School of Science & Technology

PROGRAM STRUCTURE
ACADEMIC SESSION: 2025-2026
Batch: 2025-2029

Program Code: SSTUG001
(Program Name: B.Tech)
Specialization in AI & ML, Cyber Security, IOT and Data Science

JIGYASA UNIVERSITY

Formerly
Himgiri Zee University, Dehradun
(Estd. Under Uttaranchal State Act.No.17, 2003.Approved by UGC Under Sec.2(f))
Post Office Selaqui, Chakrata Road, Dehradun, Uttarakhand,248011

Program Structure
of
Bachelor of Technology
(B.Tech)
Specialization in AI & ML, Cyber Security, IOT and Data Science
(Program Name: B.Tech)
[Applicable w.e.f. Academic Session 2025-29]

JIGYASA UNIVERSITY

Formerly
Himgiri Zee University, Dehradun
(Estd. Under Uttaranchal State Act.No.17, 2003.Approved by UGC Under Sec.2(f))
Post Office Selaqui, Chakrata Road, Dehradun, Uttarakhand,248011

VISION OF UNIVERSITY

"To infuse transformations and foster indigenous values through excellence in teaching, research, and innovations, thereby enabling students to manifest their aspirations, and serve the society as a premier institution of higher education."

VISION OF SCHOOL OF SCIENCE & TECHNOLOGY (SST)

"Empowering Future Leaders"

"To nurture innovative, ethical, and globally competent technocrats and entrepreneurs through excellence in teaching, research, and industry collaboration, driving sustainable growth and societal impact."

MISSION STATEMENTS OF SST

M1. Research and Innovation: Promoting a culture of research and innovation by providing students with opportunities to engage in hands-on experimentation, independent research projects, and collaboration with industry partners.

M2. Critical Thinking and Problem-Solving: Fostering critical thinking skills and problem-solving abilities necessary to tackle complex challenges in science and technology.

M3. Preparation for Career Success: Equipping students with the skills, knowledge, and experiences needed to excel in their chosen careers or pursue further education in graduate school.

ABOUT THE PROGRAM

A. Introduction

The National Education Policy (NEP) 2020 envisions a progressive, inclusive, and multidisciplinary transformation of India's higher education system to make it more flexible, experiential, and globally competitive. It emphasizes innovation, critical thinking, creativity, and digital literacy as key pillars for holistic education. The policy also aligns with the objectives of Sustainable Development Goal 4 (SDG-4) – Quality Education, aiming to ensure equitable, inclusive, and lifelong learning opportunities for all learners.

In alignment with the vision of NEP 2020, the Bachelor of Technology (B.Tech) program in Computer Science and Engineering (CSE) offered by the School of Science & Technology, Himgiri Zee University, has been meticulously designed to produce globally competent engineers with a balanced blend of theoretical knowledge, practical expertise, and ethical awareness. The program integrates modern pedagogy, interdisciplinary learning, and hands-on exposure to emerging technologies such as Artificial Intelligence (AI), Machine Learning (ML), Data Science, Cyber Security, and the Internet of Things (IoT).

The program provides a dynamic and flexible learning environment, where foundational knowledge in computing merges with advanced competencies in software development, data analytics, and intelligent systems. The curriculum adopts the Choice Based Credit System (CBCS) and offers multiple entry and exit options as per NEP 2020 guidelines. It promotes academic flexibility through elective courses, MOOCs, internships, and project-based learning, enabling students to personalize their academic journey in line with their interests, professional aspirations, and evolving industry needs.

B.Tech is a four-year undergraduate degree program that typically covers the fundamental principles and applications of engineering and technology. The coursework for this degree is designed to prepare students for careers in various engineering fields, such as mechanical, electrical, or computer engineering:

Specialization available for:

Artificial Intelligence & Machine Learning	Cyber Security
Data Science	Internet of Things(IOT)

B. Curriculum Framework

1. Overview

The Bachelor of Technology (B.Tech) in Computer Science & Engineering (CSE) program is structured in accordance with the National Education Policy (NEP) 2020, AICTE Model Curriculum, and the Academic Regulations of the University. The curriculum adopts a Choice Based Credit System (CBCS) and Outcome-Based Education (OBE) framework to ensure flexibility, multidisciplinary exposure, academic rigor, and industry relevance. The program is designed to produce globally competent, ethically responsible, and industry-ready computer engineers with strong foundations in computing principles, engineering sciences, mathematics, and emerging technologies such as Data Science, Artificial Intelligence, Internet of Things, Cloud Computing, Cyber Security, and Big Data Analytics. The curriculum provides multiple entry and exit options, supports credit transfer, and allows up to 20% of courses through MOOCs (SWAYAM or international platforms) as per NEP 2020 guidelines. A Bridge Course on Foundations of Engineering Mathematics is introduced during the orientation program to address learner diversity and ensure uniform preparedness.

The courses are broadly classified as follows:

Basic Science, Engineering Science, Humanities & Social Sciences, Audit Courses (Soft Skills) ,(NC),Program Core Courses ,Elective Courses, Open Elective Courses, Program Specialization (Data Science), Minor/Major Projects & MOOC

1. **Basic Science Courses (BS):** Basic Science Courses establish a strong theoretical foundation in Physics, Chemistry, and Mathematics, which are essential for understanding computational models, algorithmic thinking, data science, artificial intelligence, and advanced engineering applications. These courses enhance logical reasoning, analytical skills, and quantitative problem-solving abilities, enabling students to effectively grasp higher-level engineering and computing concepts.

2. **Engineering Science Courses (ES / ESC):** Engineering Science Courses serve as a bridge between basic sciences and core engineering disciplines. They introduce students to fundamental engineering principles, programming for problem solving, design thinking, electronics, workshop practices, and engineering graphics. These courses emphasize hands-on learning, design orientation, and engineering problem-solving, fostering creativity, innovation, and technical confidence at an early stage of the program.

3. **Humanities and Social Science Courses (HSC):** Humanities and Social Science Courses focus on the holistic development of students by enhancing communication skills, ethical awareness, human values, and societal understanding. Courses in English, Universal Human Values, emerging technologies, and related areas help students develop professional communication, ethical reasoning, teamwork, and leadership qualities, preparing them to function effectively in diverse and global work environments.

4. **Program Core Courses (PC):** Program Core Courses constitute the academic backbone of the B.Tech CSE program. These courses cover essential domains of computer science and engineering, including data structures and algorithms, programming languages, database management systems, operating systems, computer organization and architecture, software engineering, computer networks, artificial intelligence, and web technologies. The core curriculum ensures that students

acquire strong theoretical knowledge combined with practical laboratory experience, enabling them to design, develop, and analyze efficient computing systems.

5. **Elective Courses (ELE):** Elective Courses provide academic flexibility and allow students to explore advanced and emerging areas of computing based on their interests and career aspirations. These courses introduce contemporary technologies such as Machine Learning, Big Data Analytics, Blockchain Technology, Cloud Computing, Data Analytics and Visualization, Internet of Things, and Digital Marketing. Electives encourage specialization, innovation, and industry relevance, supporting diverse career pathways in industry, research, and entrepreneurship.

6. **Open Elective Courses (OELE):** Open Electives promote multidisciplinary learning by allowing students to choose courses from other engineering disciplines or allied domains. This component aligns with the NEP 2020 vision of broad-based education, enabling students to develop cross-disciplinary perspectives, entrepreneurial thinking, and problem-solving skills applicable across multiple sectors.

