hours	T	P	CIA	ESE	Total	Exam Duration
26UPCSC1P02	Major Project with Viva Voce	16	—	—	—	50	150	200	3 Hours
TOTAL		16	—	—	—			200	3 Hours

CREDIT CALCULATION

Method of Teaching	Hours	Credits
Lecture	1	1
Tutorial/Demonstration	1	1
Practical/Internship/Self-Learning	2/1	1

ATTAINMENT RUBRICS FOR THEORY COURSES

THEORY EXAMINATION EVALUATION OF INTERNAL ASSESSMENT

Test	:	5 Marks (Best one out of Two Tests)
Model Examination	:	5 Marks
Seminar	:	5 Marks
Assignment	:	5 Marks
Attendance	:	5 Marks

Total	:	25Marks

*****No Internal Minimum**

EVALUATION OF END SEMESTER EXAMINATIONS

QUESTION PAPER PATTERN (THEORY)

Section	Approaches	Mark Pattern	K Level	CO Coverage
A	One word (Answer all questions)	20X1 =20 (Multiple Choice Questions)	K1-K2	CO1 – CO5
B	100 to 200 words (Answer any three out of five questions)	3X5 =15 (Analytical type questions)	K4-K6	
C	500 to 1000 words	5X8 =40 (Essay Type questions)	K2-K6	

ATTAINMENT RUBRICS FOR LAB COURSES

PRACTICAL/MINI PROJECT EXAMINATION

EVALUATION OF INTERNAL ASSESSMENT

Test	:	20 Marks (Best one out of Two Tests)
Model Exam	:	20 Marks

Total	:	40 Marks

*****No Internal Minimum**

QUESTION PAPER PATTERN

Time duration : 3 Hours

Max. Marks : 60 Marks

Two Questions may be taken from the list of practical problems: 60 Marks

Distribution of the Marks

(i) Practical/Mini Project

○ Record Note Book	-	10
○ Problem Understanding	-	10
○ Implementation	-	20
○ Debugging and Modification	-	10
○ For correct output and viva	-	10

(ii) Industrial Training

○ Internal Assessment	-	40
○ Joint Viva-Voce	-	60

(Internal Examiner 30 and External Examiner 30)

(i) Dissertation

○ Internal Assessment	-	50
○ Report Evaluation by External Examiner	-	50
○ Joint Viva-Voce	-	100

(Internal Examiner 50 and External Examiner 50)

REGULATIONS FOR DISSERTATION WORK

- Students should attach themselves with well-reputed industries/companies/institutions to carry out their five-month dissertation work.
- The candidate should submit the filled-in format to the department for approval during the first week of December in the even semester.
- The review of the dissertation will be carried out periodically.
- The student should submit three copies of the dissertation work
- The students may use a PowerPoint presentation during their dissertation viva voce examination.

PASSING MINIMUM

The candidate shall be declared to have passed in the Theory/Practical/Dissertation Examination if the candidate secures:

- i. 50% marks in the ESE and
- ii. 50% in ESE and IA put together

GRADING SYSTEM

Evaluation of performance of students is based on ten-point scale grading system as given below.

Ten Point Scale			
Grade of Marks	Grade points	Letter Grade	Description
90-100	9.0-10.0	O	Outstanding
80-89	8.0-8.9	D+	Excellent
75-79	7.5-7.9	D	Distinction
70-74	7.0-7.4	A+	Very Good
60-69	6.0-6.9	A	Good
50-59	5.0-5.9	B	Average
00-49	0.0	U	Re-appear
ABSENT	0.0	AAA	ABSENT

Operating System with LINUX PROGRAMMING

Course Objectives

Upon successful completion of this course, students will be able:

1. To understand the fundamental principles, architecture, and evolution of operating systems, including Unix, Linux, and Android platforms.
2. To analyze process management, synchronization mechanisms, scheduling algorithms, and interprocess communication techniques in modern operating systems.
3. To examine and evaluate memory management strategies, virtual memory techniques, and resource allocation mechanisms.
4. To understand and assess file systems, I/O subsystems, storage management, and security mechanisms in operating systems.
5. To apply operating system concepts to real-world system design, problem-solving, and performance optimization scenarios.

UNIT – I: Introduction, Processes and Threads

Introduction – History of Operating Systems – Operating System Concepts – System Calls – Operating System Structure

Processes and Threads: Process - Threads– Event-Driven Servers – Synchronization – Interprocess Communication (IPC) –Scheduling

UNIT – II: Memory Management

Memory Management: No Memory Abstraction –Memory Abstraction – Virtual Memory – Page Replacement Algorithms – Design Issues for Paging Systems – Implementation Issues – Segmentation – Research in Memory Management

UNIT – III: File Systems

File Systems: Files– Directories– File-System Implementation – File-System Management and Optimization – Example File Systems – Research in File Systems

UNIT – IV: Input/Output and Deadlocks

Input/Output: Principles of I/O Hardware – Principles of I/O Software – I/O Software Layers – Mass Storage (Disk and SSD) – Clocks– User Interfaces (Keyboard, Mouse, Monitor) – Power Management – Research in Input/Output

Deadlocks: Resources and Resource Allocation – Introduction to Deadlocks – The Ostrich Algorithm – Deadlock Detection and Recovery – Deadlock Avoidance – Deadlock Prevention – Other Issues – Research in Deadlocks

UNIT – V: Case Study – Unix, Linux and Android

Case Study: Unix, Linux and Android – History of Unix and Linux – Overview of Linux – Processes in Linux – Memory Management in Linux – Input/Output in Linux – The Linux File System – Security in Linux – Android Architecture and Operating System Design

TEXT BOOKS:

1. Tanenbaum, A. S., & Bos, H. (2023), Modern operating systems, Pearson.

Reference Books

1. Buyya, R., et al. (2022), Content delivery networks and emerging operating systems. Springer.
2. Silberschatz, A., Galvin, P. B., & Gagne, G. (2022), Operating system concepts. Wiley.
3. Anderson, T., & Dahlin, M. (2021), Operating systems: Principles and practice, Recursive Books.
4. Mokhtar Ebrahim, Andrew Mallett, “Mastering Linux Shell Scripting”, Packt Publishing, 2nd Edition, 2018.

COURSE OUTCOMES

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

CO1:	Explain and compare the architecture and structure of operating systems, including Unix, Linux, and Android environments.	K1-K6
CO2:	Design and implement solutions for process synchronization, scheduling, and interprocess communication problems.	
CO3:	Analyze and evaluate memory management and file system techniques for efficient system performance.	
CO4:	Assess I/O management, storage systems, and security mechanisms in modern operating systems.	
CO5:	Demonstrate problem-solving skills in detecting and handling deadlocks, resource conflicts, and system-level challenges.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6-Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	H	H	L	H	L	L	M	M	M
CO2	H	H	M	L	H	L	L	M	M	M
CO3	H	H	M	L	H	L	L	M	M	H
CO4	H	H	M	L	H	L	L	M	M	M
CO5	H	H	M	L	H	L	L	M	M	M

H-High; M-Medium; L-Low

PYTHON PROGRAMMING

COURSE OBJECTIVES:

- To acquire programming skills in core Python
- To learn Strings and function
- To develop object oriented skills in Python
- To comprehend various Python Packages
- To develop web applications using Django

UNIT-I

Introduction : Fundamental ideas of Computer Science - Strings, Assignment and Comments - Numeric Data types and Character sets - Expressions - Loops and Selection Statements: Definite iteration: the for Loop -selection: if and if-else statements -Conditional iteration: the while Loop

UNIT-II

Strings and Text Files: Accessing Characters and substrings in strings - Data encryption-Strings and Number systems- String methods - Text - Lists and Dictionaries: Lists - Dictionaries -Design with Functions: A Quick review - Problem Solving with top-Down Design - Design with recursive Functions - Managing a Program's namespace - Higher-Order Functions

UNIT-III

Design with Classes: Getting inside Objects and Classes - Data-Modeling Examples - Building a New Data Structure - The Two - Dimensional Grid - Structuring Classes with Inheritance and Polymorphism-Graphical User Interfaces-The Behavior of terminal-Based programs and GUI-Based programs - Coding Simple GUI-Based programs - Windows and Window Components - Command Buttons and responding to events

UNIT-IV

Working with Python Packages: NumPy Library-Ndarray- Basic Operations - Indexing, Slicing and Iteration – Array manipulation - Pandas - The Series - The Data Frame -The Index Objects - Data Vizualization with Matplotlib- The Matplotlib Architecture -Pyplot- The Plotting Window - Adding Elements to the Chart - Line Charts - Bar Charts - Pie charts

UNIT-V

Django: Installing Django- Building an Application - Project Creation - Designing the Data Schema - Creating an administration site for models - Working with Query Sets and Managers - Retrieving Objects - Building List and Detail Views

TEXT BOOKS:

1. K.A.Lambert, “Fundamentals of Python: first programs”, Second Edition, Cengage Learning, 2018 **(Unit- I, II and III)**
2. FabioNelli, “Python Data Analytics: With Pandas, NumPy, and Matplotlib”, Second Edition, Kindle Edition, 2018 **(Unit - IV)**
3. Antonio Mele,“Django3 By Example”,Third Edition, 2020 **(Unit-V)**

COURSE OUTCOMES

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

CO1	Comprehend the programming skills in python and develop applications Using conditional branches and loop	K1-K6
CO2	Create python applications with strings and functions	
CO3	Understand and implement the Object Oriented Programming paradigm with the concept of objects and classes, Inheritance and polymorphism	
CO4	Evaluate the use of Python packages to perform numerical computations and data visualization	
CO5	Design interactive web applications using Django	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate,K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	H	M	H	M	H	H	H	H	M
CO2	H	H	H	M	H	H	H	H	H	H
CO3	H	M	H	H	M	H	M	H	H	M
CO4	H	H	H	H	H	H	H	M	H	H
CO5	H	H	H	H	H	H	H	H	H	M

H-High; M-Medium; L-Low

DISCRETE MATHEMATICS

(Theorems and Proofs are not expected)

COURSE OBJECTIVE

- To know the concepts of relations and functions
- To distinguish among different normal forms and quantifiers
- To solve recurrence relations, permutations and combinations
- To know and solve matrices, rank of matrix and characteristic equations
- To study the graphs and its types

UNIT-I

Relations- Binary relations-Operations on relations- Properties of binary relations in a set – Equivalence relations -Representation of a relation by a matrix -Representation of a relation by a digraph.

Functions-Definition and examples-Classification of functions-Composition of functions-Inverse function

UNIT-II

Mathematical Logic-Logical connectives-Well formed formulas- Truth table of well formed formula - Algebra of proposition - Quince's method- Normal forms of well formed formulas- Disjunctive normal form-Principal Disjunctive normal form-Conjunctive normal form-Principal conjunctive normal form-Rules of Inference for propositional calculus-Quantifiers- Universal Quantifiers- Existential Quantifiers

UNIT-III

Recurrence Relations- Formulation - Solving recurrence Relation by Iteration- Solving Recurrence Relations- Solving Linear Homogeneous Recurrence Relations of Order Two- Solving Linear Non homogeneous Recurrence Relations. **Permutations-** Cyclic permutation- Permutations with repetitions- Permutations of sets with indistinguishable objects – **Combinations-** Combinations with repetition.

UNIT-IV

Matrices- Special types of matrices-Determinants-Inverse of a square matrix-Cramer's rule for solving linear equations-Elementary operations-Rank of a matrix- Solving a system of linear equations- Characteristic roots and Characteristic vectors-Cayley-Hamilton Theorem-Problems

UNIT-V

Graphs -Connected Graphs -Euler Graphs- Euler line-Hamiltonian circuits and paths - planar graphs - Complete graph-Bipartite graph-Hyper cube graph-Matrix representation of graphs.

TEXT BOOK:

1. N.Chandrasekaran and M.Umaparvathi, “Discrete Mathematics”, PHI Learning Private Limited, New Delhi, 2010.

REFERENCE BOOKS

1. Kimmo Eriksson & Hillevi Gavel, “Discrete Mathematics & Discrete Models”, Student literature AB, 2015.
2. Kenneth H. Rosen, “Discrete Mathematics and applications”, McGraw Hill, 2012

COURSE OUTCOMES

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

CO1:	To understand the concepts of relations and functions distinguish among normal forms	K1–K6
CO2:	To analyze and evaluate the recurrence relations	
CO3:	To distinguish among various normal forms and predicate calculus	
CO4:	To solve and know various types of matrices	
CO5:	To evaluate and solve various types of graphs	

K1-Remember, K2-Understand, K3-Apply, K4-Analyze, K5-Evaluate, K6-Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	M	H	M	H	H	H	M	L	M
CO2	L	M	M	M	L	M	L	M	M	M
CO3	H	H	H	H	L	M	H	H	M	L
CO4	M	M	L	L	M	M	H	M	H	M
CO5	H	H	H	M	M	L	L	H	H	H

H-High; M-Medium; L-Low

Operating System with LINUX PROGRAMMING - LAB

COURSE OBJECTIVES

Upon successful completion of this laboratory course, students will be able to:

1. Develop system-level programs using Linux system calls and APIs for process and file management.
2. Implement concurrent and multithreaded applications using synchronization and inter process communication mechanisms.
3. Analyze and simulate scheduling, memory management, and deadlock handling algorithms.
4. Apply Linux kernel and file system concepts through practical experimentation.
5. Evaluate system performance and resource utilization in a Linux environment.

LIST OF EXPERIMENTS

1. Implement process creation and management using fork(), exec(), and wait() system calls to demonstrate parent-child relationships and process states.
2. Develop a multithreaded program using POSIX threads (pthreads) and implement synchronization using mutexes to manage shared resources.
3. Implement Interprocess Communication (IPC) using pipes and shared memory to enable communication between multiple processes.
4. Develop a solution for the Producer-Consumer problem using semaphores to ensure proper synchronization and avoid race conditions.
5. Implement CPU scheduling algorithms (FCFS, SJF, Priority, Round Robin) and develop a performance comparison module to compute waiting and turnaround times.
6. Develop and implement page replacement algorithms (FIFO, LRU, Optimal) and analyze page fault rates.
7. Implement Banker's Algorithm to demonstrate deadlock avoidance and verify safe/unsafe states.
8. Develop file handling programs using Linux system calls (open(), read(), write(), lseek(), close()) to perform low-level file operations.

9. Implement a basic Linux kernel module to demonstrate kernel-space programming and interaction with the /proc file system.
10. Develop a system performance monitoring tool using Linux utilities and system calls to analyze CPU, memory, and I/O usage.

COURSE OUTCOMES

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

CO1:	Implement process creation, CPU scheduling, and synchronization mechanisms using Linux system programming constructs.	K1-K6
CO2:	Develop multithreaded and inter-process communication (IPC) based applications ensuring efficient resource sharing and concurrency control.	
CO3:	Simulate and evaluate memory management strategies and deadlock avoidance algorithms to assess system performance.	
CO4:	Design and execute file system operations and system-level programs using Linux system calls.	
CO5:	Analyze Linux system performance using monitoring tools and demonstrate kernel-level concepts related to security and optimization.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	L	M	M	H	L	M	L	M	H	M	M	H	L
CO2	H	H	M	M	H	H	M	M	L	M	H	H	M	H	L
CO3	H	H	M	L	M	M	L	M	L	M	M	H	H	M	L
CO4	H	M	M	M	M	H	L	M	L	M	H	M	H	H	L
CO5	H	H	H	M	M	H	L	H	M	H	M	M	H	H	H

H-High; M-Medium; L-Low

PYTHON PROGRAMMING LAB

COURSE OBJECTIVES:

This course enables the students:

- To master the fundamentals of writing python scripts
- To create program using elementary data items
- To implement Python programs with conditionals and loops
- To use functions for structuring Python programs
- To develop web programming with Django

LIST OF EXPERIMENTS

IMPLEMENT THE FOLLOWING IN PYTHON:

1. Develop a student performance management system using lists, dictionaries, and tuples for structured data handling.
2. Create a payroll processing system using conditional statements and loops for salary computation and tax calculation.
3. Design a loan eligibility evaluation module using user-defined functions and validation logic.
4. Build a library management system using classes and objects to manage books and members.
5. Implement an employee hierarchy system using inheritance to model roles, responsibilities, and salary structures.
6. Develop a banking transaction system demonstrating polymorphism for different account types and interest calculations.
7. Perform financial risk analysis and portfolio return computation using NumPy for numerical modeling.
8. Conduct sales trend analysis and customer segmentation using Pandas for business intelligence.
9. Generate COVID-19 or stock market trend visualizations using Matplotlib for analytical reporting.
10. Develop a student admission or online feedback management portal using Django forms for dynamic web interaction.

COURSE OUTCOMES

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

CO1	Demonstrate proficiency in Python programming constructs and develop structured scripts for problem-solving.	K1-K6
CO2	Develop Python applications using core data structures such as lists, dictionaries, tuples, and sets for real-world scenarios.	
CO3	Design and implement object-oriented solutions using classes, inheritance, and polymorphism to model practical systems.	
CO4	Analyze and apply Python libraries (NumPy, Pandas, Matplotlib) for numerical computation and data visualization.	
CO5	Develop and deploy dynamic web applications using the Django framework.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze,K5-Evaluate,K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	M	H	L	M	L	M	H	M	L	M	L
CO2	H	H	L	M	M	H	M	M	L	M	H	H	L	M	L
CO3	H	H	M	H	H	H	M	M	L	M	H	H	M	H	L
CO4	H	H	L	M	M	H	L	M	M	L	M	M	H	H	M
CO5	H	H	M	H	H	H	H	M	M	M	H	H	M	H	M

H-High; M-Medium; L-Low

DATA VISUALIZATIONS LAB

COURSE OBJECTIVES

- To learn the basic functions and operations of Excel and tableau
- To understand principles of data visualization and visual analytics.
- To design interactive dashboards using spreadsheet and BI tools.
- To apply data transformation, aggregation, and storytelling techniques.
- To develop business intelligence solutions using Excel and Tableau.