7. **Program Specialization Courses (PS):** Program Specialization Courses offer in-depth and advanced learning in a selected specialization such as **Data Science**. These courses focus on advanced computing paradigms, intelligent systems, data-driven decision-making, and application-oriented learning. The specialization component strengthens industry readiness and research orientation by aligning academic learning with emerging technological trends and market demands.

8. **Projects, Internship, and MOOCs (MP):** This component emphasizes experiential and project-based learning through summer training, minor projects, major projects, internships, and MOOCs offered via platforms such as SWAYAM and international institutions. These activities enable students to apply theoretical knowledge to real-world problems, develop teamwork and project management skills, and gain exposure to industry and research practices. As per NEP 2020, up to 20% of the credits may be earned through approved online courses.

9. **Audit Courses / Skill Enhancement Courses (AC):** Audit and Skill Enhancement Courses focus on developing soft skills, professional communication, ethical conduct, leadership qualities, and employability skills. Though non-credit in nature, these courses play a crucial role in personality development, career readiness, and lifelong learning.

10. **Skill Enhancement Courses (SEC):** The Skill Enhancement Courses (SEC) focus on developing industry-oriented technical and analytical skills essential for employability and entrepreneurship. Typical SEC courses include Python Programming for Data Science, Data Visualization and Business Intelligence, AI Frameworks and Tools (TensorFlow, PyTorch, Keras), Cloud and DevOps Tools for AI Deployment, Version Control and Collaborative Development (Git, Docker). Each course integrates hands-on training, workshops, and practical sessions, ensuring that students gain **job-ready technical competencies** applicable in the AI and IT industries.

11. **Ability Enhancement Courses (AEC):** The **Ability Enhancement Courses (AEC)** aim to strengthen essential cognitive and soft skills required for professional growth. Key AECs include such as Technical Communication and Professional Writing, Quantitative Aptitude and Logical Reasoning,

Innovation and Design Thinking, Leadership and Team Dynamics, Analytical and Problem-Solving Skills. These courses enhance students' critical thinking, communication, and decision-making capabilities, aligning with NEP-2020's emphasis on **holistic and employability-focused education.**

12. **Value-Added Courses (VAC):** The **Value-Added Courses (VAC)** provide holistic enrichment, integrating ethics, sustainability, wellness, and cultural awareness with technical learning. Examples include such as Ethical and Responsible AI, Cybersecurity Awareness and Digital Ethics, Environmental Science and Sustainable Technologies, Yoga and Mindfulness for Engineers, Indian Knowledge Systems in Technology. These courses instill **values of integrity, environmental consciousness, and social responsibility**, supporting the UN Sustainable Development Goals (SDG-4: Quality Education).

13. **Capstone Project (CAP):** The **Capstone Project (CAP)** serves as a **culminating experiential component** of the program, bridging classroom knowledge with real-world applications. Students undertake team-based or individual projects focusing on Machine Learning Applications, Predictive Analytics, AI-Based Automation Systems, Data-Driven Decision Support Tools. Through the CAP, students demonstrate technical proficiency, creativity, and project management skills. The project is typically completed during **Semester VII**, with evaluation and presentation in **Semester VIII**, as per AICTE and institutional guidelines.

14. Research/Major Project: In line with NEP-2020's vision for **integrating research and innovation at the undergraduate level,** the **Research Project (REP)** is introduced in the **final semester (VIII).** Students undertake an **independent or supervised research investigation** in an AIML domain, leading to a **dissertation or research publication**. Objectives include like Promoting inquiry-based learning and critical analysis, Developing research and academic writing skills, Encouraging publication and participation in conferences. The REP fosters innovation, deep learning, and readiness **for higher education, research careers, or startup ventures.**

C. Program Educational Objectives:

The program educational objectives of JU Computer Science Engineering Department can be categorized under the following FIVE titles

1. To equip the students in becoming successful in Computer Science Engineering careers like Engineering Design, Consultancy and Research by broadening their origin in the field of Computer Science & Engineering.
2. To develop skill in identifying the needs of society and formulating the needs into problems that can be solved by basic knowledge of Computer Science Engineering, Mathematics and Sciences.
3. To provide opportunities for acquiring in-depth knowledge of fundamental concepts and programming skills for integrated development and provide solutions to challenging problems in their profession by applying computer engineering theory and practices.
4. To exhibit leadership quality with good communication skills with which the student will be able to work in diversified areas and work in large themes.

5. To train the student to develop the ability to design and conduct experiments, analyze, collect and interpret the data, and carry out scientific research.

D. PO's - Program Outcomes (B.Tech)

PO	Outcome
PO1	Engineering knowledge: Apply the knowledge-using fundamental of STEM i.e. Science, Technology, Engineering and Mathematics, to solve the complex engineering problems.
PO2	Problem investigation: Identify, formulate, review research articles, and analyze complex engineering problems using mathematics, sciences, and fundamental engineering.
PO3	Design and development of explanation: Design solutions for complex engineering difficulties give the solution through appropriate consideration safety, and the societal and environmental reflections.
PO4	Investigations of difficult problems: Analyze research-based skills and to provide valid conclusions.
PO5	Application of advanced tools: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools to solve complex engineering problems with an understanding of the limitations.
PO6	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge for sustainable development.
PO7	Individual and Teamwork: An ability to use work effectively as an individual, and as a leader.
PO8	Communication: Explain effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO9	Project management and finance: Assess the knowledge and understanding of the engineering and management principles and apply these to one's own work to manage projects.
PO10	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.
PO11	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and multidisciplinary environments.
PO12	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

E. Program Specific Outcomes (PSOs)

At the end of the **B.Tech in Computer Science and Engineering** program, students will be able to:

PSO	Outcome
PSO1	Apply the fundamental principles of computer science, mathematics, and engineering to identify, analyze, and design efficient algorithms and computing solutions for real-world problems.
PSO2	Design, develop, test, and maintain software systems using appropriate programming languages, development frameworks, and modern engineering tools to meet specified user requirements.
PSO3	Demonstrate the ability to apply modern technologies such as Artificial Intelligence, Data Science, Internet of Things (IoT), Cloud Computing, and Cybersecurity in developing innovative and sustainable computing solutions.
PSO4	Exhibit professionalism, teamwork, effective communication, ethical responsibility, and the ability to engage in lifelong learning to adapt to evolving technological environments.

F. Mapping of PEOs & Pos-PSOs in B.Tech

Matrix

PEO ↓ / PO →	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3	PSO 4
PEO 1	3	3	3	2	3	1	2	1	2	1	2	2	3	3	2	2
PEO 2	3	3	3	3	2	2	2	1	2	1	2	2	3	3	3	2
PEO 3	2	2	3	3	3	2	2	2	3	2	3	3	2	3	3	3
PEO 4	2	1	2	2	2	3	3	3	3	3	3	3	2	2	2	3
PEO 5	2	2	2	3	2	2	2	3	2	3	2	3	2	2	3	3

Note: *In alignment with Outcome-Based Education (OBE) principles, the mapping of Program Educational Objectives (PEOs) to Program Outcomes (POs) is presented using a* ***quantitative scale (1–3)****, where:*

- ***1*** *indicates a low level of contribution*
- ***2*** *indicates a moderate level of contribution*
- ***3*** *indicates a high level of contribution*

This mapping ensures that each PEO is ***progressively achieved through the attainment of relevant POs****, thereby providing a structured and measurable approach to evaluating the effectiveness of the program in delivering its long-term educational goals.*

G. Pedagogy, Teaching and Learning

The **pedagogical framework** of the B.Tech. (Computer Science and Engineering) program is designed to foster **active learning, critical thinking, innovation, and industry readiness** through a blend of **traditional and contemporary teaching approaches.** The focus is on achieving learning outcomes aligned with program objectives and enhancing **both technical and professional competencies** of students.