IMPLEMENT THE FOLLOWING USING EXCEL

LIST OF EXPERIMENTS

1. Create a Retail Sales Performance Dashboard using Microsoft Excel / Tableau to analyze regional sales, profit margins, and category-wise trends; Dataset: Sample Superstore (Tableau Public) – <https://public.tableau.com/app/resources/sample-data>
2. Develop a Global Financial Market Analysis dashboard using Microsoft Excel to visualize stock price movements, moving averages, and volatility indicators; Dataset: Yahoo Finance Historical Data – <https://finance.yahoo.com/lookup>
3. Create an HR Attrition Analytics Dashboard using Microsoft Excel / Tableau to examine employee turnover and departmental attrition; Dataset: IBM HR Analytics (Kaggle) – <https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset>
4. Develop a COVID-19 Trend Visualization dashboard using Tableau to present time-series analysis of confirmed cases and vaccination rates; Dataset: Our World in Data – <https://ourworldindata.org/coronavirus-source-data>
5. Create an Indian State-wise Economic Indicators Dashboard using Microsoft Excel / Tableau to visualize GDP, inflation, and sectoral growth; Dataset: Open Government Data Platform India – <https://data.gov.in/>
6. Develop a Healthcare Resource Utilization Dashboard using Tableau to analyze hospital admissions, bed capacity, and mortality indicators; Dataset: WHO Global Health Observatory – <https://www.who.int/data/gho>
7. Create a Climate Change and Environmental Trends Dashboard using Microsoft Excel / Tableau to visualize CO₂ emissions and temperature anomalies; Dataset: World Bank Open Data – <https://data.worldbank.org/>
8. Develop a Crime Statistics Geographic Dashboard using Tableau to generate heatmaps and trend analysis of crime rates; Dataset: National Crime Records Bureau – <https://ncrb.gov.in/>

9. Create an E-Commerce Customer Segmentation Dashboard using Microsoft Excel / Tableau to perform RFM-based customer segmentation analysis; Dataset: UCI Online Retail Dataset – <https://archive.ics.uci.edu/ml/datasets/Online+Retail>
10. Develop an Education Performance Analytics Dashboard using Microsoft Excel / Tableau to analyze literacy rates and enrollment statistics; Dataset: UNESCO Institute for Statistics – <http://data.uis.unesco.org/>

COURSE OUTCOMES

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

CO1:	Develop and apply advanced data preprocessing and transformation techniques using Microsoft Excel and Tableau for real-world datasets.	K1-K6
CO2:	Design and construct interactive analytical visualizations, including KPIs, trend analyses, and comparative charts, to support data-driven decision-making.	
CO3:	Create advanced geographic maps, heat maps, and parameter-driven visual models for multidimensional data interpretation.	
CO4:	Develop integrated and interactive dashboards with filters, drill-down features, and dynamic controls for executive-level reporting.	
CO5:	Evaluate, interpret, and communicate insights from diverse open-source datasets using appropriate visualization strategies and storytelling techniques.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-evaluate and K6-Create

MAPPING WITH PROGRAMME OUTCOMES:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	-	L	M	H	L	L	L	-	H	M	-	M	-
CO2	M	H	-	H	M	H	L	L	M	-	H	H	-	M	-
CO3	H	H	-	M	M	M	L	L	H	-	M	H	M	H	-
CO4	M	H	M	H	H	M	L	M	L	-	H	M	H	M	-
CO5	M	H	H	H	M	M	L	H	M	H	M	M	M	H	H

H-High; M-Medium; L-Low

SOFT SKILL DEVELOPMENT LAB

COURSE OBJECTIVE

This course enables the students:

- To enable students to gain basic communication skills in professional and social contexts effectively.
- To acquire useful words and apply the min situational context.
- To develop listening and reading skills through comprehension passages
- To enrich the leadership qualities and interpersonal communication
- To enhance essential characteristics in writing

LIST OF EXPERIMENTS

1. Analyze and rewrite a poorly drafted technical document to demonstrate the characteristics of effective technical writing.
2. Develop a personal employability skills portfolio including SWOT analysis and career goal mapping.
3. Create a contextual vocabulary journal by identifying, defining, and applying new professional terms in sentences.
4. Complete sentence construction exercises focusing on clarity, coherence, and professional tone.
5. Identify and correct grammatical and syntactical errors in business and technical passages.
6. Interpret and solve verbal analogy problems to strengthen logical reasoning skills.
7. Analyze reading comprehension passages to interpret main ideas and conceptual understanding.
8. Evaluate reasoning-based comprehension passages to infer conclusions and critical insights.
9. Draft professional emails, technical blog posts, and forum responses following industry standards.
10. Prepare and deliver a technical presentation using structured slides with effective visual aids.
11. Design a professional resume and cover letter tailored to a specific job role.

12. Participate in mock interview sessions including HR and technical rounds with feedback analysis.
13. Conduct and evaluate group discussion sessions on current industry topics.
14. Practice listening comprehension exercises using recorded lectures or interviews and summarize key points.
15. Deliver short speeches and role-play situational conversations in professional contexts.
16. Analyze English usage in mass media (news articles, advertisements, editorials) for communication effectiveness.
17. Perform structured grammar practice covering tense consistency, subject–verb agreement, and sentence transformation.
18. Engage in peer collaboration activities to solve case-based communication challenges.
19. Participate in team-building exercises to demonstrate leadership and conflict resolution skills.
20. Develop and present persuasive communication speeches addressing contemporary social or professional issues.

TEXT BOOKS

1. Uma Narula, “Development Communication: Theory and Practice”, Revised Edition, Har-Anand Publication, 2019.
2. Annette Capel and Wendy Sharp, “Cambridge English: Objective First”, Fourth Edition, Cambridge University Press, 2013.
3. Emma Sue-Prince, “The Advantage: The 7 Soft Skills You Need to Stay One Step Ahead”, First Edition, FT Press, 2013.
4. Guy Brook-Hart, “Cambridge English: Business Benchmark”, Second Edition, Cambridge University Press, 2014.
5. Norman Lewis, “How to Read Better & Faster”, Binny Publishing House, New Delhi, 1978.

REFERENCE BOOKS

1. Michael McCarthy and Felicity O’Dell, “English Vocabulary in Use: 100 Units of Vocabulary Reference and Practice”, Cambridge University Press, 1996.
2. Murphy, Raymond, “Intermediate English Grammar”, Second Edition, Cambridge University Press, 1999.

COURSE OUTCOMES

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

CO1:	Demonstrate effective verbal and written communication skills in professional and social contexts.	K1-K6
CO2:	Apply appropriate vocabulary, grammar, and sentence structures in technical and workplace communication.	
CO3:	Analyze reading and listening comprehension passages to interpret concepts, reasoning, and contextual meaning.	
CO4:	Exhibit leadership, teamwork, and interpersonal communication skills in collaborative environments.	
CO5:	Create professional documents and presentations including resumes, emails, blogs, and technical presentations.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6-Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	M	M	H	H	H	H	M	M	M	H	H	M	L	H	M
CO2	M	M	H	H	M	H	L	M	M	H	H	M	L	H	M
CO3	H	H	M	M	M	M	L	M	M	M	M	M	L	H	M
CO4	M	M	H	H	H	H	H	H	M	H	H	H	M	H	H
CO5	M	H	H	H	M	H	M	H	M	H	H	H	L	H	M

H-High; M-Medium; L-Low

INTERNET OF THINGS LAB

COURSE OBJECTIVE

This course enables the students:

1. Understand IoT architecture, embedded platforms, and sensor interfacing.
2. Design and implement IoT-based applications using microcontrollers.
3. Develop real-time monitoring systems using sensors and actuators.
4. Apply networking concepts for device communication and web-based control.
5. Build domain-specific IoT solutions for agriculture, home automation, and healthcare.

LIST OF EXPERIMENTS

1. Design and implement an IoT-based LED control system for ON/OFF switching (3.3V).
2. Develop an IoT-based object detection system using an IR sensor.
3. Implement an IoT-based environmental monitoring system using temperature and humidity sensors.
4. Design and deploy a local IoT web server for device monitoring and control.
5. Develop an IoT-based soil moisture monitoring system for smart irrigation.
6. Implement an IoT-based distance measurement system using an ultrasonic sensor.
7. Design a real-time IoT-based home automation system using a relay module (230V).
8. Develop an IoT-based fire detection and alert system.
9. Implement an IoT-based gas leakage detection and alert system.
10. Design an IoMT-based health monitoring system using a heartbeat sensor.

LABORATORY REQUIREMENTS

Hardware

- Arduino Uno / ESP8266 / ESP32
- Breadboard and jumper wires
- LED, Resistors
- IR Sensor
- DHT11/DHT22 (Temperature & Humidity Sensor)
- Soil Moisture Sensor
- Ultrasonic Sensor (HC-SR04)
- Relay Module
- Gas Sensor (MQ series)
- Flame Sensor

- Heartbeat Sensor

Software

- Arduino IDE
- Tinkercad / Proteus / Wokwi (IoT Simulation Tools)
- Embedded C / Python (optional for advanced IoT)

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

CO1	Interface sensors and actuators with embedded systems for IoT applications.	K1-K6
CO2	Develop IoT-based environmental and agricultural monitoring systems.	
CO3	Implement web-enabled IoT systems for data visualization and control.	
CO4	Design smart IoT applications for automation and safety.	
CO5	Develop IoMT-based health monitoring solutions.	

K1-Remember, K2-Understand, K3-Apply, K4 -Analyze, K5 -Evaluate, K6 –Create

MAPPING WITH PROGRAMME OUTCOMES:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	L	M	L	M	M	M	H	M	M	M	L
CO2	H	H	L	M	M	H	M	M	L	M	H	H	M	H	L
CO3	H	H	L	M	M	H	M	M	L	M	H	H	H	H	L
CO4	H	H	M	H	M	H	M	M	L	M	H	H	H	H	M
CO5	H	H	M	H	H	H	H	H	M	H	H	H	H	H	H

- H-High; M-Medium; L-Low

Subject Code: 26UPCSC1C04

Credits: 04

DATA STRUCTURES AND ALGORITHMS

COURSE OBJECTIVES:

- To get a clear understanding of various ADT structures.
- To understand how to implement different ADT structures with real-time scenarios.
- To analyze the various data structures with their different implementations.
- To get an idea of applying right models based on the problem domain.
- To realize, and understand how and where to implement modern data structures with Python language.

UNIT-I

Abstract Data Types: Introduction-Data Abstract Data Type – Bags- Iterators. **Arrays:** Array Structure-Python List-Two Dimensional Arrays-Matrix Abstract Data Type. **Sets, Maps:** Sets – Maps – Multi-Dimensional Arrays.

UNIT-II

Algorithm Analysis: Experimental Studies-Seven Functions-Asymptotic Analysis. **Recursion:** Illustrative Examples-Analyzing Recursive Algorithms-Linear Recursion- Binary Recursion-Multiple Recursion.

UNIT-III

Stacks, Queues, and De-queues: Stacks – Queues – Double-Ended Queues Linked. **Lists:** Singly Linked Lists-Circularly Linked Lists-Doubly Linked Lists. **Trees:** General Trees-Binary Trees- Implementing Trees-Tree Traversal Algorithms.

UNIT-IV

Priority Queues: Priority Queue Abstract Data Type- Implementing a Priority Queue – Heaps-Sorting with a Priority Queue. **Maps, Hash Tables, and Skip Lists:** Maps and Dictionaries-Hash Tables- Sorted Maps-Skip Lists-Sets, Multi Sets, and Multi Maps.

UNIT-V

Search Trees: Binary Search Trees-Balanced Search Trees-AVL Trees-Splay Trees. **Sorting and Selection:** Merge sort-Quick sort-Sorting through an Algorithmic Lens- Comparing Sorting

Algorithms-Selection. **Graph Algorithms:** Graphs-Data Structures for Graphs- Graph Traversals- Shortest Paths- Minimum Spanning Trees.

TEXT BOOKS:

1. RanceD. Necaise, “Data Structures and Algorithms Using Python”, John Wiley & Sons, 2011. (Unit – 1)**Chapters:** 1, 2, 3.
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python”, John Wiley & Sons, 2013. (Unit– 2, 3, 4, and 5) **Chapters:** 3 to 12, and 14.

REFERENCE BOOKS:

1. Dr. Basant Agarwal; Benjamin Baka, “Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7”, Packt Publishing, 2018.
2. Magnus Lie Hetland, “Python Algorithms: Mastering Basic Algorithms in the Python Language”, Apress, 2014.

COURSE OUTCOME:

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

CO1	Understand various ADT concepts	K1-K6
CO2	Familiar with implementation of ADT models with Python language and Understand how to develop ADT for the various real-time problems	
CO3	Apply with proper ADT models with problem understanding	
CO4	Apply and Analyze right models based on the problem domain	
CO5	Evaluate modern data structures with Python language	

K1-Remember, K2-Understand, K3-Apply, K4 -Analyze, K5 -Evaluate, K6 –Create

MAPPING WITH PROGRAMME OUTCOMES:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	L	M	L	M	M	M	H	M	M	M	L
CO2	H	H	L	M	M	H	M	M	L	M	H	H	M	H	L
CO3	H	H	L	M	M	H	M	M	L	M	H	H	H	H	L
CO4	H	H	M	H	M	H	M	M	L	M	H	H	H	H	M
CO5	H	H	M	H	H	H	H	H	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

BIG DATA ANALYTICS

COURSE OBJECTIVES

- To understand Big Data concepts and NoSQL data management systems.
- To learn Hadoop architecture, HDFS, and distributed computing concepts.
- To implement distributed data processing using MapReduce.
- To explore Big Data ecosystem tools such as Hive and Pig.
- To analyze large-scale data processing using Spark.

UNIT-I

Big Data Introduction and NoSQL Data Management : Introduction to Big Data; Characteristics of Big Data; Need for Big Data; Big Data versus other techniques; Structured and Unstructured Data; Industry Applications of Big Data; Introduction to Apache Hadoop; Hadoop Ecosystem.

Introduction to NoSQL Databases; Terminologies in NoSQL and RDBMS; Types of NoSQL Databases – Key-Value, Document, Column Family, Graph Databases; SQL versus NoSQL; Data Consistency Issues; ACID versus BASE; Relaxing Consistency; Introduction to HBase

UNIT- II

Hadoop Fundamentals and HDFS : Basics of Hadoop; Data Distribution; Hadoop Distributed File System (HDFS); NameNode and DataNode; Data Flow (File Read and Write); Hadoop I/O; Compression; Serialization; Avro File-based Data Structures; Hadoop Installation – Standalone Mode, Fully Distributed Mode; Multi-node Cluster Configuration

UNIT- III

MapReduce Framework: MapReduce Architecture; MapReduce Data Types; File Input and Output Formats; Java MapReduce; Partitioner and Combiner; **MapReduce Applications:** MapReduce Workflow; Debugging MapReduce Job; Job Chaining and Control; MapReduce 1 and MapReduce 2 (YARN); Failure Handling in MapReduce

UNIT -IV

Hive and Pig: Apache Hive – Architecture; Data Types; File Formats; HiveQL – DDL and DML; Schema on Read versus Schema on Write; Transactions and Indexes;

Apache Pig – Installation; Execution Type; Pig Data Model; PigLatin Scripts; Relational

Operations; User-Defined Functions; Comparison between Hive and traditional database.

UNIT -V

Apache Spark: Introduction to Apache Spark; Spark Programming Model; Resilient Distributed Datasets (RDD); Spark methods for Data processing; Example of programs using Scala: Shuffling; Memory Management Issues;

TEXT BOOKS

1. Mayank Bhushan, *Big Data and Hadoop: Fundamentals, Tools, and Techniques for Data-Driven Success*, 2nd Edition, BPB Publications, 2024.

REFERENCE BOOKS:

1. Seema Acharya, Subhashini Chellappan, “Big Data and Analytics”, Wiley Publications, First Edition, 2015
2. Judith Huruwitz, Alan Nugent, Fern Halper, Marcia Kaufman, “Big data for dummies”, John Wiley & Sons, Inc. (2013)
3. Tom White, “Hadoop The Definitive Guide”, O’Reilly Publications, Fourth Edition, 2015
4. Dirk Deroos, Paul C. Zikopoulos, Roman B. Melnky, Bruce Brown, Rafael Coss, “Hadoop For Dummies”, Wiley Publications, 2014
5. Robert D. Schneider, “Hadoop For Dummies”, John Wiley & Sons, Inc. (2012)
6. Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill, 2012 Chuck Lam, “Hadoop In Action”, Dream tech Publications, 2010

COURSE OUTCOMES

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

CO1	Demonstrate comprehensive understanding of Big Data concepts, characteristics, NoSQL data management models, and distributed computing paradigms.	K1–K6
CO2	Design and implement distributed storage solutions and resource management mechanisms using Apache Hadoop and HDFS architecture.	
CO3	Develop, optimize, and evaluate MapReduce programs for large-scale parallel data processing in distributed environments.	

CO4	Apply data warehousing and large-scale data transformation techniques using Apache Hive and Apache Pig for analytical queries.	
CO5	Analyze, process, and interpret large datasets using Apache Spark to support scalable and real-time analytics applications.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5 Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	L	M	L	M	M	M	H	M	M	M	L
CO2	H	H	L	M	M	H	M	M	L	M	H	H	M	H	L
CO3	H	H	L	M	M	H	M	M	L	M	H	H	H	H	L
CO4	H	H	M	H	M	H	M	M	L	M	H	H	H	H	M
CO5	H	H	M	H	H	H	H	H	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

Subject Code: 26UPCSC1C06

Credits: 04

ARTIFICIAL INTELLIGENCE CONCEPTS AND TECHNIQUES

COURSE OBJECTIVES

1. To understand the fundamental concepts and problem-solving methodologies in Artificial Intelligence.
2. To apply heuristic search and knowledge representation techniques for intelligent systems.
3. To analyze machine learning paradigms and soft computing approaches.
4. To develop probabilistic reasoning and decision-making models.
5. To examine classification, reinforcement learning, and intelligent decision processes.

UNIT I: Foundations of Artificial Intelligence

Introduction to Artificial Intelligence – Definition of Artificial Intelligence Through Problems – History of Artificial Intelligence – Artificial Intelligence – Problems and Techniques – Production Systems – Shift in Focus of AI Towards Providing Smarter Solutions

Introduction – State Space Search – Production System – Problem Characteristics – Control Strategy – Issues in the Design of Search Programs – Search Strategies – Advanced Problems

UNIT II: Search Strategies and Knowledge Representation

Introduction :Generate-and-Test Method – Hill Climbing Method – Best First Search and A* Search – Means End Analysis – Intelligent Agents and Environment – Problem Reduction, AO* Algorithm – Constraint Satisfaction with Inference, Backtracking, and Local Search – Local Search Algorithms and Optimization Problems – Local Search in Continuous Spaces

Introduction : Ontologies, Objects, and Events – Representations and Mappings – Approaches to Knowledge Representation – Forward versus Backward Chaining – Matching and Control Knowledge – Slot and Filler Structures – Issues in Knowledge Representation – Developments in the Field of Knowledge Representation

UNIT III: Neural Networks and Fuzzy Logic

Introduction : Learning in Neural Networks – Choosing Cost Function – Types of Learning – Recurrent Neural Network – Back-propagation – Convolutional Neural Networks and Deep Neural Networks – Applications of Neural Networks – Challenges in Neural Networks

Introduction : Set Theory – Fuzzy Set Theory – Terminology Associated with Fuzzy Sets – Fuzzification and Defuzzification – Formation of Fuzzy Rules – Fuzzy Logic Inference System – Fuzzy Database and Queries – Fuzzy Logic Control System – Fuzzy Inference Processing: Mamdani and Sugeno – Adaptive Neuro-Fuzzy Inference System – Applications

UNIT IV: Statistical Learning and Decision Processes

Introduction : Probability Axioms – Bayes’ Rule – Bayesian Network – Dynamic Bayesian Networks – Hidden Markov Model – Probabilistic Reasoning – Certainty Factor Theory – Dempster–Shafer Theory

Introduction to Learning – Forms of Learning – Learning Decision Trees – Theory of Learning – Learning by Examples – Inductive Learning – Explanation-Based Learning – Regression and Classification with Linear Models – Artificial Neural Networks – Parametric Models – Non-Parametric Models – Support Vector Machines – Ensemble Learning – Statistical Learning – Reinforcement Learning – Applications of Reinforcement Learning

UNIT V: Applications of Artificial Intelligence

Introduction :Minimax Search Procedure – Alpha–Beta Cutoff – Imperfect Real-Time Decisions – Stochastic Games – State-of-the-Art Game Programs – Modern Examples-Expert System – Knowledge Representation – Expert System Shells – Knowledge Acquisition of an Expert System – Applications of Expert Systems – Examples of Expert Systems – Problem Solving Examples

TEXT BOOKS

1. Lavika Goel, *Artificial Intelligence: Concepts and Applications*, Wiley India Pvt. Ltd., First Edition, 2021.

REFERENCE BOOKS

1. Das,A.(2018). Artificial Intelligence and Soft Computing for Beginners.
2. Amit, K. (2018).Artificial intelligence and soft computing: behavioral and cognitive modeling of the human brain. CRC press.
3. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Prentice Hall of India, Third Edition 2014.

COURSE OUTCOMES:

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

CO1	Demonstrate comprehensive understanding of Artificial Intelligence foundations, problem-solving strategies, and search algorithms.	K1-K6
CO2	Apply knowledge representation, logical reasoning, and heuristic techniques to model and solve complex AI problems.	
CO3	Design and implement machine learning models including neural networks and fuzzy logic systems for intelligent applications.	
CO4	Analyze and develop probabilistic reasoning, statistical learning, and reinforcement learning frameworks for decision-making under uncertainty.	
CO5	Evaluate and integrate classification, optimization, and decision-theoretic approaches to address real-world intelligent system challenges.	

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	L	M	L	M	M	M	H	M	M	M	L
CO2	H	H	H	H	M	H	M	M	L	M	H	H	M	H	L
CO3	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	H	M	M	H	H	H	H	H	M
CO5	H	H	H	H	H	H	H	H	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

DATA STRUCTURES AND ALGORITHMS LAB

COURSE OBJECTIVES

- To understand Stack, Queue and Doubly Linked ADT structures.
- To implement different ADT structures with real-time scenarios.
- To analyze the recursion concepts.
- To apply different sorting and tree techniques.
- To implement modern data structures with Python language.

LIST OF EXPERIMENTS

IMPLEMENT THE FOLLOWING PROBLEMS USING PYTHON3.4 AND ABOVE

1. Recursion concepts.
 - i) Linear recursion
 - ii) Binary recursion.
2. Stack ADT.
3. Queue ADT.
4. Doubly Linked List ADT.
5. Heaps using Priority Queues.
6. Merge sort.
7. Quick sort.
8. Binary Search Tree.
9. Minimum Spanning Tree.
10. Depth First Search Tree traversal.

COURSE OUTCOMES

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

CO1	Strong understanding in various ADT concepts	K1-K6
CO2	To become a familiar with implementation of ADT models	
CO3	Apply sort and tree search algorithms	
CO4	Evaluate the different data structure models	
CO5	Learn how to develop ADT for the various real-time problems	

K1-Remember, K2-Understand, K3- Apply,K4 -Analyze,K5 -Evaluate,K6 -Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	H	M	L	L	L	L	H	H	M	M
CO2	H	M	H	M	M	L	H	M	H	L
CO3	H	H	H	L	L	L	M	M	M	M
CO4	H	H	H	M	M	H	M	M	H	H
CO5	H	H	H	H	L	M	H	M	M	M

H-High; M-Medium; L-Low

BIG DATA ANALYTICS LAB

COURSE OBJECTIVES

- To teach the fundamental techniques for handling the big data tools.
- To familiarize the tools required to manage big data.
- To analyze big data using Hadoop, MapReduce, Hive, and Pig
- To teach the fundamental principles in achieving big data analytics with scalability and streaming capability
- To enable students to have skills that will help them to solve complex.

LIST OF EXPERIMENTS

1. Implement advanced HDFS file system operations including directory creation, file upload/download, replication factor modification, and file block inspection in a Hadoop environment.
2. Develop a MapReduce program with a single reducer to compute maximum, minimum, average, and total count from a large numerical dataset.
3. Design a MapReduce program with multiple reducers to perform word count along with frequency-based sorting and filtering of common words from a large text corpus.
4. Implement full CRUD operations on a semi-structured JSON dataset using MongoDB including indexing and validation rules.
5. Develop a MongoDB aggregation pipeline to analyze customer transaction data by grouping, sorting, filtering, and computing revenue metrics.
6. Implement Pig Latin scripts to perform input/output handling, relational operations, joins, grouping, and data transformation on large datasets.
7. Develop Pig Latin programs using User Defined Functions (UDFs) and advanced relational operations such as nested data handling and multi-stage transformations.
8. Perform distributed data transformations and key-based aggregations (map, filter, reduce By Key, join) on large CSV datasets using Spark RDD in Apache Spark (Colab environment).
9. Implement advanced Spark SQL queries including grouping, joining, window functions, and aggregation on structured datasets using Apache Spark in Colab.
10. Develop an end-to-end Spark analytics workflow to perform descriptive statistics, trend analysis, and visualization on a large real-world dataset.

COURSE OUTCOMES

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

CO1:	Demonstrate practical proficiency in distributed storage and file management using Hadoop and HDFS.	K1-K6
CO2:	Develop and execute MapReduce programs to perform large-scale parallel data processing.	
CO3:	Implement data manipulation and analytical queries using Pig, MongoDB, and Spark frameworks.	
CO4:	Apply distributed computing techniques using Spark RDD and Spark SQL for scalable data analytics.	
CO5:	Design and integrate end-to-end Big Data analytics workflows for real-world datasets using open-source tools.	

K1-Remember, K2-Understand, K3-Apply, K4-Analyze, K5-Evaluate, K6-Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	L	M	H	M	L	L	M	M	H	M	M	M	L
CO2	H	H	M	M	H	H	M	L	M	M	H	H	M	H	L
CO3	H	H	M	H	H	H	M	M	M	M	H	H	H	H	M
CO4	H	H	H	H	H	H	H	M	M	H	H	H	H	H	M
CO5	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H

H-High; M-Medium; L-Low

FUNDAMENTALS OF HUMAN RIGHTS

UNIT-I:

INTRODUCTION: Meaning and Definitions of Human Rights - Characteristics and Importance of Human Rights - Evolution of Human Rights - Formation, Structure and Functions of the UNO - Universal Declaration of Human Rights - International Covenants - Violations of Human Rights in the Contemporary Era.

UNIT-II:

HUMANRIGHTS IN INDIA: Development of Human Rights in India - Constituent Assembly and Indian Constitution - Fundamental Rights and its Classification - Directive Principles of State Policy - Fundamental Duties.

UNIT-III

Rights of Marginalized and other Disadvantaged People: Rights of Women - Rights of Children - Rights of Differently Abled - Rights of Elderly - Rights of Scheduled Castes - Rights of Scheduled Tribes - Rights of Minorities - Rights of Prisoners - Rights of Persons Living with HIVAIDS - Rights of LGBT.

UNIT-IV

Human Rights Movements: Peasant Movements (Tebhaga and Telangana) - Scheduled Caste Movements (Mahar and Ad-Dharmi) - Scheduled Tribes Movements (Santhal and Munda) - Environmental Movements (Chipko and Narmada Bachao Andolan) - Social Reform Movements (Vaikom and Self Respect).

UNIT-V

Redressal Mechanisms: Protection of Human Rights Act, 1993 (Amendment 2019) - Structure and Functions of National and State Human Rights Commissions-National Commission for SCs
- National Commission for STs - National Commission for Women - National Commission for Minorities - Characteristics and Objectives of Human Rights Education.

REFERENCE BOOKS

1. Sudarshanam Gankidi, Human Rights in India: Prospective and Retrospective, Rawat Publications, Jaipur, 2019.
2. Satvinder Juss, Human Rights in India, Routledge, New Delhi, 2020.
3. Namita Gupta, Social Justice and Human Rights in India, Rawat Publications, Jaipur, 2021.
4. Mark Frezo, The Sociology of Human Rights, John Willy & Sons, U.K. 2014.
5. Chiranjivi J. Nirmal, Human Rights in India: Historical, Social and Political Perspectives, Oxford University Press, New York, 2000.
6. Dr. S. Mehataj Begum, Human Rights in India: Issues and perspectives, A PH Publishing Corporation, New Delhi, 2010.
7. Asha Kiran, The History of Human Rights, Mangalam Publications, Delhi, 2011.
8. Bani Borgohain, Human Rights, Kanishka Publishers & Distributors, New Delhi-2, 2007.
9. Jayan Chudhary, A Text book of Human Rights, Wisdom Press, New Delhi, 2011.

CLOUD COMPUTING WITH PROMPT ENGINEERING LAB

COURSE OBJECTIVES

- To understand the fundamental concepts of cloud computing and its role in deploying scalable applications.
- To introduce the principles of prompt engineering for effective interaction with AI systems.
- To develop skills in designing, refining, and optimizing prompts to improve AI-generated outputs.
- To gain practical experience in integrating AI models with cloud-based services and workflows.
- To apply ethical practices and responsible approaches in the use of AI technologies.

LIST OF LABORATORY EXERCISES

1. Implement a basic cloud storage application using Python by creating a storage bucket, uploading a file, and retrieving the file from the cloud.
2. Create and configure a virtual machine in a cloud platform, install Python, and execute a simple Python program on the virtual machine.
3. Develop a serverless function using Python that performs a basic task such as addition of numbers and trigger it through a web request.
4. Design a simple REST API using a Python framework such as Flask, implement GET and POST methods, and test the API locally or on the cloud.
5. Create a basic cloud workflow that accepts input, processes the data, and produces output through a sequence of steps.
6. Generate responses using zero-shot prompting by providing a direct instruction to an AI model without giving any examples.
7. Generate responses using few-shot prompting by including one or two examples in the prompt and compare the results with zero-shot prompting.
8. Develop role-based prompts by assigning different roles such as teacher or doctor and analyze how the responses vary based on the role.
9. Perform prompt refinement by modifying a simple prompt with additional instructions and constraints to improve the quality of the output.
10. Develop a basic AI-based text generation application using Python or Streamlit that accepts user input and produces generated text.

COURSE OUTCOMES

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

CO1	Demonstrate the application of fundamental prompt engineering techniques to generate structured and meaningful outputs using AI tools.	K1-K6
CO2	Analyze and refine prompts to enhance the accuracy, relevance, and consistency of AI-generated responses.	
CO3	Evaluate AI outputs with respect to bias, hallucination, ethical considerations, and reliability.	
CO4	Design AI-assisted solutions integrated with cloud platforms for business, communication, and decision-making applications.	
CO5	Develop prompt workflows to address real-world problems with entrepreneurial and societal relevance.	

MAPPING WITH PROGRAMME OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	1	2	1	3	1	1	1	1	3	2	1	2	1
CO2	3	3	2	2	2	3	2	1	1	2	3	3	2	3	1
CO3	2	3	3	2	2	2	1	2	2	3	2	2	3	3	2
CO4	3	3	2	3	3	3	3	2	2	2	3	3	2	3	2
CO5	3	3	3	3	3	3	3	3	2	3	3	3	3	3	2

Subject Code: 26UPCSC1C07

Credits: 04

ADVANCED JAVA PROGRAMMING

COURSE OBJECTIVES

- To understand core Java programming fundamentals and program structure
- To apply object-oriented programming principles in application development
- To design GUI-based desktop applications using AWT and Swing
- To develop restful web applications using Spring Boot
- To integrate database connectivity and micro service concepts in Java applications

UNIT –I:

JAVA BASICS: Java Architecture – JVM, JRE, JDK – Java Development Environment Setup – Compilation and Execution – Structure of a Java Program – Data Types – Variables – Operators – Type Casting – Control Statements (if, if-else, switch) – Looping Statements (for, while, do-while) – Arrays (1D and 2D) – Strings – Methods – Parameter Passing – Command Line Arguments.

UNIT -II:

OBJECT ORIENTED PROGRAMMING : OOP Concepts – Abstraction – Encapsulation – Inheritance – Polymorphism – Classes and Objects – Constructors – Method Overloading – Method Overriding – Access Modifiers – Static Members – this and super keywords – Interfaces – Packages – Exception Handling (Basics) – Composition vs Inheritance.

UNIT- III

GUI PROGRAMMING : Introduction to GUI Programming – AWT Components (Label, Button, TextField) – Event Handling – Event Classes – Listener Interfaces – Adapter Classes – Swing Architecture – JFrame – Content Pane – Layout Managers – Handling Mouse and Key Events – Development of Simple GUI Applications.

UNIT-IV:

SPRING BOOT FUNDAMENTALS : Introduction to Spring Boot – Features and Advantages – Microservices Architecture – Setting up Development Environment – Spring Initializr – Maven and Gradle – Project Structure – pom.xml and build.gradle – Starter Dependencies – Auto-Configuration – Core Annotations – Dependency Injection – Configuration Properties – Profiles – Building First Spring Boot Application – Embedded Tomcat – 12-Factor Application Principles.

UNIT- V:

RESTFUL SERVICES AND DATABASE INTEGRATION : Spring Boot Annotations – @Rest Controller – @Request Mapping – @Auto wired – @Component – @Service – @Repository – Spring Data JPA – Working with H2 and My SQL – Caching – RESTful API Development – Exception Handling using @Controller Advice – Actuator Endpoints – Service Discovery (Introduction) – Environment Profiles – Basic Micro services Concepts.

TEXT BOOKS:

1. **Michael B. White**, *Mastering Java: An Effective Project-Based Approach Including Web Development, Data Structures, GUI Programming and Object-Oriented Design*, Mercury Learning and Information, 2018.
2. **Shagun Bakliwal**, *Hands-on Application Development using Spring Boot: Building Modern Cloud Native Applications*, BPB Publications, First Edition, 2022.

REFERENCE BOOKS:

1. Herbert Schildt, “JavatheCompleteReference”, 10th edition, McGraw Hill Publishing Company Ltd, New Delhi, 2017.
2. TonyGoddis, “StartingoutwithJavafromControlStructuresThroughObjects” 6th Edition, Pearson Education Limited, 2016
3. John Dean, Raymond Dean, “Introduction to Programming with JAVA – A Problem Solving Approach”, TMGH Publishing Company Ltd, New Delhi, 2012.

COURSE OUTCOMES

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

CO1	Develop Java programs using core language constructs.	K1-K6
CO2	Apply object-oriented programming principles in application development.	
CO3	Design GUI-based applications using AWT and Swing.	
CO4	Develop restful web applications using Spring Boot framework.	
CO5	Build database-driven and micro service-based applications using Spring Boot and JPA.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	H	M	L	M	M	M	H	H	M	M	L
CO2	H	H	M	M	H	M	L	M	M	M	H	H	M	M	L
CO3	H	M	M	H	H	H	M	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	H	M	M	H	H	H	H	H	M
CO5	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H

H-High; M-Medium; L-Low

WEB TECHNOLOGIES**COURSE OBJECTIVES**

- To provide a strong foundation in web technologies including HTML5, CSS3, and JavaScript.
- To develop client-side and server-side programming skills using modern frameworks.
- To design and implement RESTful APIs and dynamic web applications using Node.js and Express.
- To integrate relational databases with web applications using SQL concepts.
- To deploy secure, scalable web applications with authentication and cloud hosting mechanisms.

UNIT –I:

Web Fundamentals and HTML: Full Stack Web Development – Layers of Full Stack Development – Who is a Full Stack Web Developer – Why are Full Stack Web Developers in High Demand at Present – How to Become a Full Stack Web Developer – Basic Data Structures and Algorithms – Concepts of the Web Every Developer Needs to Know – Understanding the Types of Web – Understanding Web Development – A Web Designer and a Web Developer – Types of Web Developers – Understanding Web Technology Stacks – Front-End, Client-Side Technology Stack – Back-End, Server-Side Technology Stack – Web Development Fundamentals

UNIT- II

Web Foundations and Front-End Development: Front-End Development – HTML – HTML Document Structure – HTML Elements – HTML Forms – HTML Media and Graphics – HTML APIs – HTML Boilerplate and Doctypes – HTML Lists, Tables and Links – CSS3 – CSS Selectors and Properties – CSS Box Model – CSS Positioning – CSS Display and Layout – CSS Specificity – CSS Styling (Inline, Internal, External) – CSS Typography and Web Design – CSS Float and Clear

UNIT -III:

Backend Development with Node.js Back-End Development – Introduction to Back-End Development – Front-End vs Back-End Development – Back-End Developer Roles and Responsibilities – Back-End Languages – Back-End Tools and Technologies – Back-End Developer vs Back-End Engineer – Skills for Back-End Development – Back-End Development Careers – Node.js – Features of Node.js – Applications of Node.js – Node.js Environment Setup – Node.js REPL – Node.js Modules and NPM – Package.json – Callbacks – Event Loop and Event-Driven Programming – File System – HTTP Module – Node.js as Web Server.