I. Teaching–Learning Methodologies:

To ensure comprehensive understanding and skill development, a variety of teaching methods are employed:

a. Lectures and Tutorials: Conceptual knowledge is delivered through structured lectures supported by multimedia presentations, animations, and case studies. Tutorials reinforce theoretical concepts and promote problem-solving skills.

b. Laboratory-Based Learning: Each core and elective course includes practical sessions to help students gain hands-on experience with modern software tools, programming environments, and simulation platforms.

c. Project-Based Learning (PBL): Students apply their learning to real-world problems through mini-projects, capstone projects, and interdisciplinary projects, fostering innovation and teamwork.

d. Experiential and Industry-Integrated Learning: Industry visits, expert lectures, and internships bridge the gap between academic learning and industrial practices. Collaboration with industries through MoUs and live projects provides students with practical exposure to emerging technologies.

e. ICT-Enabled Learning: Use of Learning Management Systems (LMS), virtual labs, and online resources (MOOCs, NPTEL, SWAYAM, Coursera) to supplement classroom instruction. Integration of digital platforms for flipped classrooms, online discussions, and assessments.

f. Collaborative and Peer Learning: Group assignments, seminars, workshops, and hackathons encourage peer interaction, cooperative learning, and leadership development.

g. Research-Based Learning: Students are encouraged to participate in research projects, publications, and innovation activities under faculty mentorship, promoting lifelong learning and creativity.

II. Student-Centric Learning Approach

The teaching approach is **learner-centric**, emphasizing:

a) Inquiry and curiosity-driven learning.

b) Personalized mentoring and remedial sessions for academic support.

c) Encouragement for self-paced and independent learning through MOOCs and online platforms.

III. Assessment and Continuous Evaluation

The evaluation process ensures consistent monitoring of student progress through:

a) **Formative Assessment:** Class tests, quizzes, assignments, viva-voce, and laboratory performance.

b) **Summative Assessment:** Mid-semester and end-semester examinations.

c) **Continuous Assessment:** Evaluation of projects, presentations, attendance, and participation.

d) **Emphasis** on **outcome-based assessment** to ensure attainment of **Program Outcomes (POs)** and **Course Outcomes (COs).**

IV. Outcome-Based Education (OBE) Integration

The pedagogy aligns with the **Outcome-Based Education (OBE)** framework prescribed by **AICTE** and **NBA**, ensuring:

a) Clear mapping between COs, POs, and PSOs.

b) Use of rubrics and performance indicators for continuous quality improvement.

c) Regular curriculum review based on feedback from stakeholders (students, alumni, industry, and faculty).

V. Blended Learning and Innovation :

a) Incorporation of **AI tools, virtual simulations, and gamified learning** to enhance engagement.

b) Use of **innovation labs and centers of excellence** to nurture problem-solving and entrepreneurial mindsets.

c) Promotion of **Design Thinking** and **Hackathon culture** to stimulate creativity.

H. Question Paper and Assessment Tools

The **assessment and examination system** for the *B.Tech. in Computer Science and Engineering (CSE)* program at *Himgiri Zee University* is designed in accordance with the **Outcome-Based Education**

(OBE) framework and the principles of the **National Education Policy (NEP) 2020**. The objective is to evaluate not only the theoretical understanding of students but also their analytical skills, creativity, problem-solving ability, and application of knowledge to real-world contexts.

I. Assessment Philosophy

The evaluation process focuses on **continuous and comprehensive assessment** to ensure that each student achieves the desired *Course Outcomes (COs)*, *Program Outcomes (POs)*, and *Program Educational Objectives (PEOs)*. Assessment is both **formative** (to guide and improve learning) and **summative** (to measure learning outcomes at the end of a course).

The system ensures:

a) Transparency, fairness, and objectivity in evaluation.

b) Integration of academic performance with professional and ethical development.

c) Continuous monitoring of student progress through diversified tools.

II. Components of Assessment

Component	Nature	Weightage	Evaluation Mode / Tool	Purpose
Continuous Internal Evaluation (CIE)	Formative	40%	Class tests, quizzes, tutorials, lab work, mini-projects, assignments, case studies, presentations, viva-voce	To assess regular learning, conceptual clarity, and application skills
End Semester Examination (ESE)	Summative	60%	Written examination conducted at semester end	To assess comprehensive understanding, synthesis, and higher-order thinking
Laboratory / Practical Evaluation	Continuous	As per course	Lab performance, records, viva, project demonstrations	To test technical proficiency and practical problem-solving
Project / Internship / MOOC Evaluation	Continuous + Summative	Variable	Project report, viva, presentation, innovation index	To evaluate integration of knowledge and experiential learning outcomes

III. Structure of Question Paper

Each theory course is evaluated through a **3-hour End-Semester Examination** carrying **60 marks**, structured as follows (may vary slightly across subjects as per faculty board approval):

Part A – Objective Type Questions (1x5= 5 marks): 5 objective type questions covering all units of the syllabus, testing comprehension, definitions, basic principles, and recall.

Part B – Short Answer Questions (2x5= 10 marks): 5 short answer questions out of 6, evaluating understanding and explanation of concepts.

Part C – Descriptive/Analytical Questions (3x5=15 marks): 5 descriptive/analytical questions out of 7, assessing application, explanation, and synthesis of concepts.

Part D – Problem Solving/Design-Based Questions (10x3=30 marks): 3 problem-solving/design-based questions out of 5, evaluating problem-solving, creativity, and engineering judgment.

Note: The question paper follows a balanced approach, covering all Course Outcomes (COs) and mapping COs to POs (Program Outcomes).Note:

- Questions shall be framed covering all *Course Outcomes (COs)*.
- Mapping of COs → POs shall be indicated in the course file.
- Question papers shall follow Bloom's Taxonomy levels to ensure balanced cognitive assessment.

IV. Continuous Internal Evaluation (CIE) Tools

The **CIE** component (40%) is distributed among multiple tools to ensure comprehensive evaluation of student performance throughout the semester:

Tool / Method	Description	Typical Weightage
Class Tests / Mid-Term Exams	Two written tests per semester covering defined syllabus units	15%
Assignments / Tutorials	Analytical or design-based exercises, coding tasks, or mathematical problem solving	10%
Quizzes / Oral Tests	Short assessments, online quizzes, or viva to promote continuous learning	5%
Presentations / Seminars	Individual or group presentations on recent trends or case studies	5%
Attendance / Participation	Active involvement in class, labs, and discussions	5%

For laboratory courses, evaluation shall be continuous and comprehensive, and shall be based on the following components, in line with NAAC, AICTE, and Outcome-Based Education (OBE) guidelines:

1. **Regularity and Attendance in Laboratory Sessions:** Assessment of student punctuality, preparedness, and active participation during laboratory hours.

2. **Laboratory Performance and Practical Skills:** Evaluation of the student's ability to understand experiments, execute programs/experiments correctly, use tools and equipment efficiently, and troubleshoot errors independently.
3. **Laboratory Record / Practical File:** Assessment of the completeness, correctness, and timely submission of laboratory records, including experiment objectives, algorithms, code, outputs, observations, and results.
4. **Continuous Internal Assessment (CIA)**
   Periodic evaluation through:
   - Lab quizzes
   - Viva-voce
   - Short practical tests
   - Demonstration of experiments/program
5. **Mini Projects / Assignments (where applicable) :** Evaluation of problem identification, solution design, coding practices, documentation, and innovation in mini-projects or case-based lab assignments.
6. **End-Semester Practical Examination (ESE):** Formal practical examination assessing(10 Marks):
   - Conceptual understanding
   - Practical implementation
   - Debugging skills
   - Viva-voce and result interpretation
7. **Professional Conduct and Safety Practices :** Assessment of adherence to laboratory rules, ethical computing practices, data integrity, and safety norms.