UNIT- IV:

APIs and Server-Side Rendering: APIs – API Abstraction – Working of APIs – SOAP APIs – REST APIs – Types of APIs – HTTP Protocol – HTTP Requests and Responses – HTTP Methods – HTTP Headers and Status Codes – API Communication – API Security – API Documentation – API Authentication – JSON – XML – REST Principles – API Design – Public API Implementation – CRUD Operations using APIs – Express Route Handlers

UNIT -V:

Deployment and Security: Deployment – Web Hosting – Hosting Types – Website Deployment – Deployment Process – GitHub Pages – Heroku Deployment – MongoDB Atlas – Authentication – Encryption – Hashing (Bcrypt) – Sessions and Cookies – Local Authentication – Passport.js – Environment Variables – OAuth – API and Data Security

TEXT BOOKS:

1. Sammie Smith, *Full Stack Web Development: Everything Beginners to Expert Guide on Modern Full-Stack Web Development Using Modern Web Development Tools*, Fullstack Fulfillment Publications, First Edition, 2022.

REFERENCEBOOKS:

1. Robert W. Sebesta: *Programming the World Wide Web*, Eighth Edition, Pearson education, 2015.
2. Alex Banks, Eve Porcello, "Learning React", O'Reilly Media, Inc, 2nd Edition, 2020
3. Marc Wand schneider, "Learning Node", Addison-Wesley Professional, 2nd Edition, 2016
4. Chris Bates: *Web Technology Theory and Practice*, Pearson Education, 2012.
5. Raj Kamal: *Internet and Web Technologies*, McGrawHill Education.

COURSE OUTCOMES:

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

CO1	Explain the architecture and working principles of full stack web applications including client-server model, HTTP protocol, and web security fundamentals.	K1-K6
CO2	Design and develop responsive and interactive web interfaces using HTML5, CSS3, and modern JavaScript concepts.	
CO3	Develop scalable server-side applications and RESTful APIs using Node.js and Express framework.	
CO4	Integrate relational databases with full stack applications and perform structured data modeling and CRUD operations.	
CO5	Deploy secure full stack web applications implementing authentication, encryption, session management, and cloud hosting techniques.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	H	M	L	M	M	M	H	H	M	M	L
CO2	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO3	H	H	H	H	H	H	H	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	M	M	M	H	H	H	H	H	H
CO5	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

ADVANCED MACHINE LEARNING TECHNIQUES

COURSE OBJECTIVES

- To understand the concepts of Machine Learning.
- To understand the theoretical and practical aspects of types of machine learning
- To teach and get familiarized with supervised learning and their applications.
- To teach and get familiarized with the concepts and algorithms of unsupervised learning.
- To appreciate the concepts and algorithms of deep learning.

UNIT- I:

Introduction to Machine Learning: Introduction to Machine Learning – Human Learning and Types – Definition of Machine Learning – Well-Posed Learning Problems – Applications of Machine Learning – Issues in Machine Learning – Types of Machine Learning –Preparing to Model – exploring structure of data– Data Quality and Remediation– Data Preprocessing

UNIT –II:

Modeling and Feature Engineering: Modeling and Evaluation – Training and Testing – Holdout Method – Cross Validation – Bootstrap – Bias–Variance Tradeoff – Underfitting and Overfitting –Feature Engineering – Feature Construction – Feature Extraction – Feature Subset Selection.

UNIT –III:

Supervised Learning: Classification Techniques – k-Nearest Neighbour – Decision Tree – Random Forest – Support Vector Machine –Regression Techniques – Simple Linear Regression – Multiple Linear Regression – Logistic Regression – Maximum Likelihood Estimation

UNIT-IV:

Unsupervised Learning and Ensemble Methods: Unsupervised Learning – Clustering – Partitioning Methods – K-Medoids – Hierarchical Clustering – Density-Based Clustering (DBSCAN) – Association Rule Mining – Apriori Algorithm – Ensemble Learning – Bagging – Boosting – Random Forest – Regularization Algorithms

UNIT-V:

Neural Networks and Deep Learning: Basics of Neural Network: Introduction – Biological Neuron – Artificial Neuron – Activation Functions – Early Implementations of ANN – Neural Network Architectures – Learning Process in ANN – Backpropagation – Deep Learning

TEXT BOOKS

1. Saikat Dutt, S. Chandra mouli, Amit Kumar Das – *Machine Learning*, Pearson

REFERENCEBOOKS:

1. Brett Lantz, “Machine Learning with R”, Addison-Wesley Packt Publishing, 2013.
2. Taweh Beysolow, “Introduction to Deep Learning Using R: A Step-by-Step Guide to Learning and Implementing Deep Learning Models Using R”, San Francisco, California, USA, 2017.
3. Bertt Lantz, “Machine Learning with R: Expert techniques for predictive modeling”, 3rd Edition, April 15, 2019,
4. Jason Bell, “Machine Learning: Hands-On for Developers and Technical Professionals”, Wiley Publication, 2015.

COURSE OUTCOMES

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

CO1	Demonstrate comprehensive understanding of machine learning foundations, data preprocessing, and modeling concepts.	K1-K6
CO2	Apply feature engineering and model evaluation techniques to improve predictive performance.	
CO3	Implement supervised learning algorithms for classification and regression problems.	
CO4	Analyze and apply unsupervised learning and ensemble techniques for pattern discovery and decision making.	
CO5	Design and implement neural network models for solving complex learning problems.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	M	M	L	M	M	M	H	M	M	M	L
CO2	H	H	H	H	H	H	M	M	M	H	H	H	M	H	M
CO3	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	H	M	M	H	H	H	H	H	M
CO5	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

ADVANCED JAVA PROGRAMMING LAB

COURSE OBJECTIVES

- To provide practical exposure to core Java programming constructs.
- To implement object-oriented programming principles through application development.
- To design and develop GUI-based desktop applications using AWT and Swing.
- To build RESTful web applications using Spring Boot framework.
- To develop database-driven applications using Spring Data JPA.

LIST OF EXPERIMENTS

1. Develop a Java program to perform arithmetic operations using control statements, loops, arrays (1D & 2D), and command-line arguments.
2. Implement a menu-driven application to demonstrate object-oriented concepts such as classes, constructors, method overloading, method overriding, inheritance, and use of this and super.
3. Design a package-based Java application that demonstrates interfaces, abstraction, static members, and basic exception handling with user-defined exceptions.
4. Develop a file-handling program using byte streams and character streams to read, write, and append data to files with proper exception management.
5. Create a GUI-based application using AWT to perform simple form validation and event handling using action and item listeners.
6. Develop a Swing-based desktop application (e.g., Student Information System or Calculator) using JFrame, layout managers, mouse/key events, and adapter classes.
7. Create a basic Spring Boot application using Spring Initialize and Maven/Gradle to expose a simple REST endpoint.
8. Develop a RESTful CRUD application using Spring Boot with MySQL/H2 database integration using Spring Data JPA.
9. Implement exception handling and global error management in a Spring Boot REST application using @Controller Advice.
10. Develop a mini micro service-based application with multiple REST endpoints and database connectivity, demonstrating profiles and basic actuator endpoints.

COURSEOUTCOMES

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

CO1:	Implement Java programs using control structures, arrays, methods, and string handling.	K1-K6
CO2:	Develop object-oriented applications demonstrating inheritance, polymorphism, interfaces, and exception handling.	
CO3:	Design GUI-based desktop applications using AWT and Swing with event handling mechanisms.	
CO4:	Develop RESTful web services using Spring Boot framework.	
CO5:	Build database-driven applications integrating Spring Boot with JPA and relational databases.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	H	M	L	M	M	L	H	H	M	M	L
CO2	H	H	H	M	H	M	L	M	M	L	H	H	M	M	L
CO3	H	M	H	H	H	H	M	M	M	M	H	H	H	H	M
CO4	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO5	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

WEB TECHNOLOGIES LAB

COURSE OBJECTIVES:

- To provide practical exposure to designing structured web pages using HTML5 and CSS3.
- To develop fundamental client-side programming skills using JavaScript and DOM manipulation.
- To introduce server-side application development using Node.js and Express framework.
- To enable integration of web applications with relational databases for basic CRUD operations.
- To familiarize students with simple authentication mechanisms and deployment practices.

LIST OF EXPERIMENTS

1. Create a basic HTML webpage demonstrating headings, paragraphs, lists, images, hyperlinks, tables, and forms.
2. Design a responsive webpage using CSS3 to apply selectors, box model, positioning, typography, and layout styling.
3. Develop a Bootstrap-based webpage implementing grid system, navigation bar, forms, and responsive components.
4. Write a JavaScript program to demonstrate variables, data types, operators, loops, conditionals, and functions.
5. Create a dynamic webpage using JavaScript DOM manipulation to validate form inputs and handle user events.
6. Develop a simple Node.js application to create a basic HTTP server and serve static HTML files.
7. Build a basic Express.js application with routing and middleware to handle GET and POST requests.
8. Design a simple RESTful API using Express to perform CRUD operations on in-memory data

(JSON object).

9. Connect a Node.js application with a SQL database to insert and retrieve records using basic queries.
10. Implement a simple login system using sessions and basic authentication concepts.

COURSE OUTCOMES

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

CO1	Design structured and responsive web pages using HTML5, CSS3, and Bootstrap.	K1-K6
CO2	Develop interactive client-side applications using JavaScript and DOM manipulation techniques.	
CO3	Implement basic server-side applications using Node.js and Express framework.	
CO4	Integrate web applications with relational databases to perform CRUD operations.	
CO5	Develop simple full stack applications incorporating authentication and deployment concepts.	

K1-Remember, K2-Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	M	H	M	L	M	M	L	H	H	M	M	L
CO2	H	H	H	M	H	M	L	M	M	M	H	H	H	M	M
CO3	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	M	M	M	H	H	H	H	H	H
CO5	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

Subject Code: 26UPCSC1L10

Credits: 02

**ADVANCED MACHINE LEARNING TECHNIQUES WITH AI ASSISTED
CODING LAB**

COURSE OBJECTIVES

- To provide practical exposure to implementing advanced machine learning algorithms using standard datasets.
- To develop the ability to perform data preprocessing, exploratory analysis, and feature engineering for real-world data.
- To enable students to evaluate and compare machine learning models using appropriate performance metrics.
- To integrate AI-assisted coding techniques for efficient model development, optimization, and debugging.
- To design and deploy machine learning applications with user interfaces using frameworks such as Streamlit or Flask.

LIST OF EXPERIMENTS

Implement the following using Python 3.4 and above.

1. Implement data preprocessing techniques such as handling missing values, normalization, and encoding of categorical data using a dataset such as the Iris dataset available in scikit-learn or from the UCI repository: <https://archive.ics.uci.edu/ml/datasets/iris>
2. Perform exploratory data analysis and visualization to identify patterns and trends using the Wine dataset from scikit-learn or UCI:
<https://archive.ics.uci.edu/ml/datasets/wine>
3. Apply feature selection and dimensionality reduction techniques using the Breast Cancer dataset available in scikit-learn or UCI:
[https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+\(diagnostic\)](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic))
4. Implement Linear Regression and evaluate performance using Mean Squared Error and R^2 score with the California Housing dataset.
Dataset: scikit-learn / https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_california_housing.html
5. Implement classification models such as Logistic Regression, k-Nearest Neighbour, and Decision Tree using the Titanic dataset and evaluate their performance.
Dataset: <https://www.openml.org/d/40945>
6. Develop a Support Vector Machine classification application using AI-assisted coding and create an interactive interface using Streamlit or Flask with the Digits dataset.
Dataset: scikit-learn / https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html
7. Design a clustering-based customer segmentation application using AI-assisted coding and visualize clusters through a Streamlit or Flask interface using the Mall Customers dataset.
Dataset: <https://www.kaggle.com/datasets/vjchoudhary7/customer-segmentation-tutorial>
8. Develop an ensemble learning-based prediction system using AI-assisted coding and deploy it with a simple front-end interface using Streamlit or Flask based on the Heart Disease dataset.
Dataset: <https://archive.ics.uci.edu/ml/datasets/heart+disease>

9. Build an Artificial Neural Network-based digit recognition application using AI-assisted coding and provide a user interface using Streamlit or Flask with the MNIST dataset.
Dataset: tensorflow / <https://www.openml.org/d/554>
10. Develop a complete end-to-end machine learning web application using AI-assisted coding, including preprocessing, model training, and prediction with a user interface using Streamlit or Flask based on the Adult dataset.
Dataset: <https://archive.ics.uci.edu/ml/datasets/adult>

COURSE OUTCOMES:

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

CO1	Apply data preprocessing, exploratory analysis, and feature engineering techniques on real-world datasets.	K1-K6
CO2	Implement and evaluate supervised and unsupervised machine learning algorithms using appropriate metrics.	
CO3	Develop machine learning models using AI-assisted coding tools for improved efficiency and performance.	
CO4	Design and deploy machine learning applications with interactive user interfaces using Streamlit or Flask.	
CO5	Analyze model results and interpret outcomes for real-world decision-making applications.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	H	M	L	M	M	L	H	H	M	M	L
CO2	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO3	H	H	H	H	H	H	M	M	M	H	H	H	H	H	M
CO4	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H
CO5	H	H	H	H	H	H	H	M	M	H	H	H	H	H	H

H-High; M-Medium; L-Low

EXTENSION ACTIVITY**Course Description**

This course focuses on developing a **working prototype or model addressing a real-world societal need**, encouraging students to apply technical knowledge for community benefit and sustainable development.

COURSE OBJECTIVES

1. To identify real-world societal problems and analyze their impact on communities.
2. To design and develop innovative prototypes or models using appropriate technologies.
3. To apply interdisciplinary knowledge for solving social, environmental, or economic challenges.
4. To promote teamwork, communication, and project management skills.
5. To encourage ethical responsibility and community engagement through technology-driven solutions.

COURSE OUTCOMES

CO	Course Outcome	Bloom's Level
CO1	Identify and analyze societal problems requiring technological solutions.	K1- K6
CO2	Design feasible solutions and develop working prototypes or models.	
CO3	Apply appropriate tools and technologies to address real-world challenges.	
CO4	Demonstrate teamwork, communication, and project presentation skills.	
CO5	Evaluate the societal impact and sustainability of the developed solution.	

PEACE EDUCATION

Course Outcomes (COs)

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

CO1: Understand the importance of Peace (Level: Understand)

CO2: Explain how to recover the state of Peace (Level: Understand)

CO3: Internalize and Practice the Values of Peace (Level: Apply)

CO4: Understand sustainable Peace and the role of oneself (Level: Understand)

CO5: Become the peace messenger who spreads the culture of Peace (Level: Apply)

Unit I:

Introduction to Peace Education and HWPL- Aims and objectives of Peace Education – Development of Peace work of HWPL – Making groups for assignments – Rapport building

Unit II:

Finding Peace, and me Diversity – Harmony – Original State of All Creation – Connectivity – Value – Role – Duty – Interpersonal relationship – Greed – Love – Order

Unit III:

Peace Values Gratitude – Consideration – Sacrifice – Understanding – Forgiveness – Respect for Parents, Teachers and Peace – Scripture – Cultural sphere

Unit IV:

Peace Citizen Heritage – World Peace – Great legacy; a case of Peace – Law – Law-abiding Spirit – Treaty – DPCW – NGO; HWPL – Peace Citizen – Courage – Peace-loving Heart – Messenger of Peace

Unit V:

Peace Messenger Peace Messenger – Peace experience – Will for Peace

Tasks and Assignments:

- Writing an Essay on World Peace and Gratitude
- Engaging in debates on topics related to Peace
- Group Project for practicing Peace Values
- Group Project to promote the importance of Peace

References:

<https://www.un.org/en/>

<https://www.youtube.com/watch?v=18LhBZc6Sn4>

<https://www.hwpl.kr/language/en/home-hwpl-en/>

Heavenly Culture World Peace Restoration of Light (2022), Road to Peace

Oh Ik-soo (1996), Youth Group Counseling

ELECTIVES

VERTICAL I: Artificial Intelligence and Data Science (AIDS)

26UPCSC1E01	Generative AI
26UPCSC1E02	Advanced Large Language Models
26UPCSC1E03	Agentic AI

Subject Code: 26UPCSC1E01

Credits:04

GENERATIVE AI

COURSE OBJECTIVES

- Understand the basics of Generative AI and how data is used in real-world applications.
- Learn how Large Language Models work and how they are used in practice.
- Get familiar with different generative models like GANs, Diffusion Models, and Transformers.
- Learn how to improve models using fine-tuning and prompt engineering.
- Understand the importance of ethics, safety, and responsible use of AI.

Unit I: AI Fundamentals

AI Fundamentals – Learning Approaches in Machine Learning – Supervised, Unsupervised, Semi-supervised and Reinforcement Learning – Representation and Embedding – Training Strategies and Performance Improvement

Unit II: GenAI Foundation

Mathematics for AI – Deep Learning Concepts – Gradient Descent and Backpropagation – Regularization Techniques – Introduction to Generative AI – Prompt Engineering – Ethics and Use Cases

Unit III: GenAI for Images

Computer Vision and CNN – Convolution, Pooling and Fully Connected Layers – Autoencoders and Variational Autoencoders – Image Segmentation (U-Net) – Generative Models for Images – GAN and Diffusion Models

Unit IV: GenAI for Text and ChatGPT

Natural Language Processing – Sequence Models (RNN, LSTM) – Transformer Architecture – Tokenization and Embeddings – ChatGPT and OpenAI Tools – Prompt Engineering Techniques

Unit V: LLM Frameworks and Applications

LangChain and LLM Frameworks – Vector Stores and Retrieval Augmented Generation (RAG) – LLMops and Model Deployment – Optimization Techniques – Enterprise Applications of GenAI – Responsible AI and Sustainability

TEXT BOOKS

1. Sabeesan, K., Sivagamisundari, S., & Dutta, N. *Generative AI for Everyone: Deep Learning, NLP, and LLMs for Creative and Practical Applications*. BPB Publications. 2025

Reference Books

2. David Baum, *Generative AI and LLMs For Dummies, Snowflake Special Edition*, Wiley, 2024.

Unit -1-2

3. Carlos Rodriguez, *Generative AI Foundations in Python*, Packt Publishing, 2024.

Unit-3-5

COURSE OUTCOMES:

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

CO1	Understand the basics of enterprise-level Generative AI and data	K1-K6
CO2	Explain how Large Language Models work, including their lifecycle and usage..	
CO3	Understand and use different generative models like GANs, Diffusion, and Transformers.	
CO4	Use fine-tuning and prompt engineering to improve AI models for specific tasks.	
CO5	Understand ethics, risks, and responsible use of Generative AI	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	M	H	M	L	L	L	M	H	M	L	M	L
CO2	H	H	H	H	H	H	M	M	L	M	H	H	M	H	M
CO3	H	H	H	H	H	M	L	L	L	M	H	H	M	H	M
CO4	H	H	H	H	H	M	M	M	L	M	H	H	H	H	H
CO5	M	H	M	H	M	H	H	H	M	M	M	H	H	H	H

H-High; M-Medium; L-Low

ADVANCED LARGE LANGUAGE MODELS

COURSE OBJECTIVES

- To understand the fundamental concepts behind Large Language Models (LLMs).
- To implement transformer-based GPT architecture from scratch.
- To perform pre training and fine-tuning of LLMs using PyTorch.
- To evaluate generative models using appropriate metrics.
- To develop practical skills in building and customizing LLM systems.

UNIT I: Foundations of Large Language Models

Understanding Large Language Models – Applications of LLMs – Stages of Building and Using LLMs – Introducing the Transformer Architecture – A Closer Look at the GPT Architecture – Working with Text Data – Tokenizing Text – Byte Pair Encoding – Data Sampling with Sliding Window – Creating Token Embeddings – Encoding Word Positions

UNIT II: Attention Mechanisms and GPT Architecture

Coding Attention Mechanisms – Self-Attention Mechanism – Causal Attention – Multi-Head Attention – Implementing Self-Attention with Trainable Weights – Implementing Multi-Head Attention – Coding an LLM Architecture – Layer Normalization – Feed Forward Networks – Transformer Blocks – Coding the GPT Model

UNIT III: Pretraining Large Language Models

Pretraining on Unlabeled Data – Evaluating Generative Text Models – Training an LLM – Training and Validation Loss – Decoding Strategies – Temperature Scaling – Top-k Sampling – Text Generation – Saving Model Weights – Loading Pretrained Weights

UNIT IV: Fine-Tuning Large Language Models

Fine-Tuning for Classification – Preparing the Dataset – Creating Data Loaders – Adding a Classification Head – Calculating Loss and Accuracy – Fine-Tuning on Supervised Data – Instruction Fine-Tuning – Preparing Instruction Dataset – Extracting Responses – Evaluating Fine-Tuned LLM – Parameter-Efficient Fine-Tuning

UNIT V: PyTorch for Large Language Models

Introduction to PyTorch – Tensors – Automatic Differentiation – Neural Network Modules – Dataset and DataLoader – Model Training – GPU Utilization – Optimization Techniques – Building Models in PyTorch

TEXT BOOKS

1. Sebastian Raschka, **Build a Large Language Model (From Scratch)**, Manning Publications, 2025.

REFERENCE BOOKS

1. TanmoyChakraborty, Introduction to Large Language Models, Wiley Publications
2. Jay Alamar, Maarten Grootendorst, Hands on Large Language Models, Language Understanding and Generation, Shroff Publishers.
3. Amaratunga, Thimira, Understanding Large Language Models: Learning Their Underlying Concepts and Technologies, Springer Verlag, 2023
4. Raj Arun R, Mastering Large Language Models with Python, AVA publications 2024

COURSE OUTCOMES:

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

CO1	Explain the theoretical foundations of Large Language Models, transformer architecture, and generative AI mechanisms.	K1-K6
CO2	Implement data preprocessing, tokenization, embedding techniques, and attention mechanisms using PyTorch.	
CO3	Design and build GPT-based architectures and perform pretraining using next-word prediction objectives.	
CO4	Apply fine-tuning strategies for classification and instruction-following tasks using labeled datasets.	
CO5	Evaluate, optimize, and deploy LLM models considering performance, ethical, and societal aspects.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	L	L	M	L	L	M	M	H	M	M	L	L
CO2	H	H	M	L	M	H	L	L	L	L	H	M	M	M	L
CO3	H	H	M	M	H	H	M	L	L	L	H	H	H	M	L
CO4	H	H	M	M	H	H	M	M	L	M	H	H	H	M	M
CO5	M	H	H	M	M	H	M	H	H	H	M	H	H	H	H

H-High; M-Medium; L -Low

AGENTIC AI

COURSE OBJECTIVES

- To understand the theoretical foundations of generative and agentic AI systems.
- To analyze intelligent agent architectures and reasoning mechanisms.
- To design and implement LLM-powered autonomous agents.
- To explore multi-agent collaboration and workflow orchestration.
- To evaluate trust, safety, and ethical considerations in agentic AI deployment.

UNIT I: Foundations of Generative AI and Agentic Systems

Fundamentals of Generative AI – Introduction to generative AI – Types of generative AI models – VAEs – GANs – Autoregressive models and Transformer architecture – LLM-powered AI agents - Applications of generative AI - Principles of Agentic Systems: Technical requirements - Understanding self-governance, agency, and autonomy - Reviewing intelligent agents and their characteristics - Exploring the architecture of agentic systems – Understanding multi agent systems- Technical requirements Knowledge representation in intelligent agents- Reasoning in intelligent agents- Decision-making and planning in agentic systems- Enhancing agent capabilities with generative AI

UNIT II: Reflection and Tool Use in Agents

Reflection and Introspection in Agents – The importance of reflection in agents – Introspection in intelligent agents – Implementing reflective capabilities – Meta-reasoning – Self-explanation – Self-modeling – Use cases and examples -Enabling Tool Use and Planning in Agents – Understanding the concept of tool use in agents – Tool and function calling – Defining tools for agents – Types of tools – Planning algorithms for agents – Integrating tool use and planning – Exploring practical implementations

UNIT III: Multi-Agent Coordination and Design Techniques

Exploring the Coordinator, Worker, and Delegator Approach – Understanding the CWD model – Designing agents with role assignments – Roles and responsibilities of each agent – Communication and collaboration between agents – Implementing the CWD approach in generative AI systems - Coordination mechanism – Negotiation and conflict resolution – Knowledge sharing – Effective Agentic System Design Techniques : Focused system prompts and instructions for agents – Defining objectives – Task specifications – Contextual awareness – State space representation – Environment modeling – Agent memory architecture and context management – Short-term memory – Long-term memory – Episodic memory – Sequential and parallel processing in agentic workflows

UNIT IV: Trust and Ethical AI Systems

Building Trust in Generative AI Systems Importance of trust in AI — Techniques for establishing trust – Transparency and explainability – Dealing with uncertainty and biases – User control and consent – Ethical development and responsibility – Implementing transparency and explainability -Handling uncertainty and biases - Managing Safety and Ethical Considerations : Understanding potential risks and challenges – Adversarial attacks – Bias and discrimination – Misinformation and hallucinations –

Data privacy violations – Intellectual property risks – Ensuring safe and responsible AI – Exploring ethical guidelines and frameworks – Human-centric design – Accountability and responsibility – Privacy and data protection- Addressing privacy and security concerns

UNIT V: Applications and Future Directions

Common Use Cases and Applications – Creative and artistic applications – Natural language processing and conversational agents – Robotics and autonomous systems – Decision support and optimization – Conclusion and Future Outlook – Recap of key concepts – Emerging trends and research directions – Artificial general intelligence – Challenges and opportunities

TEXT BOOKS

1. Anjanava Biswas & Wrick Talukdar, *Building Agentic AI Systems: Create Intelligent, Autonomous AI Agents that can Reason, Plan, and Adapt*, Packt Publishing, 2025.

REFERENCE BOOKS

1. Artificial Intelligence: A Modern Approach, Russell & Norvig, 4th Ed., Pearson, 2021
2. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations, Shoham & Leyton-Brown, CUP, 2009
3. Reinforcement Learning: An Introduction, Sutton & Barto, 2nd Ed., MIT Press, 2018

COURSE OUTCOMES:

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

CO1	Explain the theoretical foundations of Generative AI, Large Language Models, and Agentic System architectures.	K1-K6
CO2	Analyze intelligent agent components including knowledge representation, reasoning, reflection, and planning mechanisms.	
CO3	Design and implement LLM-powered autonomous agents with tool use and multi-agent coordination frameworks.	
CO4	Evaluate trust, transparency, safety, and ethical considerations in the deployment of agentic AI systems.	
CO5	Develop real-world agentic AI solutions for complex applications using structured workflows and optimization techniques.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	H	M	L	L	L	–	H	H	M	M	L
CO2	H	H	H	H	H	M	L	L	L	–	H	H	H	M	M
CO3	H	H	H	H	H	H	M	M	M	–	H	H	H	H	M
CO4	M	H	M	H	M	H	H	H	M	–	M	H	M	H	H
CO5	H	H	H	H	H	M	M	M	H	–	H	H	H	H	H

H-High; M-Medium; L-Low

VERTICAL II: Full Stack Development for IT

26UPCSC1E04	App Development
26UPCSC1E05	UI and UX Design
26UPCSC1E06	Software Testing and Automation

Subject Code: 26UPCSC1E04

Credits:04

APP DEVELOPMENT

COURSE OBJECTIVES

- To understand the fundamentals of web and mobile application development paradigms.
- To develop native mobile applications using platform-specific technologies.
- To design and implement hybrid and cross-platform mobile applications using modern frameworks.
- To analyze non-functional characteristics such as performance, maintainability, usability, and scalability of mobile application frameworks.
- To develop deployable mobile and web applications with database integration and cloud connectivity.

UNIT I

Introduction to Flutter: Flutter: Introduction – Why Flutter— The other options – Native solutions – Developing in Flutter: The Flutter toolchain – The Flutter SDK – IDEs – IDE DevTools – Emulators – Keeping the tools up to date – The Flutter development process – Scaffolding the app and files – Running your app

UNIT II

Foundational Flutter: Everything Is Widgets : UI as code – Built-in Flutter widgets – Value widgets – Layout widgets – Navigation widgets – Other widgets – How to create your own stateless widgets – Widgets have keys – Passing a value into your widget – Stateless and Stateful widgets – Value Widgets :The Text widget – The Icon widget – The Image widget – Embedded images – Network images – Sizing an image – Input widgets – Text fields – Form widget – FormField widget

UNIT III

Responding to Gestures: Meet the button family – RaisedButton – FlatButton and IconButton – FloatingActionButton – CupertinoButton – Dismissible – Custom gestures for your custom widgets – GestureDetector widget – Laying Out Your Widgets – MaterialApp widget – The Scaffold widget – The AppBar widget – SafeArea widget – SnackBar widget – ListView widget – Container widget – Stack widget – GridView widget – The Table widget

UNIT IV

Navigation and Routing: Stack navigation – Navigating forward and back – Drawer navigation – The Drawer widget – Tab Navigation – TabController – TabBarView – TabBar and Tabs – The Dialog widget – showDialog() and AlertDialog – Styling Your Widgets : Thinking in Flutter Styles – Styling Text – TextStyle – Custom fonts – Container decorations – Border – BorderRadius – BoxShape – Themes – Managing State : Define State – StatefulWidget – Passing state down – Lifting state back up – InheritedWidget – BLoC – Provider – Redux

UNIT V

Flutter App Can Work with Files : Including libraries in your Flutter app – Futures, async, and await – Including a file with your app – Writing a file – Reading a file – Using JSON – Shared preferences – Making RESTful API Calls with HTTP : API call – HTTP GET or DELETE request – HTTP PUT, POST, or PATCH request – HTTP responses to widgets – FutureBuilder – Strongly typed classes – Using Firebase with Flutter : Introducing Firebase – Cloud Firestore – Cloud Functions – Authentication – Setting up Firebase – Using Firestore

TEXT BOOKS

1. **Rap Payne**, *Beginning App Development with Flutter: Create Cross-Platform Mobile Apps*, Apress, 2019. ISBN: 978-1484251805.

REFERENCE BOOKS

1. Booker Blunt, App Development with Flutter Build Cross Platform Apps: A Step-by-Step
2. Guide to Mastering Mobile App Development with Flutter, Independently Published, 2025
3. Alex Forrester, EranBoudjnah, AlexandruDumbravan, JomarTigcal, How to Build Android
4. Apps with Kotlin: A practical guide to developing, testing, and publishing your first Android apps, Packt Publishing Limited, 2023

COURSE OUTCOMES:

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

CO1	Explain the concepts of web, native, hybrid and cross-platform mobile application development architectures.	K1-K6
CO2	Develop native mobile applications using platform-specific tools and GUI components.	
CO3	Design and implement hybrid and cross-platform applications using modern frameworks such as React Native and Flutter.	
CO4	Integrate applications with backend services, databases and cloud platforms.	
CO5	Analyze and evaluate mobile application frameworks based on performance, security, scalability and maintainability.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	L	M	M	L	L	M	L	M	M	L	L	L
CO2	H	H	H	M	H	M	L	L	M	M	H	M	M	L	L
CO3	H	H	H	H	H	M	M	L	M	M	H	H	M	L	L
CO4	H	H	H	H	H	M	M	L	M	M	H	H	M	M	L
CO5	M	H	M	H	M	H	H	M	M	M	M	M	M	M	M

H-High; M-Medium; L-Low

USER INTERFACE AND USER EXPERIENCE (UI/UX) DESIGN

COURSE OBJECTIVES

- To provide foundational knowledge in User Interface (UI) and User Experience (UX) design principles.
- To understand the importance of UI and UX in digital product development.
- To explore research methods used in design thinking and user-centered design.
- To familiarize students with modern UI/UX tools and design workflows.
- To develop wireframes, prototypes, and conduct usability testing for digital applications.

UNIT –I

FOUNDATIONS OF DESIGN: UI vs. UX Design – Core Stages of Design Thinking – Divergent and Convergent Thinking – Brainstorming Techniques – Game storming – Observational Empathy – Human-Centered Design Principles – Problem Framing – Creativity in Design.

UNIT-II

FOUNDATIONS OF UI DESIGN : Principles of Visual Design – Elements of UI Design – Layout and Composition – Typography – Color Theory – UI Components and Patterns – Interaction Behaviors and Design Principles – Branding Fundamentals – Design Systems – Style Guides.

UNIT-III

FOUNDATIONS OF UX DESIGN : Introduction to User Experience – Importance of UX – Understanding User Experience – UX Design Process and Methodology – Research in UX Design – Research Tools and Methods – Understanding User Needs and Goals – Aligning UX with Business Goals – User Journey Mapping.

UNIT-IV

WIREFRAMING, PROTOTYPING AND TESTING: Sketching Principles – Rapid Sketching (Red Routes) – Responsive Design – Low-Fidelity Wireframing – High-Fidelity Mockups – Creating Workflows – Building Interactive Prototypes – Design Tools and Software – Interaction Patterns – Conducting Usability Testing – Evaluative User Research Methods – Synthesizing Test Findings – Prototype Iteration.

UNIT –V

RESEARCH, IDEATION AND INFORMATION ARCHITECTURE: Problem Identification and Writing Problem Statements – Selecting Appropriate Research Methods – Creating Personas – Solution Ideation – Creating User Stories – Scenario Development – Flow Diagrams – Flow Mapping – Information Architecture – Structuring Content for Usability.

TEXT BOOKS

1. **Joel Marsh**, *UX for Beginners: A Crash Course in 100 Short Lessons*, O'Reilly Media, 2022.
2. **Jon Yablonski**, *Laws of UX: Using Psychology to Design Better Products & Services*, O'Reilly Media, 2021.

REFERENCE BOOKS

1. **Jenifer Tidwell, Charles Brewer, Aynne Valencia**, *Designing Interfaces*, 3rd Edition, O'Reilly Media, 2020.
2. **Steve Schoger, Adam Wathan**, *Refactoring UI*, 2018.
3. **Steve Krug**, *Don't Make Me Think, Revisited: A Common Sense Approach to Web & Mobile Usability*, 3rd Edition, New Riders, 2015.

COURSE OUTCOMES:

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

CO1	Design and build user interfaces for web and mobile applications using UI design principles.	K1-K6
CO2	Evaluate and analyze the UX design of products or applications based on usability and user-centered design principles.	
CO3	Apply UX research methods and demonstrate UX skills in product development.	
CO4	Implement sketching and ideation techniques for interface design.	
CO5	Develop wireframes and interactive prototypes incorporating usability testing and iterative refinement.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	M	H	H	M	H	L	L	M	M	L	H	M	L	L	L
CO2	M	H	M	H	M	M	M	L	M	L	M	M	L	L	L
CO3	M	H	H	H	M	M	M	L	M	L	M	H	L	L	M
CO4	L	M	H	M	M	L	L	M	M	L	M	M	L	L	L
CO5	M	H	H	H	M	M	M	L	M	L	H	H	L	M	M

H-High; M-Medium; L-Low

SOFTWARE TESTING AND AUTOMATION

COURSE OBJECTIVES

- To understand fundamental concepts, principles, and importance of software testing.
- To learn various testing types and techniques used in software projects.
- To design effective test plans and test cases.
- To perform structured test execution and defect management.
- To understand test automation, Agile testing, and modern testing challenges.