- Regular performance and punctuality (10%)
- Experiment record and analysis (20%)
- Practical skill and innovation (30%)
- Viva / practical test (40%)

V. Project, Internship, and MOOC Evaluation

Component	Mode of Assessment	Evaluation Criteria
Mini Project (2nd–6th Sem.)	Departmental committee evaluation	Problem definition, implementation, innovation, teamwork
Major Project (7th–8th Sem.)	Internal + External Examiner evaluation	Research depth, design quality, innovation, documentation, societal relevance
Internship / Industrial Training	Evaluation through reports, presentations, and industry feedback	Practical exposure, application of learning, professional conduct
MOOCs (SWAYAM/NPTEL/Coursera)	University-approved certificate submission	Credit equivalence based on course hours and grades obtained

VI. Mapping of Assessment to Bloom's Taxonomy

Cognitive Level	Assessment Tool(s)
Remember / Understand	Short answer questions, quizzes, assignments
Apply	Numerical / programming questions, lab tasks
Analyze	Case studies, mid-term exams, design problems
Evaluate	Major project reviews, viva-voce
Create	Capstone projects, research papers, innovative prototypes

VII. Result Analysis and Feedback

a) Post-examination, a detailed **Course Outcome (CO) attainment report** is prepared.

b) The **Result Analysis Committee** reviews student performance for each course.

c) Faculty analyze low-attainment areas and recommend corrective pedagogical measures.

d) Continuous improvement is ensured through **CQI (Continuous Quality Improvement)** practices in line with NBA and AICTE standards.

VIII. Key Features of the Assessment Framework

a) Balanced evaluation of knowledge, skills, and attitude.

b) Transparent, reliable, and verifiable grading mechanisms.

c) Encouragement of innovation and research aptitude.

d) Alignment with NEP-2020 emphasis on *competency-based, holistic assessment.*

e) Support for lifelong learning and professional excellence.

PO's - Program Outcomes (B.Tech)

PO	Outcome
PO1	Engineering knowledge: Apply the knowledge-using fundamental of STEM i.e. Science, Technology, Engineering and Mathematics, to solve the complex engineering problems.
PO2	Problem investigation: Identify, formulate, review research articles, and analyze complex engineering problems using mathematics, sciences, and fundamental engineering.
PO3	Design and development of explanation: Design solutions for complex engineering difficulties give the solution through appropriate consideration safety, and the societal and environmental reflections.
PO4	Investigations of difficult problems: Analyze research-based skills and to provide valid conclusions.
PO5	Application of advanced tools: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools to solve complex engineering problems with an understanding of the limitations.
PO6	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge for sustainable development.
PO7	Individual and Teamwork: An ability to use work effectively as an individual, and as a leader.
PO8	Communication: Explain effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO9	Project management and finance: Assess the knowledge and understanding of the engineering and management principles and apply these to one's own work to manage projects.
PO10	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.
PO11	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and multidisciplinary environments.
PO12	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Mapping of PEOs & POs in B.Tech

Matrix

PEO ↓ / PO →	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
PEO 1	3	3	3	2	3	1	2	1	2	1	2	2	3	3	2	2
PEO 2	3	3	3	3	2	2	2	1	2	1	2	2	3	3	3	2
PEO 3	2	2	3	3	3	2	2	2	3	2	3	3	2	3	3	3
PEO 4	2	1	2	2	2	3	3	3	3	3	3	3	2	2	2	3
PEO 5	2	2	2	3	2	2	2	3	2	3	2	3	2	2	3	3

STRUCTURE OF UG PROGRAM IN B.TECH

Matrix

Category Name	Course Category Code	No. of Courses	Credit Calculation	Total Credits
Basic Science	**BS**	5	(4×4) + (1×1) = 21	17
Engineering Science	**ES / ESC**	4	(4x1)+(3x1)+(2x1)+(1x1)=10	10
Humanities & Social Science	**HSC**	8	(2×7) + (1×1) = 14	15
Audit Courses (Soft Skills / MOOCs)	**AC / AU**	6	(2x5)+(1x1)=11	11
Core Courses	**PC**	22	(4×14) + (3×8) = 80	80
Elective Courses (Department / Domain)	**ELE**	8	(3×6)+(4x2) = 26	26
Capstone Project	**MP**	2	(1×2) + (1x4)= 6	6
Dissertation	Dissertation	3	(1x3)	3
Total	—	**58**	—	**168**

*Minor variation is allowed as per need of the respective disciplines.

Matrix

S. No.	Category	Credit Breakup for CSE students
1	Humanities and Social Sciences including Management courses	15

2	Basic Science courses	17
3	Engineering Science courses including workshop, drawing, basics of electrical/mechanical/computer etc.	10
4	Professional core courses	80
5	Professional Elective courses relevant to chosen specialization/branch	26
6	Project work, seminar and internship in industry or elsewhere	9
7	Mandatory Courses [Environmental Sciences, Induction Program, Indian Constitution, Essence of Indian Knowledge Tradition]/Audit Course	11
	Total	168

*Minor variation is allowed as per need of the respective disciplines.

Program Matrix

Bachelor of Technology (B.Tech)

S. No.	Course Code	Course Name	Category	Numbers of Hours/Week			C	Marks				Total Marks
			Core / Elective / Others	L	T	P		Theory		Practical		
								I	E	I	E	
SEMESTER I												
1	BSC 101	Physics I	Basic Science	3	0	2	4	50	50	25	25	150
2	ESC 102	Basic Electrical Engineering	Engineering Science	3	0	2	4	50	50	25	25	150
3	BSC 102	Mathematics-I	Basic Science	3	1	0	4	50	50	-	-	100
4	ESC 104	Design Thinking	Engineering Science	1	0	2	2	50	50	25	25	150
5	ESC 105	Engineering Graphics & Design	Engineering Science	1	0	4	3	50	50	25	25	150
6	CSE C106	Introduction to the Computer Programming using C	Core Course	3	0	2	4	50	50	25	25	150
7	JIU C 101	Industrial Communication and Workplace Ethics	Humanities & Social Science	2	0	0	2	25	25	-	-	50
Total				16	1	12	23					
SEMESTER II												
1	BSC111	Chemistry I	Basic Science	3	0	2	4	50	50	25	25	150
2	BSC112	Mathematics II	Basic Science	3	1	0	4					
3	CSC114	Data Structure using C	Core Course	3	0	2	4	50	50	25	25	150
4	ESC115	Workshop/Manufacturing Practices	Engineering Science	0	0	2	1	-	-	25	25	50
5	CSC116	Python Programming	Core Course	3	0	2	4	50	50	25	25	150
6	BSC117	Environmental Science	Basic Science	1	0	0	1	25	25	-	-	50
7	JIUC102	Verbal ability and Logical Reasoning	Humanities & Social Science	2	0	0	2	25	25	-	-	50
8	JIUA 102	Community Connect Program II	Humanities & Social Science	0	0	2	1			25	25	50
10	CSI 108	Internship-I	3-4 Weeks internship to be completed at the end of first or second semester during vacation period and its evaluation/ credit to be added in third semester									
Total				15	1	10	21					
Cumulative Total				31	2	22	44					
SEMESTER III												
1	CSE201	Design & Analysis of Algorithms	Core Course	3	0	0	3	50	50	25	25	150