UNIT I: Software Testing Fundamentals and Types

The Importance of Software Testing – Software Testing Types and Techniques – Functional Testing – Unit Testing – Integration Testing – System Testing – User Acceptance Testing – Nonfunctional Testing –Static Testing

UNIT II: Software Development Life Cycle and Test Planning

Software Development Life Cycle : Planning Phase – Requirements Gathering Phase – Design Phase – Development Phase – Deployment Phase – Maintenance Phase – The Role of Testing in the SDLC – Test Planning : Defining Testing Objectives – Determining the Scope of Testing – Selecting the Testing Approach – Identifying Testing Resources- Developing the Test Schedule – Defining Test Cases – Identifying Test Data – Defect Management Process – Stop Testing Criteria – Reviewing and Approving the Test Plan – Benefits of Test Planning – Test Plan Document – Test Planning Tools and Techniques

UNIT III: Test Design Techniques

Black-Box Testing – Equivalence Partitioning – Boundary Value Analysis – Decision Table Testing – State Transition Testing – Use-Case Testing – Pairwise Testing – Error Guessing – Exploratory Testing – Random Testing – Ad Hoc Testing – White-Box Testing – Statement Coverage – Branch Coverage – Path Coverage – Condition Coverage – Decision Coverage – Multiple Condition Coverage – Modified Condition/Decision Coverage – Loop Testing – Data Flow Testing – Static Testing- **Test Execution:** Test Execution Process – Test Environment Setup – Defect Reporting and Retesting – Regression Testing – Test Case Status Reporting – Test Case Completion – Techniques and Tools Used in Test Execution – Quality Metrics

UNIT IV: Test Automation, and Agile Testing

Test Automation – Benefits of Test Automation – Record and Playback Tools – Scripting Tools – Hybrid Tools – Frameworks – Automated Testing Tools – Automated Test Scripts – Testing in Agile Environment – Agile Testing Principles – Agile Testing Quadrants – Test-Driven Development – Behavior-Driven Development – Acceptance Test-Driven Development – Continuous Integration and Continuous Delivery – Test Automation in Agile – Agile Testing Best Practices

UNIT V: Challenges and Solutions

Challenges and Solutions in Software Testing – Lack of Clear Requirements – Impact of Lack of Clear Requirements on Testing – Mitigating the Impact of Lack of Clear Requirements – Time Constraints – Impact of Time Constraints on Software Testing – Strategies for Managing Time Constraints in Testing – Lack of Skilled Resources – Impact of Lack of Skilled Resources on Software Testing – Strategies for Managing Lack of Skilled Resources in Testing – Automation Challenges – Common Automation Challenges – Strategies for Addressing Automation Challenges – Communication and Collaboration – Change Management – Testing Across Platforms – Future of Software Testing – Risk-Based Testing

TEXT BOOK

1. Panagiotis Leloudas, *Introduction to Software Testing: A Practical Guide to Testing, Design, Automation, and Execution*, Apress, 2023.

REFERENCES

1. Glenford J. Myers, Corey Sandler, Tom Badgett, *The Art of Software Testing*, 3rd Edition, Wiley.
2. Paul C. Jorgensen, *Software Testing: A Craftsman's Approach*, 4th Edition, CRC Press.
3. Ron Patton, *Software Testing*, Sams Publishing.

COURSE OUTCOMES:

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

CO1	Explain the fundamental concepts, principles, and types of software testing across SDLC models.	K1-K6
CO2	Design comprehensive test plans and apply appropriate black-box and white-box test design techniques.	
CO3	Execute test cases, manage defects systematically, and evaluate software quality using metrics.	
CO4	Analyze nonfunctional requirements and perform performance, security, and reliability testing.	
CO5	Develop and implement automated test scripts and apply testing practices in Agile and CI/CD environments.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	M	H	H	M	H	L	L	M	M	L	H	M	L	L	L
CO2	M	H	M	H	M	M	M	L	M	L	M	M	L	L	L
CO3	M	H	H	H	M	M	M	L	M	L	M	H	L	L	M
CO4	L	M	H	M	M	L	L	M	M	L	M	M	L	L	L
CO5	M	H	H	H	M	M	M	L	M	L	H	H	L	M	M

H-High; M-Medium; L-Low

VERTICAL III: Cloud Computing and Data Centre Technologies

26UPCSC1E07	Cloud Computing
26UPCSC1E08	Virtualization
26UPCSC1E09	Cloud Security

Subject Code: 26UPCSC1E07

Credits: 04

CLOUD COMPUTING

COURSE OBJECTIVES

- To understand the fundamental concepts and evolution of cloud computing
- To analyze cloud service and deployment models
- To study virtualization, containerization, and cloud-native architectures
- To evaluate cloud pricing, security, and governance mechanisms
- To examine transition strategies and real-world public cloud implementations

UNIT I: Introduction and Historical Perspective

A Definition of Cloud Computing – Virtualization – Application virtualization – Server virtualization – Containerization – Cloud Services – Service-level agreement (SLA) – Service-level objectives (SLOs) – Service Models: Levels of Abstraction – Infrastructure-as-a-Service (IaaS) – Platform-as-a-Service (PaaS) – Software-as-a-Service (SaaS) – Business Process-as-a-Service (BPaaS) – Information-as-a-Service (INaaS) – Cloud Deployment Models – Public cloud – Private cloud – Community cloud – Hybrid cloud – Five Characteristics of Cloud Computing – Broad network access – On-demand self-service – Resource pooling – Rapid elasticity – Measured service – Internetworking – Protocol wars – TCP/IP protocol – Internet layers – Virtualization – Virtual machines – Hypervisor – VMware – Open-source virtualization – Xen – Containers – Cloud Computing evolution – SaaS emergence – AWS development – Public cloud providers

UNIT II: Types of Cloud Computing

Abstraction Levels – Infrastructure-as-a-Service (IaaS) – Platform-as-a-Service (PaaS) – Software-as-a-Service (SaaS) – Information-as-a-Service (INaaS) – Business Process-as-a-Service (BPaaS) – Deployment Models – Public clouds – Private clouds – Community clouds – Hybrid clouds – Types of Cloud – Personal cloud – Cloud of Things – Cloud Cells – Cloud Cell Patterns – Cloud Relationships – Encapsulation – Aggregation – Cloud-native concepts – Containers – Container orchestration – Kubernetes – Microservices foundations – DevOps – Continuous integration – Continuous delivery – Scalability – Resilience – Observability

UNIT III: Microservices and Cloud Paradigm

Microservices architecture – Service decomposition – API communication – Service discovery – Load balancing – Fault tolerance – Circuit breaker pattern – API gateway – Data management in microservices – Event-driven architecture – Cloud as paradigm shift – CapEx vs OpEx – Business transformation – IT agility – Scalability benefits – Cost optimization – Risks and challenges – Vendor lock-in – Organizational change

UNIT IV: Pricing, Data, and Security

Utility pricing – Subscription pricing – Pay-as-you-go – Reserved pricing – Spot pricing – Cost optimization strategies – Billing models – Cost management – Data in cloud – Data storage models – Structured data – Unstructured data – Data lifecycle – Data governance – Data consistency – Data replication – Big data in cloud – Security in cloud – Threat models – Identity and access management – Encryption – Network security – Compliance – Risk management – Security best practices

UNIT V: Cloud Migration and Applications

Migration strategies – Lift and shift – Refactoring – Replatforming – Cloud adoption frameworks – Change management – Risk mitigation- Amazon Web Services (AWS) – Microsoft Azure – Google Cloud Platform – Service comparisons – Use cases

TEXT BOOKS

1. Nayan B. Ruparelia, *Cloud Computing*, Revised and Updated Edition, MIT Press, 2023

REFERENCES

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, “Distributed and Cloud Computing, From Parallel Processing to the Internet of Things”, Morgan Kaufmann Publishers, 2012.
2. James Turnbull, “The Docker Book”, O’Reilly Publishers, 2014.
3. Krutz, R. L., Vines, R. D, “Cloud security. A Comprehensive Guide to Secure Cloud Computing”, Wiley Publishing, 2010.

COURSE OUTCOMES:

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

CO1	Explain the fundamental concepts, characteristics, and economic models of cloud computing.	K1-K6
CO2	Compare and evaluate cloud service models and deployment models for different organizational needs.	
CO3	Design cloud-native architectures using virtualization, containers, and micro services principles.	
CO4	Analyze pricing strategies, data governance, and security mechanisms in cloud environments.	
CO5	Evaluate cloud migration strategies, public cloud platforms, and future cloud technologies.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	L	L	M	L	L	L	L	L	M	L	L	L	L
CO2	M	H	M	M	M	M	L	L	L	L	M	M	L	L	L
CO3	M	H	H	H	H	M	L	L	M	L	H	H	M	L	M
CO4	M	H	M	H	M	H	M	M	L	L	M	H	M	M	L
CO5	M	H	H	M	M	M	M	M	M	M	H	H	M	M	H

H-High; M-Medium; L-Low

VIRTUALIZATION

COURSE OBJECTIVES

- To understand the fundamentals and significance of virtualization
- To study hypervisors, virtual machines, and virtual infrastructure components
- To gain hands-on knowledge in creating and configuring virtual machines
- To analyze performance tuning of CPU, memory, storage, and networking in virtual environments
- To explore virtualization availability, protection, and its role in cloud computing

UNIT I: Virtualization Fundamentals and Concepts

Understanding Virtualization – Describing Virtualization – Microsoft Windows Drives Server Growth – Explaining Moore’s Law – Understanding the Importance of Virtualization – Examining Today’s Trends – Virtualization and Cloud Computing – Hyperconverged Infrastructure – Understanding Virtualization Software Operation – Virtualizing Servers – Virtualizing Desktops – Virtualizing Applications- Describing a Hypervisor – Exploring the History of Hypervisors – Understanding Type 1 Hypervisors – Understanding Type 2 Hypervisors – Understanding the Role of a Hypervisor – Holodecks and Traffic Cops – Resource Allocation – Comparing Today’s Hypervisors – VMware ESX – Citrix Hypervisor (Xen) – Microsoft Hyper-V – Other Solutions

UNIT II: Virtual Machines

Describing a Virtual Machine – Examining CPUs in a Virtual Machine – Examining Memory in a Virtual Machine – Examining Network Resources in a Virtual Machine – Examining Storage in a Virtual Machine – Understanding How a Virtual Machine Works – Working with Virtual Machines – Understanding Virtual Machine Clones – Understanding Templates – Understanding Snapshots – Understanding OVF – Understanding Containers- Performing P2V Conversions – Investigating the Physical-to-Virtual Process – Hot and Cold Cloning – Loading Your Environment – Loading VMware Workstation Player – Exploring VMware Workstation Player – Loading VirtualBox – Building a New Virtual Machine

UNIT III: Installing Virtual Machines

Thinking About VM Configuration – Creating a First VM – Loading Windows into a Virtual Machine – Installing Windows 11 – Installing VMware Tools – Understanding Configuration Options – Optimizing a New Virtual Machine – Loading Linux into a Virtual Machine – Exploring Oracle VM VirtualBox – Installing Linux into a Virtual Machine – Installing VirtualBox Guest Additions – Understanding Configuration Options – Optimizing a New Linux Virtual Machine

UNIT IV: Resource Management in Virtual Machines

Understanding CPU Virtualization – Configuring VM CPU Options – Tuning Practices for VM CPUs – Choosing Multiple vCPUs vs. a Single vCPU – Hyperthreading – Working with Intel and AMD Servers – Understanding Memory Virtualization – Configuring VM Memory Options – Tuning Practices for VM Memory – Calculating Memory Overhead – Memory Optimizations – Understanding Storage Virtualization –

Configuring VM Storage Options – Tuning VM Storage – Understanding Network Virtualization – Configuring VM Network Options – Tuning Practices for Virtual Networks

UNIT V: Advanced Virtualization and Applications

Cloning a Virtual Machine – Working with Templates – Saving a Virtual Machine State – Creating a Snapshot – Merging Snapshots – Using Virtual Machine Tools – Understanding Virtual Devices – Configuring a CD/DVD Drive – Configuring a Floppy Disk Drive – Configuring a Sound Card – Configuring USB Devices – Configuring Graphic Displays – Configuring Other Devices – Increasing Availability – Protecting a Virtual Machine – Protecting Multiple Virtual Machines – Protecting Data Centers – Examining Virtual Infrastructure Performance Capabilities – Deploying Applications in a Virtual Environment – Understanding Virtual Appliances and vApps – Open Stack and Containers – Cloud and the Future of Virtualization

TEXT BOOKS

1. Matthew Portnoy, *Virtualization Essentials*, Wiley Publications, Third Edition, 2023

REFERENCES

1. David Marshall, Wade A. Reynolds, *Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center*, Auerbach
2. Chris Wolf, Erick M. Halter, “Virtualization: From the Desktop to the Enterprise”, APress, 2005.
3. James E. Smith, Ravi Nair, “Virtual Machines: Versatile Platforms for Systems and Processes”, Elsevier/Morgan Kaufmann, 2005.

COURSE OUTCOMES:

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

CO1	Explain the principles and architecture of virtualization technologies.	K1-K6
CO2	Compare hypervisor types and configure virtual machines effectively.	
CO3	Deploy and manage virtual machines using industry-standard tools.	
CO4	Optimize CPU, memory, storage, and network resources in virtual environments.	
CO5	Evaluate virtualization availability, protection mechanisms, and its integration with cloud computing.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	L	L	M	L	L	L	L	L	M	L	L	L	L
CO2	M	H	M	M	H	L	L	L	L	L	H	M	L	L	L
CO3	M	H	H	M	H	M	L	L	M	L	H	H	M	L	M
CO4	M	H	H	H	H	M	M	M	L	L	H	H	M	M	L
CO5	M	H	M	M	M	H	M	M	M	M	H	H	M	M	H

H-High; M-Medium; L -Low

CLOUD SECURITY

COURSE OBJECTIVES

- To understand the fundamentals of cloud computing and associated security challenges.
- To analyze security and privacy risks in cloud environments.
- To study cloud security architecture, cryptography, and data protection techniques.
- To explore identity management, access control, and cloud security assessment methods.
- To examine advanced cloud security concepts including blockchain and confidential computing.

UNIT I: Introduction to Cloud Computing and Security

Cloud Computing Overview – Service Models – Advantages of Cloud Computing – Disadvantages on Cloud Computing – Types of Cloud Computing – Key Security Challenges in Cloud Computing – Risk Assessment for Cloud Environments – Threat Modeling – Cloud Security Controls

UNIT II: Security and Privacy Risks in Cloud computing

Data breaches and data loss in cloud environments – Data breach – Data loss – Compliance and regulatory issues – Shared Responsibility Model – Legal considerations in cloud computing – Ethical considerations in cloud computing – Case studies on security and privacy breaches

UNIT III: Cloud Security Architecture and Data Protection

Cloud Security Architecture – Secure Network Design – Cloud Security Models – Data Storage and Processing Security – Securing Cloud-Based Applications – API – API Security – Cryptography – Key Management – Data Protection – Encryption – Decryption – Data backup – Disaster recovery

UNIT IV: Identity Management and Cloud Security Assessment

Authentication – Access Control – Identity and access management (IAM) – IAM technologies and tools – Single Sign-On (SSO) – Federation – Monitoring and Auditing – Security Assessment – Risk Management – Cloud security auditing – Cloud incident response – Cloud forensic

UNIT V: Cloud Security and Confidential Computing

Cloud security – Cloud app administrator identity – App deployment identity – Cloud admin identity – Blockchain approach – Blockchain's approach to preventing app compromises – Economy of security – Cloud security conclusions – Security vs. availability ANDs and ORs – Azure – Future appears serverless – Azure Security Services – Conclusions – Exercises – Different Clouds and Confidential Compute Promise – Introduction – Structure – Objectives – Cloud admin compromise should not be fatal – Promise of confidential computing – Bootstrapping Enclave proofs – Clean room setup – Identity provider for users, apps, and Enclaves – Compute and storage – Compute and storage architecture security considerations – Blockchain architectures – Features of Azure – Types of Enclaves – Different philosophies – Achieving public blockchain confidentiality – Homomorphic encryption – Tumblers – Other techniques – Blockchain architectures

TEXT BOOK

1. Mateu, T. (Ed.), Privacy and Security in Cloud Computing: Methods and Techniques, Intelliz Press, 2024
2. Rouskov, Y. R., Cloud Application Security Essentials with Azure: Azure application security, confidential computing, app identity, authentication, and authorization, BPB Publications, 2025.

REFERENCES

1. Thomas Erl, Robert Cope, Amin Naserpour Cloud Computing Design Patterns
2. Praveen Ayyappa Economics of Cloud Computing, LAP Lambert Academic Publishing
3. RajkumarBuyya, Christian Vechhiola, S. ThamaraiSelvi, Mastering Cloud Computing Foundations and Applications Programming
4. Enamul Haque, *Cloud Service Management and Governance – Smart Service Management in Cloud Era*, Enel Publications, 2020.
5. Thomas Erl, Ricardo Puttini, Zaigham Mohammad, Cloud Computing: Concepts, Technology & Architecture by 2013

COURSE OUTCOMES

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

CO1	Explain cloud computing concepts, service models, and security ch	K1-K6
CO2	Identify and analyze security and privacy risks in cloud computing	
CO3	Apply cloud security architecture, cryptography, and data protection techniques.	
CO4	Demonstrate identity management, access control, and cloud security assessment practices	
CO5	Evaluate advanced cloud security approaches such as blockchain and confidential computing.	

K1-Remember, K2-Understand, K3- Apply, K4-Analyze, K5-Evaluate, K6- Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	L	L	M	L	L	L	L	L	M	L	L	L	L
CO2	M	H	M	H	M	L	L	L	M	L	M	M	L	L	L
CO3	M	H	H	H	M	M	L	L	M	L	H	H	M	L	M
CO4	M	H	M	H	M	H	M	M	L	L	M	H	M	M	L
CO5	M	H	H	M	M	H	M	M	M	M	H	H	M	M	H

H-High; M-Medium; L -Low

VERTICAL IV: Cyber Security and Data Privacy

26UPCSC1E10	Network Security and Ethical Hacking
26UPCSC1E11	Digital and Mobile Forensics
26UPCSC1E12	Modern Cryptography and Blockchain Technologies

Subject Code: 26UPCSC1E10

Credits:04

NETWORK SECURITY AND ETHICAL HACKING

COURSE OBJECTIVES

- To understand networking fundamentals and traffic interception techniques.
- To study cryptographic foundations and secure communication protocols.
- To analyze social engineering and reconnaissance methodologies.
- To explore vulnerability discovery and exploitation techniques.
- To examine malware, password attacks, and web application vulnerabilities.

UNIT- I

NETWORK FUNDAMENTALS: Virtual Lab Setup – VirtualBox – pfSense firewall configuration – Internal LAN setup – Kali Linux – Metasploitable – Ubuntu lab configuration. Internet architecture – Packets – MAC addresses – IP addressing – CIDR notation – Routers and LAN – ARP protocol – ARP tables – ARP spoofing – Detecting ARP attacks – Traffic capture using Wireshark – Internet protocol stack – Packet inspection – TCP handshake – Reverse shell – Port scanning – Botnet fundamentals – SYN scan and detection.