2	CSE202	Object Oriented Programming Using Java	Core Course	3	0	2	4	50	50	25	25	150
3	CSEC211	Relational Database Management System	Core Course	3	0	2	4	50	50	25	25	150
4	BSC204	CBNST [Computer Based Numerical and Statistical Techniques]	Core Course	3	1	0	4	50	50	-	-	100
5	ELE-I	Elective	Elective	3	0	0	3	50	50	-	-	100
6	ASC206	Essence of Indian Knowledge Tradition	Audit Course	1	0	0	1	25	25	-	-	50
7	JIUS 101	Quantitative and Qualitative Aptitude for Engineers	Humanities & Social Science	2	0	0	2	25	25			50
9	VAMC 001	Value Added Course	Audit Course	2	0	0	2					
10	CSI 208	Evaluation of Internship-I completed at I year level /Seminar for Lateral Entry students	Dissertation	-	-	2	1	-	-	50		50
Total				20	1	6	24					
	Cumulative Total			51	3	28	68					
SEMESTER IV												
1	CSEC204	Theory of Computation	Core Course	3	0	0	3	50	50	-	-	100
2	CSEC212	Operating System	Core Course	2	0	2	3	50	50	25	25	150
3	CSEC213	Computer Organization and Architecture	Core Course	3	0	0	3	50	50	-	-	100
4	CSE218	Discrete Structures	Core Course	3	1	0	4	50	50	-	-	100
5	ELE-II	Elective	Elective	3	0	2	4	50	50	25	25	150
7	JIUS 101	Leadership and Team Dynamics for Engineer	Humanities & Social Science	2	0	0	2	50	50	-	-	100
8	VAMC 002	Value Added Course	Audit Course	2	0	0	2					
9	CSI 209	4-6 Weeks Internship based on using various software's –Internship -II	To be completed anytime during the Third/ fourth semester. Its evaluation/credit to be added in fifth semester									
Total				18	1	4	21					
Cumulative Total				69	4	32	89					
SEMESTER V												
1	CSEC301	Data Communication and Computer Networks	Core Course	2	1	0	3	50	50	-	-	100
2	CSEC302	Compiler Design	Core Course	3	0	2	4	50	50	25	25	150
3	CSEC2	Software Engineering	Core Course	3	0	0	3	50	50	-	-	100

	14											
4	HSC304	Human Values	Humanities & Social Science	2	0	0	2	25	25			50
5	ELE-III	Elective	Elective	3	0	2	4	50	50	25	25	150
6	ELE-IV	Elective	Elective	3	0	0	3	50	50	25	25	150
8	JIUE 001	CCNA Routing and Switching :Introduction to Network	Humanities & Social Science	2	0	0	2	25	25			50
9	VAMC 003	Value Added Course	Audit Course	2	0	0	2					
10	CSI 308	Evaluation of Internship-II completed at II year level	Dissertation	-	-	2	1	-	-	50		50
Total				20	1	6	24					
Cumulative Total				89	6	38	113					
SEMESTER VI												
1	CSEC316	Web Technology with PHP	Core Course	3	0	2	4	50	50	25	25	150
2	CSEC317	Artificial Intelligence	Core Course	3	0	2	4	50	50	-	-	100
3	ELE-V	Elective	Elective	3	0	0	3	50	50	-	-	100
4	ELE-VI	Elective	Elective	3	0	0	3	50	50	-	-	100
5	CSE 320	Wireless Technology	Elective	2	1	0	3	50	50	-	-	100
7	ASC322	Research and Publication Ethics	Audit Course	2	0	0	2	50	50	-	-	100
8	JIUE002	Data Storytelling and Visualization	Humanities & Social Science	2	0	0	2	50	50	-	-	100
9	VAMC 003	Value Added Course	Audit Course	2	0	0	2					
10	CSI 309	4-6 Weeks Internship based on using various software's –Internship -III	To be completed anytime during the Third/ fourth semester. Its evaluation/credit to be added in Seventh semester									
Total				20	1	4	23					
Cumulative Total				109	7	42	136					
SEMESTER VII												
1	CSE 401	Advance Web Technology	Core Course	3	0	2	4	50	50	25	25	150
2	CSE 403	Advance Computer Architecture	Core Course	3	0	0	3	50	50	-	-	100
3	CSE 404	Decision Support System	Core Course	3	0	0	3	50	50	-	-	100
4	CSE 405	Real Time System	Core Course	3	0	0	4	50	50	-	-	100
5	CSE	Big Data Analytics	Core Course	3	0	0	4	50	5	-	-	100

	408								0			
6	CSI 408	Evaluation of Internship-III completed at III year level	Dissertation	-	-	2	1			50		50
7	JIUI 001/JIUD001/JIUA001/JIUC001	Minor Project-II	Capstone Project			4	2			50	50	100
Total				15	7	8	21					
Cumulative Total				124	7	50	157					
SEMESTER VIII												
	CSE 407	.Net Technology & VB .Net	Core Course	3	0	2	4	50	50	25	25	150
	ELE-VII	Elective I	Elective	3	0	0	3	50	50	-	-	100
1	JIUI 002/JIUD002/JIUA002/JIUC002	Major Project/Industrial Report	Capstone Project	-	-	8	4			100	100	250
Total				6	0	10	11					
Grand Total				**130**	**7**	**60**	**167**					
L – Lecture 1L = 1Hr Paper per week		T- Tutorial 1T= 1 Hr	P- Practical 1P=1 Hr	C-Credits 1C = 1 Hr of Theory = 2 Hrs of Practical/Tutorial per week								

- NC means No Credit
- Note: 20% of subjects can be allowed to be taken online through SWAYAM or any other international Institute.
- Bridge course(Foundation of engineering mathematics) is introduced in orientation program

Elective Courses (ELE) : Grouped on the basis of Specializations in B.Tech

Artificial Intelligence & Machine Learning			
CSEA205	AI & ML Foundations	CSEA219	Fuzzy Logic & Neural Network
CSEA305	Machine Learning using python	CSEA319	Deep Learning using R
CSEA404	Computer Vision theory		
Cyber Security			
CSEC205	Information Security Fundamentals	CSEC219	Cryptography and Network Security
CSEC305	Security and Privacy in cloud	CSEC319	Blockchain Technology
CSEC404	Ethical Hacking		
Data Science			
CSED205	Introduction to Data Science	CSED219	Python for Data Science
CSED305	Statistical Machine Learning & Modeling	CSED319	Data Analytics and Visualization
CSED404	Big Data Acquisition and Analysis		
Internet of Things(IoT)			
CSEI205	IOT Architectures & Protocols	CSEI219	Embedded System & Internet of Things
CSEI305	Software Development in IOT	CSEI319	Machine Learning for Internet of Things
CSEI404	Internet of Things Security		

Skill Enhancement Courses (SEC) : List

JIU C 101	Industrial Communication and Workplace Ethics	JIU E 001	CCNA Routing and Switching : Introduction to Networks
JIU C102	Verbal Ability and Logical Reasoning	JIU E 002	Data Storytelling and Visualization
JIUS 001	Quantitative and Qualitative Aptitude for Engineers	JIUS 002	Leadership and Team Dynamics for Engineers

Value Added Mandatory Courses (VAMC) / MOOCS / Audit Course: List

VACS 001	Introduction to Power BI	VACS 002	Cyber Law & Ethics
VACS 003	Simulation Tools	VACS 004	Tools for algorithm Analysis

CAPSTONE PROJECT (CAP): List

PCS 405	Minor Project-II
PCS 501	Major Project/Industrial Report

Semester: 1st								
Course Name: Physics I	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC 101	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To introduce core concepts of fibre optics, holography, lasers, electrostatics, magneto statics, and superconductivity.
2. To understand the structure, principles, and applications of optical fibres.
3. To explain the basics and uses of holography and lasers.
4. To strengthen the fundamentals of electrostatics and magnetostatics.
5. To outline the properties and applications of superconductors.

Course Outcomes: After completing this course the student will be able to

CO1: State the basics of optical fibre and its real life applications.
CO2: Illustrate the phenomenon of Holography technology.
CO3: Articulate the basic concept of Coherent sources and its application in LASER.
CO4: Assess the knowledge about Electrostatics and Magnetostatics.
CO5. Function of superconductivity.

Detailed Syllabus:

Unit I **8 Hours**

Fibre Optics: Introduction, advantages, principle of total internal reflection; Structure of optical fibre –core, cladding, coating, refractive index profile; Light guidance – ray optics, wave optics , modes of propagation; Acceptance angle and cone – definition, derivation, measurement; Numerical aperture –relation with acceptance angle, light-gathering ability; Types of optical fibres – single-mode, multimode, step-index, graded-index; Attenuation – absorption, scattering, bending losses; Dispersion – material, waveguide, modal; Applications – communication, medical, sensing, defence, emerging uses.

Unit II **8 Hours**

Holography: Basic principle of holography – interference and diffraction; Recording of holograms – setup, coherent light source, reference and object beams; Reconstruction process – illumination, image formation; Types of holograms – transmission, reflection, phase, volume; Applications – data storage, security, microscopy, 3D imaging, industrial measurements.

Unit III **8 Hours**

Laser: Coherent sources and their characteristics; Interaction of radiation with matter – spontaneous emission, stimulated emission, and absorption; Einstein's relations and significance; Conditions for laser action – population inversion and pumping methods; Active components of a laser – gain medium, optical resonator, pumping source; Ruby laser – construction, working principle, and output characteristics; He–Ne laser – construction, operation, and applications; Advantages and limitations of common lasers.

Unit IV **8 Hours**

Electrostatics and Magnetostatics – Coulomb's law, Electric field and field due to a point charge, Electric flux; Gauss's theorem and applications – infinitely long straight wire; Electric potential and potential difference; Biot–Savart law and application to current carrying circular loop; Ampere's law and applications – infinitely long straight wire and solenoids.

Unit V **8 Hours**

Superconductivity – Essential properties of superconductors – zero resistivity, perfect diamagnetism; London equations and their implications; Penetration depth and coherence length; Meissner effect – experimental observation and significance; Classification – Type I and Type II superconductors, critical magnetic field and temperature; Characteristics of superconductors in the superconducting state; High-temperature superconductors – brief introduction; Applications – power transmission, magnetic levitation, superconducting magnets, and medical imaging.

Suggested Readings

1. **Text Book**
   1. Introduction to fiber: A.K.Ghatak & K.Thyagarajan
   2. Concepts of modern Physics, A. Beiser, TMH.

1. **Reference Books**
   1. Lasers (Theory and Application): K. Thyagarajan & A. K. Ghatak
   2. Introduction to Electrodynamics ,David J. Griffith (PH I).
   3. Solid State Physics, A.J. Dekker.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	2	-	-	-	-	-	-	-
CO2	2	3	3	2	2	-	-	-	-	-	-	-
CO3	2	-	3	1	-	-	-	-	-	-	-	1
CO4	2	-	2	2	3	-	-	-	-	-	-	2
CO5	2	2	-	2	3	-	-	-	-	-	-	1
AVG.	2.2	1.4	1.6	1.4	2	0	0	0	0	0	0	0.8

List of Practicals

1. Build and analyze simple DC circuits using resistors, capacitors, and power supplies.
2. Use breadboards and basic electronic components to construct and test circuit configurations.
3. Practice using a multimeter to measure voltage, current, and resistance in different circuit elements.
4. Learn how to use an oscilloscope to observe waveforms and measure signal parameters.
5. Build and test diode-based rectifiers, such as half-wave and full-wave rectifiers.
6. Explore diode clamping and clippers to understand their applications in signal conditioning.
7. Construct and analyze transistor amplifier circuits using bipolar junction transistors (BJTs) or field-effect transistors (FETs).
8. Observe the amplification and signal conditioning capabilities of transistors.
9. Learn how to calculate voltage, current, and resistance in simple DC circuits.
10. Explore the behavior of diodes and their role in rectifying AC to DC.
11. Practice solving circuit problems using Ohm's Law and Kirchhoff's Laws.
12. Build and test basic rectifier circuits using diodes.

Course Outcomes (COs) – Physics I Lab

After completing the Physics I Laboratory, students will be able to:
CO1: Build and analyze basic DC circuits using resistors, capacitors, diodes, and power supplies.
CO2: Use laboratory instruments such as multimeters and oscilloscopes for electrical measurements.
CO3: Design and test rectifier, clipper, and clamper circuits for signal conditioning.
CO4: Construct and analyze transistor-based amplifier circuits and study their characteristics.
CO5: Apply Ohm’s Law and Kirchhoff’s Laws to solve and verify practical circuit problems.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	2	2	–	–	–	–	–	–	–
CO2	2	2	–	3	3	–	–	–	–	–	–	1
CO3	2	3	3	2	2	–	–	–	–	–	–	1
CO4	2	3	3	2	2	–	–	–	–	–	–	1
CO5	3	2	–	2	2	–	–	–	–	–	–	2
AVG.	2.4	2.4	1.6	2.2	2.2	0	0	0	0	0	0	1

Course Name: Basic Electrical Engineering	**Credit Scheme**				**Evaluation Components**			
Course Code: ESC102	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To introduce the fundamental concepts of electrical engineering and electrical circuit analysis.
2. To understand DC and AC circuit behavior using basic electrical laws and theorems.
3. To explain the working principles of electrical machines and measuring instruments.
4. To familiarize students with basic electrical wiring, safety practices, and components.
5. To develop practical skills in constructing, testing, and analyzing simple electrical circuits.

Course Outcomes: After completing this course the student will be able to

CO1: Explain basic electrical quantities, laws, and circuit elements used in electrical systems.
CO2: Analyze DC and AC circuits using Kirchhoff's laws and network theorems.
CO3: Describe the construction, working principles, and applications of electrical machines.
CO4: Interpret the working and usage of electrical measuring instruments and sensors.
CO5: Apply electrical safety practices and perform basic electrical wiring and circuit experiments.

Detailed Syllabus:

Unit I **8 Hours**

Introduction to electrical engineering; Electrical quantities – current, voltage, power, energy; SI units and standards; Active and passive elements; Voltage and current sources; Ohm's Law; Kirchhoff's Current Law (KCL) and Voltage Law (KVL); Series and parallel circuits; Power calculation and energy consumption.

Unit II **8 Hours**

Resistive circuits; Star–Delta and Delta–Star transformations; Network theorems – Superposition theorem, Thevenin's theorem, Norton's theorem (numerical applications); Maximum power transfer theorem; Introduction to capacitors and inductors; Transient response of RC and RL circuits.

Unit III **8 Hours**

Alternating quantities; Sinusoidal waveforms; RMS and average values; Phasor representation; AC circuits with resistance, inductance, and capacitance; Impedance and admittance; Power factor; Real, reactive, and apparent power; Power triangle; Resonance in RLC circuits (series and parallel).