UNIT -II

CRYPTOGRAPHY AND SECURE COMMUNICATION: Encryption – One-time pad – Pseudorandom generators – Insecure Block Ciphers Modes- Secure Block Ciphers Modes – File encryption and decryption – Email encryption. Public-key cryptography – Rivest–Shamir–Adleman Theory - The RSA Math– Optimal Asymmetric Encryption Padding (OAEP). Transport Layer Security (TLS) – Certificates and Certificate Authorities – Message authentication – Diffie-Hellman key exchange – Elliptic Curve Diffie-Hellman – Key derivation – TLS socket programming – SSL stripping – HSTS bypass

UNIT- III

SOCIAL ENGINEERING AND RECONNAISSANCE: Social engineering attacks – Phishing – Email spoofing – SMTP communication – DNS lookup – Fake websites – Deepfake fundamentals – Voice cloning basics. Open Source Intelligence (OSINT) – Link analysis – Maltego – Leaked credential databases – SIM jacking. Network scanning – nmap – Masscan – Shodan – Vulnerability databases – Vulnerability scanners – IPv6 considerations – NAT limitations – Writing simple OSINT and scanning tools.

UNIT- IV

VULNERABILITY DISCOVERY AND MALWARE: Case study: Heartbleed vulnerability – Writing exploits – Fuzzing fundamentals – Writing a basic fuzzer – Symbolic execution – Dynamic symbolic Password hashing – MD5 structure – Cracking password hashes – Salting – Building salted hash crackers – John the Ripper – Hashcat – Hydra – Brute force techniques – NoSQL injection – Burp Suite basics. Cross-Site Scripting (XSS) – Stored and Reflected XSS – OWASP Zed Attack Proxy – Browser Exploitation Framework (BeEF) – Cookie theft – Website exploitation case studies. execution – Angr framework overview. Trojans – Malware architecture – Implant development – Hiding malware in legitimate files – Encoding techniques – Windows, Linux and Android trojan concepts. Linux kernel modules – Rootkits – System call hooking – Hiding files and processes – Keylogger fundamentals.

UNIT -V

PASSWORD ATTACKS AND WEB EXPLOITATION: SQL Injection – Extracting database information – Writing SQL injection tools – SQLMap – HTTP request manipulation.

TEXT BOOKS

1. Daniel G. Graham, *Ethical Hacking: A Hands-on Introduction to Breaking In*, No Starch Press, 2021.

REFERENCE BOOKS

1. Michael T. Simpson et al., *Hands-On Ethical Hacking and Network Defense*, Cengage.
2. William Stallings, *Cryptography and Network Security*, Pearson.
3. Jon Erickson, *Hacking: The Art of Exploitation*, No Starch Press.

COURSE OUTCOMES

CO1	Explain networking fundamentals, traffic interception techniques, and ARP-based attacks.	K1-K6
CO2	Apply cryptographic principles and secure communication mechanisms such as TLS and Diffie-Hellman.	
CO3	Analyze social engineering and reconnaissance techniques to identify security weaknesses.	
CO4	Design and evaluate exploitation techniques including fuzzing, malware, and rootkits in controlled environments.	
CO5	Implement web application and password attack techniques such as SQL injection and XSS for vulnerability assessment.	

K1–Remember, K2–Understand, K3–Apply, K4–Analyze, K5–Evaluate, K6–Create

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	M	H	L	L	M	M	L	H	M	L	L	L
CO2	H	H	M	M	H	M	L	M	M	L	H	H	M	L	L
CO3	M	H	H	M	M	M	M	L	M	L	M	H	M	L	L
CO4	M	H	H	H	M	M	M	M	M	L	M	H	H	M	L
CO5	H	H	H	H	H	M	M	M	M	M	H	H	H	M	M

H-High; M-Medium; L-Low

DIGITAL AND MOBILE FORENSICS

COURSE OBJECTIVES

- To understand the principles and methodology of digital forensic investigation
- To establish and configure a forensic laboratory environment
- To perform evidence acquisition and forensic analysis across operating systems
- To investigate mobile, browser, memory, and cloud artifacts
- To prepare legally admissible forensic reports

UNIT -I

FUNDAMENTALS OF DIGITAL FORENSICS: Defining digital forensics – Digital forensics goals – Defining cybercrime – Sources of cybercrime – Computers in cybercrimes – Digital forensics categories – Computer forensics – Mobile forensics – Network forensics – Database forensics — Law enforcement – Civil litigation – Intelligence and counterintelligence – Digital forensics investigation types – Forensics readiness – Type of digital evidence –Locations of electronic evidence – Chain of custody – Examination process – Seizure – Acquisition –Essential Technical Concepts-Decimal (Base-10) – Binary – Hexadecimal (Base-16) – Hexadecimal (Base-64) – Character encoding schema – File carving – File structure – Digital file metadata – Timestamps decoder – Hash analysis – Calculate file hash – System memory – Types of computer memory storage – Primary storage — SSD – DCO and HPA – Considerations for data recovery – File system – NTFS – FAT – Environment for computing – Cloud computing – Windows versions – Internet protocol (IP) address – Getting an IP address

UNIT II: Storage and Forensic Lab Setup

Hard disk and file systems – File systems – Hard disk – Hard disk forensics – Analyzing the registry files - Digital Forensic Lab – Physical requirements – Environment controls – Digital forensic equipment – Forensic hardware – Office electrical equipment – Networked devices – Forensic workstation – Commercial digital forensic workstations – Forensic software applications – Commercial forensics tools – Open-source forensic tools – Linux distributions – Virtualization – Lab information management system (LIMS) – Lab policies and procedures – Documentation – Lab accreditation – Conclusion

UNIT III: Evidence Acquisition and Analysis

Raw format – Advanced forensic format – EnCase: Expert witness transfers – Other file formats – Validation of forensic imaging files – Live memory acquisition – Virtual memory: Swap space – Challenges acquiring RAM – Administration privilege – Live RAM capturer – Magnet RAM capture – FTK imager – Acquiring nonvolatile memory – Hard disk acquisition – Acquiring physical resources – Logical acquisition – Sparse acquisition – Capturing hard drives using FTK imager – Network acquisition – Limitations of a forensic tool - Arsenal Image Mounter – OSFMount – Autopsy – Analyzing RAM forensic image – Memoryze – Redline – Volatility framework

UNIT IV: System and Browser Forensics

Timeline analysis tools – File recovery – Undeleting files – Recycle bin forensics – Data carving – Associated user account action – Windows registry analysis – Windows registry architecture – Acquiring windows registry

– Registry examination – Windows registry program keys – USB device forensics – Most recently used list – Network analysis – Windows shutdown time – UserAssist forensics – Printer registry information – File format identification – Windows thumbnail forensics – Windows 10 forensics – Notification area database – Cortana forensics – Web browser forensics – Google chrome browser forensics – Top sites and shortcuts – Login data – Web data – Bookmarks – Bookmarks.bak – Cache folder – Mozilla Firefox Browser Forensics – Microsoft Edge browser forensics – Other Web browser investigation tools

UNIT V: E-mail Forensics and Anti-Forensics

E-mails around us – E-mail communication steps – E-mail protocols – Examine e-mail headers – Reveal header information – View Gmail headers – View Outlook mail header – View Mozilla Thunderbird headers – View Outlook mail client header – Analyzing e-mail headers – Determine the sender's geolocation and time zone Anti-forensics techniques – Digital Steganography – Text Steganography – Image Steganography – Audio-video Steganography – Network Steganography – Metadata manipulation – Encryption techniques – Disk encryption using open-source tools – Anonymity techniques – Digital forensic reports

TEXT BOOKS

1. Akashdeep Bhardwaj, Keshav Kaushik, Practical Digital Forensics: Forensic Lab Setup, Evidence Analysis, and Structured Investigation Across Windows, Mobile, and Cloud, BPB Publications, 2023

REFERENCE BOOKS

1. Rohit Tamma, Practical Mobile Forensics: Forensically investigate and analyze iOS, Android, and Windows 10 devices, 4th Edition , 2020
2. Gerard Johansen, Digital Forensics and Incident Response: Incident response techniques and procedures to respond to modern cyber threats, 2020
3. Deepanshu Khanna Digital Forensics and Incident Response: A practical guide to using Kali Linux for cyber investigations (English Edition) , 2024

COURSE OUTCOMES (COs)

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

CO1	Explain digital forensic investigation processes and legal frameworks	K1-K6
CO2	Perform forensic acquisition and analysis on Windows and Linux systems	
CO3	Analyze mobile and browser artifacts for digital evidence extraction	
CO4	Investigate volatile memory, network, and cloud-based evidence	
CO5	Prepare structured forensic reports and present digital evidence professionally	

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	H	M	M	M	H	M	H	L	M	H	M	L	L	L
CO2	H	H	H	H	H	M	M	M	L	M	H	H	M	L	M
CO3	H	H	H	H	H	M	M	M	L	M	H	H	M	M	M
CO4	H	H	H	H	H	H	M	H	M	M	H	H	M	M	H
CO5	M	M	H	M	M	H	M	H	H	H	M	M	M	M	H

H-High; M-Medium; L -Low

MODERN CRYPTOGRAPHY AND BLOCKCHAIN TECHNOLOGIES

COURSE OBJECTIVES

- To understand the theoretical foundations of modern cryptography.
- To study encryption techniques for data at rest and in transit.
- To explore emerging cryptographic domains including cloud, AI, and post-quantum cryptography.
- To understand the principles and architecture of blockchain systems.
- To examine smart contracts and decentralized applications.

UNIT –I

FUNDAMENTALS OF CRYPTOGRAPHY: Fundamentals of Cryptography – History of Cryptography – Historical Development of Cryptography – Need for Cryptography – Introduction to Cryptography – Definitions – Classical versus Modern Cryptography – Encoding, Hashing, and Encryption – Security Concepts – Types of Modern Cryptography – Steganography – Additional Techniques – Confusion and Diffusion – Refresher on Mathematics – Primer on Ciphers - Primer on Symmetric Cryptography - Symmetric Key Algorithms

UNIT –II

DATA ENCRYPTION AND WEB SECURITY: Primer on Data Security- Data-at-Rest Encryption- Primer on Web Security - Web Security Protocols- Implementing VPNs Using IPS - Securing Web-Based Applications - Public Key Infrastructure

UNIT- III

CLOUD, AI AND POST-QUANTUM CRYPTOGRAPHY: Primer on Cloud Cryptography- Encryption Key Management - Cryptography as a Service by Major Cloud Service Providers - Cryptography as a Service by Major Cloud Service Providers - AI for Cryptography- Cryptography for AI - Best Practices and Ethical Use of AI - Primer on Quantum Computing- Quantum Computing and Cryptography

UNIT –IV

BLOCKCHAIN FOUNDATIONS: Origins of Blockchain Technology – Distributed Systems – Centralized vs Decentralized Architecture – Bitcoin Predecessors – Bitcoin Experiment – Bringing Bitcoin to Life – Cryptocurrency Fundamentals

UNIT- V

BLOCKCHAIN EVOLUTION AND SMART CONTRACTS: Forks and Altchains – Bitcoin Improvement Proposals – Understanding Forks- Bitcoin Cash Fork- Altcoins- 2.0 Chains- Privacy focused cryptocurrency – Ripple and Stellar - Improving Bitcoin's Limited Functionality- b. Ethereum:

TEXTBOOKS

1. **Sandip Dholakia**, *Modern Cryptography: The Practical Guide*, Rheinwerk Publishing, **First Edition, 2024**.
2. **Lorne Lantz and Daniel Cawrey**, *Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications*, O'Reilly Media, **First Edition, 2021**.

REFERENCE BOOKS

1. Kumar Saurabh and Ashutosh Saxena, *Blockchain Technology*, Wiley Publication, 2020
2. Asharaf S., Sivadas Neelima, Franklin John, *Blockchain Technology: Algorithms and Applications*, Wiley Publication, 2023
3. Jean-Philippe Aumasson, *Serious Cryptography: A Practical Introduction to Modern Encryption*, no starch press, 2024

COURSE OUTCOMES

CO1	Explain the fundamental principles, mathematical foundations, and classical-to-modern evolution of cryptography.	K1-K6
CO2	Apply encryption techniques for securing data at rest and data in transit across computing environments.	
CO3	Analyze advanced cryptographic domains including cloud cryptography, AI-integrated cryptography, and post-quantum cryptographic models.	
CO4	Examine blockchain architecture, consensus mechanisms, and cryptocurrency fundamentals.	
CO5	Design and evaluate smart contract-based decentralized applications within blockchain ecosystems.	

(K1–Remember, K2–Understand, K3–Apply, K4–Analyze, K5–Evaluate, K6–Create)

MAPPING WITH PROGRAMME OUTCOMES

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	H	M	M	M	M	L	L	M	M	L	H	M	L	L	L
CO2	H	H	M	M	H	M	L	M	M	L	H	H	M	L	L
CO3	M	H	H	M	M	M	M	M	M	L	M	H	H	M	L
CO4	H	H	H	H	H	M	M	M	M	M	H	H	H	M	M
CO5	H	H	H	H	H	M	M	M	H	M	H	H	H	H	M

H-High; M-Medium; L-Low

NON MAJOR ELECTIVE –II

Subject Code: 26UPCSC1N01

Credit:02

ADVANCED MICROSOFT OFFICE LAB

LIST OF EXPERIMENTS

MS-Word:

1. Design an admission/enquiry form using shapes, textboxes, colors, tables with formatting options.
2. Design a text book with cover page, content page and text using indenting options, rulers, page layout, header/footer and hyperlinks.
3. Design News paper advertisement with images and texts.
4. Design mark statement copy using tables, images and watermarks.
5. Design conference/ seminar in variation with logos, formatting options, margins and borders.
6. Write a Research article with Chart, Tables, Symbols, Equations and References.

MS-Excel:

7. Design an application for student Exam Result using Data validation, Aggregate functions and Conditional formatting.
8. Prepare Cost-Benefit Analysis for an organization using Statistical tools.
9. Develop an application to predict population of a city using analysis Macros.

MS-Access:

10. Design an Employee payroll system with Forms and Reports using Macros.
11. Develop relational integrity databases.
12. Develop an application which automatically updates a table using Triggers.
13. Design an inventory database and generate conditional report.

MS-PowerPoint:

14. Prepare a presentation with embedding multimedia objects.
15. Prepare presentation with slide layout, animations, font effects, hyperlinks.

MS-Publisher:

15. Prepare an academic calendar for an institution.

REFERENCE BOOKS:

1. Lisa A. Bucki, “MSOffice 2013 Bible”, Wiley Publications, 2013.
2. Richard Mansfield, “Mastering VBA for Microsoft office 2016”, Wiley Publications, 2016.
3. Wayne L. Winston, “Microsoft Excel Data Analysis and Business Modeling”, PHI, 2017.
4. Manisha Nigam, “Data Analysis with Excel”, BPB Publications, 2019.
5. Michael Alexander and Dick Kusleika, “Excel Power Programming with VBA”, Wiley Publications, 2016.
6. Michael Alexander and Dick Kusleika, “Access the Comprehensive tutorial guide”, Wiley Publications, 2016.

NONMAJORELECTIVE–II

Subject Code: 26UPCSC1N02

Credit: 02

BIOPYTHON PROGRAMMING LAB

LIST OF EXPERIMENTS

1. Program to implement Functions.
2. Program to perform Basic Operations on Sequence objects.
3. Program to perform Operations on Sequence an notation objects.
4. Program to perform Operations on Sequence Input/Output.
5. Program to perform Operations on Multiple Sequence Alignment objects.
6. Program to perform Operations on BLAST.
7. Program to perform Sequence motif analysis.
8. Program to perform Cluster analysis.
9. Program to perform Supervised learning methods.
10. Program to perform Genome Data visualization.

REFERENCE BOOK

1. Via,A., Rother, K.,& Tramontano,A. (2014). Managing your biological data with Python. Chapman and Hall/CRC.
2. Rocha, M.,& Ferreira,P.G.(2018)Bioinformatics Algorithms: Design and Implementation in Python. Academic Press.
3. Chun,W.(2001).Core python programming (Vol.1).Prentice Hall Professional.

Value Added Course

COURSE: 26UPCSCVA01

MS-Office (Excel)

[Credits: 4]

Course Objectives:

This course should enable the students:

- Give students an in-depth understanding of how the office tool is essential in business, education and society
- To understand the basics and advanced features of Excel spread program
- To improve the Data interpretation and analysis skills of the learner
- Learner get familiar in working of Statistical functions to analyze the data

1. Create an excel sheet and perform the following operation

i) Creating and running a macro. ii) Assigning button to a defined macro. iii) Editing a macro.

2. Create a table and apply the various formatting properties and generate the chart for the table

3. Create a table and apply the various mathematical functions.

4. Prepare Excel worksheet with chart for Employee salary with the following information

(i) Name of People with their age (at least 10)

(ii) Number of years they have worked with salary per month

(iii) Show the HRA, MA, TA and Hard allowances

Calculate the total, average, maximum, minimum salary of employees and give the chart for salary

5. Create Excel worksheet for Wedding cost estimation with the following information

(i) Name of guest along with who invited them (Bride / Groom) (at least 10) (ii) Guests may be divided into adult and children

(iii) Cost of the reception for adult Rs 650, children 60% of adult cost and 183% of GST

Calculate the total number of guests, number of adult and children and total reception cost. Show the graph for adult and children.

6. Prepare Excel table for Balance check book with the following information Ck# (Check book number), date, item description (at least 10 transactions), Debit and Credit

Calculate the balance using the formula $\text{Balance} = \text{Credit} - \text{Debit}$

7. Prepare Excel worksheet table for Cost-Benefit analysis of stadium for 10 years with the following information

Year, Costs, Benefits, Total benefits, Discount factor, Present Value

107

Calculate the total benefits ($\text{Total benefits} = \text{Benefits} - \text{Cost}$), discount factor ($\text{discount factor} = 1/(1+0.1)^{\text{year}}$), Present value

($\text{Present value} = \text{total benefits} * \text{discount factor}$).

8. Develop a spread program for the following case scenario

One day at a family dinner, your dad and your uncle got into a debate about which basketball team is better. Your dad is a loyal fan of the State University while your uncle graduated from the University of State, a big rival of the State University. Your great aunt chimed in that the University of Next- State's basketball team is the best.

Compare the performance of the basketball teams and you will bring in your analysis to the next family dinner.

9. Develop a spread program for the following case scenario

National Park Services (NPS) is a federal agency of the United States Government responsible for managing all U.S. National parks, many American national monuments, and many other conservation and historical properties. Information about visitation at units of the NPS is publicly available and may be downloaded at <https://irma.nps.gov/Stats/> You work for a company that specializes in guided tours of national parks across the United States. As an assistant to the Director of Marketing, you were asked to analyze some visit data at selected national parks. Your boss is specifically interested in the number of visitors as well as the trend of recreational visits over the years. The information will be used to develop marketing strategies, promotional packages, as well as advertising campaigns for the next five years.

10. Develop a spread program for the following case scenario

Headquartered in Memphis, TN, Grenadier Super Store (GSS) specializes in office supplies and furniture. The company's customers range from individual consumers and small businesses (retail), to corporate organizations (wholesale) located in the United States and Canada.

You are an intern working for the Canada division of GSS. Your supervisor has given you an Excel file containing Order data from 2009-2012 and he would like you to analyze Orders/Customers/Sales data using PivotTables and Pivot Charts.

References

1. Wayne L. Winston – *Microsoft Excel Data Analysis and Business Modeling* – PHI Learning Pvt. Ltd. – 7th Edition – 2024
2. Bill Jelen & Michael Alexander – *Excel 2019 Bible* – Wiley – 1st Edition – 2018
3. Michael Alexander & John Walkenbach – *Excel Dashboards and Reports* – Wiley – 1st Edition – 2010
4. Conrad Carlberg – *Statistical Analysis with Excel* – Pearson – Latest Edition – 2020
5. Hector Guerrero – *Excel Data Analysis* – Springer – 2nd Edition – 2019

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

1. Understand the basics of MS-Excel and its functions concepts
2. Gain knowledge to apply the appropriate Excel functions in the real data
3. Understand the functionality of the analysis tool pack to analyze the data
4. Learner get familiar with the analysis functions of statistics

VALUE ADDED COURSE

COURSE: 26UPCSCVA02

R Programming

[Credits: 4]

Course Objective:

This course should enable the students:

- o To be able to learn how to program in R and how to use R for effective data analysis.
- o Familiar with programming in R, reading data into R, accessing R packages, writing R functions, debugging, profiling R code, and organizing and commenting R code.

Implement the following concepts using R:

1. R data types and objects,
2. Reading and writing data
3. Control structures
4. Functions
5. Dates and times
6. Loop functions
7. Vector and Matrix Operations
8. List and Data-Frame Operations
9. Data Import and export Operations
10. Data Exploration and Visualization Operations

References

1. Larry Pace, "Beginning R: An Introduction to Statistical Programming, Apress, Second Edition, 2015
2. Richard Cotton, "Learning R: A Step-by-Step Function Guide to Data Analysis, O'Reilly, 2013

Course Outcomes:

The students should enable to:

1. Gain knowledge to identify appropriate R Programming language constructs to write basic programs for the given problem.
2. Identify and implement the right data representation formats and data structure based on the requirements of the problem.
3. Understand the functionality of R package and to implement the same for the problem solving.

Value Added Course

COURSE: 26UPCSCVA03

Bio Python

[Credits: 4]

Course Objective:

This course should enable the students:

- To be able to introduce programming basics and design using Python Programming Language
- To understand in-depth biological data and information processing techniques using Bio Python.

Implement the following concepts using Python:

1. Basics of Python
2. Advanced Data Structure
3. Control Flow
4. Functions and modules
5. Files
6. Retrieve Sequence from FASTA and Genebank files
7. Split a Sequence File into a set of smaller files.
8. Sequence file formats conversion
9. Manipulations of sequence files
10. Visualization and analyze of Sequence data.

References

1. Dusty Philips, “Python 3 Object Oriented Programming”, PACKT Publishing, 2nd Edition, 2015.
2. Sebastian Bassi, Python for Bioinformatics, Second Edition, Chapman & Hall and CRC Press, 2009
3. Jeff Chang, Brad Chapman, Iddo Friedberg, Thomas Hamelryck, Michiel de Hoon, Peter Cock, Tiago Antao, Eric Talevich, Bartek Wilczyński, Biopython Tutorial and Cookbook, 2019 (<http://biopython.org/DIST/docs/tutorial/Tutorial.pdf>)

Course Outcomes:

The students should enable to:

1. Gain knowledge to identify appropriate Python language constructs to write basic programs for the given problem.
2. Identify and implement the right data representation formats and data structure based on the requirements of the problem.
3. Understand the functionality of Bio Python package and to implement the same for the problem solving.

Value Added Course

COURSE: 26UPCSCVA04 Large Language Models (LLMs)

[Credits: 4]

Course Objectives

1. Understand the basic concepts and functions of Large Language Models (LLMs).
2. Apply LLMs (e.g., ChatGPT, Gemini) for summarization, analysis, and content generation.
3. Develop effective prompts to extract relevant information from LLMs.
4. Analyze sentiment, bias, and public opinion using LLM tools.
5. Demonstrate ethical considerations in using AI tools in research.

Lab Exercises

1. Introduction to LLMs & Applications
2. Working with ChatGPT / Gemini
3. Prompt Engineering Basics
4. Text Summarization & Paraphrasing
5. Sentiment & Opinion Analysis
6. Translation & Code-Switching
7. LLMs as Data Assistants
8. Generate Questionnaire
9. Visual Dashboard Plan
10. Ethical Use of AI in Social Sciences
11. Mini-Project & Presentation

Reference Books

1. Andriy Burkov – The Hundred-Page Language Models Book – Self-Published – 1st Edition – 2024
2. Jay Alammar & Maarten Grootendorst – Hands-On Large Language Models: Language Understanding and Generation – O'Reilly Media – 1st Edition – 2024
3. Sebastian Raschka – Build a Large Language Model (From Scratch) – Manning Publications – 1st Edition – 2024
4. Sinan Ozdemir – Quick Start Guide to Large Language Models – Addison-Wesley – 1st Edition – 2024

Course Outcomes (COs)

By the end of this course, learners will be able to:

CO No.	Course Outcome Description	Bloom's Level
CO1	Recall the fundamental concepts of LLMs and their evolution and its applications.	K1 - K6
CO2	Apply LLMs to summarize, paraphrase, and classify social content.	
CO3	Analyze texts for sentiment, bias, or opinion using prompt engineering.	
CO4	Evaluate the ethical use of AI tools in research and media.	
CO5	Design and present a mini-project applying LLMs to a social science issue.	

Value Added Course

COURSE: 26UPCSCVA05

[Credits: 4]

Basics of Computer Hardware & Introduction to Networking

Course Objectives:

The main objectives of this course are:

1. To understand core hardware components
2. To demonstrate Hardware Assembly and Troubleshooting
3. To grasp Networking Fundamentals
4. To set up and configure Network Devices
5. To apply safety and maintenance practices

Expected Course Outcomes:

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

CO1	Understand the functions and roles of essential computer hardware components such as the CPU, RAM, motherboard, storage devices, and input/output peripherals.	K1-K6
CO2	Develop hands-on skills in assembling a computer system, identifying hardware issues, and applying basic trouble shooting techniques.	
CO3	Understand basic networking concepts including types of networks, network topologies, IP addressing, and essential protocols like TCP/IP.	
CO4	Learn to configure basic networking devices such as routers, switches, and modems, and establish simple local area networks (LANs).	
CO5	Known for hardware handling, ESD safety, and routine maintenance to ensure system reliability and longevity.	

S.No	Module Name	Outcome	Theory (Hrs)	Practical (Hrs)	Total (Hrs)
1	Basic Computer Fundamentals and Basics of Power System in a Computer	Identify different parts of a computer system	01	01	02
2	Introduction to SMPS & UPS	Install Test and Troubleshoot Switch Mode Power Supply & UPS	01	01	02
3	Introduction to Basic Input Output System	Demonstrate configurations and changes in BIOS or CMOS Setup and POST	01	01	02
4	Introduction to Operating System	Install an OS in a new Computer System and configure the machine	01	02	03
5	Computer Management	Manage the resources of a Computer System according to users need	01	01	02

6	Disk Partitioning	Create partition of the HDD, SSD of a Computer System	01	01	02
7	Details about Central Processing Unit & Main board	Install a CPU on motherboard and under lying technology used	01	01	02
8	Primary and Secondary Memory	Install RAM, SSD, HDD and explain primary memory and secondary memory in a computer system	01	01	02
9	Computer Accessories	Install and connect accessories like optical drive, keyboard, mouse, monitor, Printer and troubleshoot	01	02	03
10	Introduction to Networking	Basic of Essential Network Components	01	01	02
11	Connecting Devices	Identify different types of Cables, RJ 45 Connectors, Punching/Crimping Tools and I/O Sockets	01	02	03
12	Network Communication Devices	Explain Working Principle of Communication and utility of Network	01	01	02
13	Knowledge of Sharing of Resources	Connect computers in a networking system and able to share, files, printers or other Resources in a system	01	01	02
14	Knowledge of Internet and Internet Network Configuration	Configure the modem and setup a secured connection via Internet.	01	01	01
Total (Hrs)					30

References Books:

1. Mike Meyers, "CompTIA A+ Certification All-in-One Exam Guide", 11th Edition, McGraw-Hill Education, 2021. (ISBN: 978-1260462189).
2. Mike Meyers, "Networking + Guide to Managing and Troubleshooting Networks", 7th Edition, McGraw-Hill Education, 2019. (ISBN: 978-1260458083).
3. Mike Meyers, "PC Hardware and Maintenance", 4th Edition, McGraw-Hill Education, 2016. (ISBN: 978-0071848232)
4. Charles Severance, "Introduction to Networking: How the Internet Works", 1st Edition, CreateSpace Independent Publishing Platform, 2014. (ISBN: 978-0993449200)
5. V. Rajaraman, "Hardware and Computer Organization", 5th Edition, Prentice Hall, 2018. 115 (ISBN: 978-9332573483)

Module No.1: Basic Computer Fundamentals & Basics of Power System in a Computer.

Outcome: Identify different parts of a computer system.

Theory Content:

- Computer Fundamentals History and Generations of Computer. (Gen1to5)
- Architecture of the Computer.
- Description of different parts of a Computer.
- Idea about Operating system, System Software and Application Software.

Practical Content:

- Identification of different Components of a computer and demonstration and uses of them.
- Different Tools / equipment used for assembling/disassembling a PC.
- Demonstrations of the power flow inside a computer through slides/PPT.
- Identification of the power source, coloured wires of interest etc.

ModuleNo.2: Introduction to SMPS & UPS.

Outcome: Install Test and Troubleshoot Switch Mode Power Supply & UPS.

- SMPS Features and Functions.
- Types of SMPS.
- Power distribution in SMPS.
- Components and Circuits inside the SMPS Unit.
- Types of Uninterrupted Power Supply (UPS) Offline and Online.

Practical Content:

- A running SMPS demonstration.
- Installation of a SMPS on a cabinet.
- Isolated Power testing by sorting.
- Live UPS Demonstration.

ModuleNo.3: Introduction to Basic Input Output System.

Outcome: Demonstrate configurations and changes in BIOS or CMOS Setup and POST.

Theory Content:

- Introduction to BIOS/CMOS Setup, POST. (Power on Self-Test)
- Demonstration of BIOS/CMOS Configuration. (Date, Time, Enable/Disable Devices)
- Dual BIOS Feature BIOS/CMOS Setup, Booting Sequence/Boot Order.

Practical Content:

- Demonstration of BIOS/CMOS setup, POST in a Computer step by step.
- Demonstration of different types of configurations and effect of changes in an existing BIOS feature for system.
- Implementation of Dual Booting setup preferably in windows.

ModuleNo.4: Introduction to Operating System.

Outcome: Install an OS in a new Computer System and configure the machine.

Theory Content:

- Definition and types of Operating Systems. Functions & Features of OS.
- MS-Dos (Prelims), Windows8.1, 10, 11, RedHat7 Linux Server, Ubuntu, MACOS, Android etc.
- Process of Booting the Operating System.
- Installation of Operating Systems such as Windows 8.1, 10, 11 Activation and Automatic Updating procedures.

Practical Content:

- Installation of different types of Operating Systems.
- Demonstration on Windows Using: Safe Mode, Safe Mode Boot options, Last Known Good Configuration, etc.
- Demonstrating Windows Diagnostic Tools, System Restore, Creating

Restore point.

- Windows Recovery using System Factory Defaults/Recovery tools.
- Installation of driver routines for the system.

ModuleNo.5: Computer Management.

Outcome: Manage the resources of a Computer System in Windows according to users need.

Theory Content:

- Computer Management, Disk Management, Defragmentation.
- Services and Applications, Local Users and Groups.
- Advanced System Settings, Device Manager, Task Manager, Windows Registry.

Practical Content:

- Demonstrating computer management in Windows using Disk manager, Shrink, Extend and Creating Logical Drive etc.
- Creating users, groups etc.
- Installing & uninstalling program, devices driver from control panel, uses of task manager, termination of a process etc.

ModuleNo.6: Disk Partitioning.

Outcome: Create partition of the HDD, SSD of a Computer System in Windows according to users need.

Theory Content:

- Partitioning of Hard Drive/SSD-Primary, Extended, Logical partitions using Partition Tools in Windows.

Practical Content:

- Demonstration of Partitioning of Hard Drive/ SSD - Primary, Extended and Logical partitions using Partition Tools in Windows through Disk Manager.

ModuleNo.7: Details about Central Processing Unit & Main board

Outcome: Install a CPU on mother board & underlying technology used.

Theory Content:

- Definition of CPU, Architecture, Speed, Types of CPU (XT and AT technologies) 32/64Bit.
- Types processor technologies like Clock speed, Bus speed, Cache memory.
- Threading technologies, Core technologies, IRQ & DMA Technologies.
- Brands (Intel/AMD), CPU Series—like Pentium, PI, PII, PIII, PIV etc.
- What is “I” Technology, Core series (i3, i5, i7).
- I/O Ports (PS/2, Serial, Parallel, USB, VGA, HDMI, Audio, Ethernet, etc.)

Practical Content:

- Demonstration of different generation CPU (Intel /AMD), showing clock speed, Bus Speed & Other features.
- Demonstration of technological difference between different generation’s CPU through Slides, Lecture, picture etc.
- Hand on practice to install CPU on the slot provided on the compatible main board.
- Installation of external cards viz. graphics, sound, Ethernet and etc., on the main board.

ModuleNo.8: Primary and Secondary Memory.

Outcome: Install RAM, SSD, HDD and explain Primary Memory and Secondary Memory in a Computer system.

Theory Content:

- Introduction to RAM, ROM, Cache Memory, Buffer Memory, Virtual Memory.

Speed, Timeline (EDO, NON-EDO, SD, RD, DDR, DDR2, DDR3, DDR4), Hybrid Memory.

- Comparing and Installing RAM, Memory Module Form Factor, etc.
- Introduction to HDD & SSD, Types, Functioning of HDD, Functioning of SSD, Dual Usage.
- Tracks, Sectors, Cylinders, Cluster MBR, FAT Area, DIR Area, Zero Track.
- IDE Jumper settings (Primary/Secondary), (Master/Slave/Cable Select).
- Installing and Configuring a New HDD/SSD.

Practical Content:

- Demonstration of different generation's RAM compatible with the system.
- Hand on practice to install RAM on the slot provided on the compatible main board.
- Demonstration of different types of HDD/SSD, showing speed, characteristics & other features.
- Hand on practice to install HDD/ SSD or both in a System.
- Hands on practice to make a HDD/SSD as Primary/Secondary Master/Slave using IDE Jumper Settings or cable select.
- Fresh installation a HDD/SSD, Partition, Installation of OS, Application S/W.

ModuleNo.9: Computer Accessories

Outcome: Install & connect accessories like optical drive, keyboard, mouse, monitor, Printer and troubleshoot

Theory Content:

Optical Drive

- Types of Optical Drives-ROM/Writer,
- Types of Media-(CD/DVD), Layers of CD and DVD, Difference between CD and DVD.
- Storage Technology-Reading, Writing &Re-Writing data, Drive Interfaces - (IDE/SATA).
- Burning CD/DVD's using multiple Burning Tools, i.e., Nero, NTI, etc.

- Creating, Managing and Burning ISO Images using tools like: Magic ISO, Power ISO etc.

Keyboard & Mouse

- Types of Keyboards.
 - Working Principles of Keyboard (Make/Break Scan Code, Key Matrix).
 - Keyboard Interfaces (DIN Type, PS/2, USB, wireless).
 - Types of Mouses (Ball Mouse, Optical Mouse, Trackball).
 - Principles of Mouse(X,Y Axis in Ball Mouse & DSP, Sensor in Optical Mouse).
- Mouse Interfaces(Serial, PS/2, USB, wireless)

Monitor

- Types of Monitors(CRT/LCD/LED) and Working Principles of each type.
- Demonstration and Practical on Monitor Installation, Menu Configuration, Adjusting Monitor Settings, Monitor Power Supply types, Possible Problems and Troubleshooting.

Printer

- Types of Printers (Dot Matrix, Inkjet, Laser, Thermal, All-in-One Printers, etc.)
- Interface Cables, Ports & Connectors.
- Working Principles of each type, network Printer.
 - Installing a Printer and Configuring Drivers.
 - Possible Printer Problems and Troubleshooting Techniques.
 - Types of Scanners (Handheld, Flatbed, Sheet fed, Portable Scanners). Interface Cables, Ports and Connectors.
 - Installation of Scanner, Device Driver Installation.
 - Scanner Settings, Scanning Documents, Photos in different Formats like JPG, PDF, etc.

Practical Content:

Optical Drive

- Hand on practice to learn on burning through requisite application Software.
- Connecting drives with the system with different interfaces.

- Hand on practice on Burning CD and DVD to create ISO image of software.

Keyboard & Mouse

- Demonstration of different types of Keyboards.
- Hands on practice with different types of mice.

Monitor

- Hands on practice with different types of monitor.
- Monitor Problems and Troubleshooting.

Printer

- Demonstration of different types of Printers (Dot Matrix, Inkjet, Laser, Thermal, All-in-One, network Printers etc.)
- Installing a Printer and Configuring Drivers.
- Printer Problems and Troubleshooting Techniques.
- Demonstration of different types of Scanners (Handheld, Flatbed, Sheet fed, Portable Scanners).
- Installing a Scanner and Configuring Drivers.
- Scanner Problems and Printer Troubleshooting Techniques.

ModuleNo.10: Introduction to Networking.

Outcome: Basic of Essential Network Components.

Theory Content:

- Essential Network Components for Data Transfer (Two Systems Required, Communication Media, Transmission Media, Connecting Devices and Protocol)

Practical Content:

- Demonstration of different types of Communication Media, Transmission Media and Connecting Devices.
- Hands on practice with Essential Network Components.