Unit IV **8 Hours**

Introduction to electrical machines; Construction, working principle, and applications of DC machines; Transformers – construction, EMF equation, losses, efficiency, and applications; Introduction to AC machines – induction motor and synchronous motor (basic concepts and applications).

Unit V **8 Hours**

Electrical measuring instruments – ammeter, voltmeter, wattmeter, energy meter; Moving coil and moving iron instruments; Errors and calibration; Basics of electrical wiring – domestic wiring, earthing, circuit protection devices (MCB, ELCB, fuses); Electrical safety practices and standards.

Suggested Readings

Text Book

1. D.P. Kothari & I.J. Nagrath, Basic Electrical Engineering, McGraw Hill Education.
2. B.L. Theraja & A.K. Theraja, Elect*rical Technology*, S. Chand & Company.

Reference Books

1. V.K. Mehta & Rohit Mehta, Basic Electrical Engineering, S. Chand.
2. H. Cotton, Electrical Technology, CBS Publishers.
3. C.L. Wadhwa, Electrical Power Systems, New Age International.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	3	2	–	–	2	–	–	–	–	–	–	–	1	–	–
CO2	3	3	2	2	2	–	–	–	–	–	–	–	2	–	–
CO3	2	2	3	2	–	–	–	–	–	–	–	1	1	–	–
CO4	2	2	2	3	2	1	–	–	–	–	–	1	1	–	–
CO5	2	2	–	2	3	2	–	–	–	–	–	2	1	1	–
AVG.	2.4	2.2	1.8	2.2	1.8	1	0	0	0	0	0	0.8	1.2	0.2	0

List of Practicals

1. Verification of Ohm's Law using DC power supply and resistive circuits.
2. Measurement of voltage, current, and resistance using a digital multimeter.
3. Verification of Kirchhoff's Voltage Law and Current Law.
4. Study of series and parallel resistor circuits and power calculation.
5. Construction and testing of RC and RL circuits.
6. Measurement of AC voltage and current using AC meters.
7. Study of single-phase transformer (open-circuit and short-circuit tests – demonstration).

8. Wiring of a lamp, switch, fuse, and socket (domestic wiring).
9. Measurement of power using wattmeter.
10. Study of earthing and electrical safety devices.

Laboratory Course Outcomes (LCOs)

LCO1: Identify and use basic electrical components, tools, and measuring instruments safely.
LCO2: Construct and analyze simple DC and AC electrical circuits using standard laboratory practices.
LCO3: Perform experiments on diodes, rectifiers, and transistor-based circuits and interpret results.
LCO4: Measure electrical parameters using multimeters and oscilloscopes and validate theoretical concepts.
LCO5: Demonstrate electrical safety practices, teamwork, documentation, and ethical conduct.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	3	2	–	2	2	–	–	–	–	–	–	1	3	2	–
CO2	3	3	2	3	2	–	–	–	–	–	–	1	3	3	2
CO3	2	2	3	2	2	–	–	–	–	–	–	1	2	2	3
CO4	2	2	2	3	3	1	–	–	–	–	–	1	2	2	2
CO5	2	2	–	2	3	2	–	–	–	–	–	2	2	2	–
AVG.	2.4	2.2	1.4	2.4	2.4	0.6	0	0	0	0	0	1.2	2.4	2.2	1.4

Course Name: Mathematics-I	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC103	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	4	0	0	50	0	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1. Develop a strong foundation in algebra, trigonometry, and calculus for engineering applications.
2. Apply differentiation and integration methods to solve real-world engineering problems.
3. Analyze the behavior of functions using limits, continuity, and differentiability concepts.
4. Enhance problem-solving skills through the application of mathematical models and techniques.
5. Foster analytical thinking for interpreting and solving problems in science and engineering contexts.

Course Outcomes: After completing this course the student will be able to

CO1: Apply differentiation and integration techniques to solve engineering problems, such as finding rates of change, optimization, and area/volume calculations.
CO2: Demonstrate proficiency in solving equations and inequalities involving algebraic, exponential, logarithmic, and trigonometric functions.
CO3: Understand and apply the concept of limits to analyze the behavior of functions, including continuity and differentiability.
CO4: Solve problems involving related rates, curve sketching, and optimization using calculus techniques.
CO5: Apply calculus techniques to solve engineering problems involving rates of change, motion, growth, and decay.

Detailed Syllabus:

Unit I **8 Hours**

Matrices: Elementary transformations, Inverse of a matrix, Rank of matrix, Solution of system of linear equations, Characteristic equation, Cayley-Hamilton Theorem and its application, Linear Dependence and Independence of vectors, Eigen values and Eigen vectors, Complex Matrices, Hermitian, Skew-Hermitian and Unitary Matrices, Applications to Engineering problems.

Unit II **8 Hours**

Differential Calculus- I: Successive Differentiation (nth order derivatives), Leibnitz theorem, Curve tracing, Partial derivatives, Euler's Theorem for homogeneous functions, Total derivative, Change of variables.

Unit III **8 Hours**

Differential Calculus-II: Expansion of functions by Taylor's and Maclaurin's theorems for functions of one and two variables, Maxima and Minima of functions of several variables, Lagrange's method of multipliers, Jacobians, Approximation of errors.

Unit IV **8 Hours**

Multivariable Calculus-I: Multiple integration: Double integral, Triple integral, Change of order of integration, Change of variables, Beta and Gamma function and their properties, Dirichlet's integral and its applications to area and volume, Liouville's extensions of Dirichlet's integral.

Unit V **8 Hours**

Vector Calculus :Vector differentiation: Gradient, Curl and Divergence and their Physical interpretation, Directional derivatives Vector Integration: Line integral, Surface integral, Volume integral, Gauss's Divergence theorem, Green's theorem and Stoke's theorem (without proof) and their applications.

Suggested Readings

Text Books:

1. Reena Garg Engineering Mathematics Khanna Book Publishing company
2. G.B.Thomos and R.L. Finney , Calculus and Analytic Geometry 9th Edition Pearson Reprint

Reference Books:

1. Erwin Kreyszig Advanced Engineering Mathematics 9th Edition John wiley &sons
2. Veerarajan T, Engineering Mathematics for first year, McGraw-Hill New Delhi
3. Ramana B.V. Higher Engineering Mathematics. Tata McGraw New Delhi 11th Reprint
4. N.P. Bali , A text Book of Engineering Mathematics Laxmi publications Reprint
5.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	-	-	-	-	-	-	-	3
CO2	3	3	3	3	-	-	-	-	-	-	-	3
CO3	3	3	3	2	-	-	-	-	-	-	-	3
CO4	3	1	2	1	-	-	-	-	-	-	1	3
CO5	2	2	2	2	-	-	-	-	-	-	2	3
AVG.	2.8	2.4	2.6	2.2							0.6	3

Course Name: Design Thinking	Credit Scheme				Evaluation Components			
Course Code: ESC 104	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	2	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. Develop awareness of diverse learning styles, memory techniques, and their application in engineering problem-solving.
2. Understand and analyze emotional experiences to empathize with users and enhance product design.
3. Foster creativity by applying the design thinking process and innovation cycle for product development.
4. Apply appropriate frameworks, strategies, and techniques to create, test, and refine prototypes.
5. Recognize individual differences and their influence on decision-making to improve user experience and satisfaction.

Course Outcomes: After completing this course the student will be able to

CO1: Compare and classify the various learning styles and memory techniques and Apply them in their engineering education
CO2: Analyze emotional experience and Inspect emotional expressions to better understand users while designing innovative products
CO3: Develop new ways of creative thinking and Learn the innovation cycle of Design FThinking process for developing innovative products
CO4: Propose real-time innovative engineering product designs and Choose appropriate frameworks, strategies, techniques during prototype development
CO5: Perceive individual differences and its impact on everyday decisions and further Create a better customer experience.

Detailed Syllabus:

Unit I **8 Hours**

An Insight to Learning: Understanding the Learning Process, Kolb's Learning Styles, Assessing and Interpreting; Remembering Memory: Understanding the Memory process, Problems in retention, Memory enhancement techniques.

Unit II **6 Hours**

Emotions: Experience & Expression: Understanding Emotions: Experience & Expression, Assessing Empathy, Application with Peers; Basics of Design Thinking: Definition of Design Thinking, Need for Design Thinking,

Objective of Design Thinking, Concepts & Brainstorming, Stages of Design Thinking Process- Empathize, Define, Ideate, Prototype, Test.

Unit III **6 Hours**

Being Ingenious & Fixing Problem: Understanding Creative thinking process, Understanding Problem Solving, Testing Creative; Problem Solving Process of Product Design Process of Engineering Product Design, Design Thinking Approach, Stages of Product Design, Examples of best product designs and functions, Assignment – Engineering Product Design.

Unit IV **8 Hours**

Prototyping & Testing : What is Prototype? Why Prototype? Rapid Prototype Development process, Testing, Sample Example, Test Group Marketing Celebrating the Difference Understanding Individual differences & Uniqueness, Group Discussion and Activities to encourage the understanding, acceptance and appreciation of Individual differences.

Unit V **8 Hours**

Design Thinking and Customer Centricity: Practical Examples of Customer Challenges, Use of Design Thinking to Enhance Customer Experience, Parameters of Product experience, Alignment of Customer Expectations with Product Design Feedback, Re-Design & Re-Create: Feedback loop, Focus on User Experience, Address "ergonomic challenges, User focused design, rapid prototyping & testing, final product, Final Presentation – "Solving Practical Engineering Problem through Innovative Product Design & Creative Solution".

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	2	2	-	2	-	1	3	3	2	2
CO2	2	3	2	2	-	3	2	1	3	3	2	2
CO3	2	3	3	3	2	2	2	1	3	3	2	3
CO4	2	3	3	3	2	2	2	1	3	3	2	3
CO5	2	3	2	2	-	3	2	1	3	3	2	3
AVG.	2	3	2.4	2.4	0.8	2.4	1.6	1	3	3	2	2.6

List of Practicals:

1. Understanding different problems in organization.
2. Understanding different problems in nature.
3. Analyzing Strength, weakness, opportunity and threat (SWOT Analysis).
4. Analyzing memory improving techniques.
5. Investigating an issue by asking why and how? Five times to arrive at the root cause and define the challenge.
6. Clustering- group ideas according to emerging themes.
7. Customer Journey- considers the user or customer of a product or service and maps their touch point.
8. Dot voting- Splitting ideas and finding the best solution.
9. Explain to a stranger-Outline a challenge for a person who is not familiar with the issues.
10. Mind map-Sort information according to themes to build a picture of an issue or problem.
11. Peer View- Feedback from people outside of the development team.
12. Understanding the problem and framing the problem statement.
13. Simple Random Sampling for given data.
14. Making wire frame prototype models.

Course Outcomes (COs)

After completing this course, the student will be able to:

CO1: Compare and classify various learning styles and memory techniques and apply them in engineering education.
CO2: Analyze emotional experiences and inspect emotional expressions to empathize with users during innovative product design.
CO3: Develop creative thinking skills and apply the innovation cycle of the Design Thinking process.
CO4: Propose real-time innovative engineering product designs using suitable frameworks, strategies, and prototyping techniques.
CO5: Perceive individual differences and create improved customer-centric design solutions.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	2	2	–	2	–	1	3	3	2	2
CO2	2	3	2	2	–	3	2	1	3	3	2	2
CO3	2	3	3	3	2	2	2	1	3	3	2	3
CO4	2	3	3	3	2	2	2	1	3	3	2	3
CO5	2	3	2	2	–	3	2	1	3	3	2	3
AVG.	2	3	2.4	2.4	0.8	2.4	1.6	1	3	3	2	2.6

Course Name: Engineering Graphics and Design	**Credit Scheme**				**Evaluation Components**			
Course Code: ECS 105	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	1	0	4	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To develop the ability to interpret and represent engineering ideas using the universal language of technical drawings.
2. To enhance visualization skills and spatial reasoning for effective engineering communication.
3. To introduce fundamental concepts of engineering design and its applications in real-life and industrial contexts.
4. To familiarize students with CAD tools for creating accurate and efficient engineering drawings.
5. To provide exposure to orthographic, isometric, and projection methods used in engineering graphics.

Course Outcomes: After completing this course the student will be able to

CO1: To understand the International language of engineering graphics to express worldwide ideas, convey instruction while carrying out engineering jobs.
CO2: To develop skilled exposure to engineering communication.
CO3: Introduction to engineering design its place in (life) or industry.
CO4: Exposure to CAD. computer aided design or creating working drawing.
CO5: Exposure to isometric projectiles.

Detailed Syllabus:

Unit I **8 Hours**

Introduction of engineering graphics, Importance of engineering graphics,Tools and technique used in engineering graphics, Geometrical Instrument. Lay out of drawing standard sheet size,Title block size and importance of title block.

Unit II **8 Hours**

Different types of line, Free hand (single stroke vertical letter given size) capital letter, small letter and numbers drafting as per Standard ratio. Types of line and its use in drawing. Convention of material, types of scale and measurement, Use of Scale metric measures, Quadrants, Projection–1st angle projection its symbol. III rd angle projection its symbol.

Unit III **8 Hours**

Type of dimension, Dimensioning technique and tolerance, Arrow heads and standard size. Projection – Orthographic (2D) and isometric (3D) projection, conversion of orthographic projection to isometric projection, Projection of solid.

Unit IV **8 Hours**

Construction of isometric projection simple objects. Conversion of orthographic to isometric, Sectional view, Principal of sectional view, construction of sectional view of simple object, Type of sectional view, convention used in sectional view.

Unit V **8 Hours**

AUTO CAD, Introduction of cad, Tool bar, Use of various command and tool bar Practice, short commands. Set up the drawing file. Draw a simple 2 Dimensional view, circle, Hexagon, pentagon, square, rectangular etc. (Orthographic drawing),print or plot of the drawing.

Suggested Reading

Text Book

1. Latest Edition **Punjab Rai, Engineering drawing I**

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	-	-	-	-	-	-	-	-	-	-	-
CO2	1	-	-	-	-	1	-	1	-	-	-	-
CO3	1	-	-	-	-	1	-	1	1	-	-	-
CO4	1	-	-	-	-	-	2	-	2	1	1	-
CO5	1	-	-	-	-	-	2	-	2	1	1	1
AVG.	1	0	0	0	0	0.4	0.8	0.4	1	0.4	0.4	0.2

List of Practicals:

1. Introduction Of engineering graphics tools.
2. Lay out the drawing s sheet and title block Recommended size.
3. Convention of line.
4. Conventional of material.
5. Symbol for general reference electrical symbol.
6. Letter drafting, capital letter, small letter, numbers.
7. Dimensioning technique and tolerance, use Arrow head.
8. 1st angle projection its symbol.
9. III rd angle projection its symbol.
10. Construction of Orthographic projection.
11. Construction of Isometric projection.

12. Simple object. Conversion of orthographic to isometric.
13. Construction of a sectional view of a simple object.
14. Draw a simple 2 Dimensional view in Autocad.

Course Outcomes (COs)

After completing this course, the student will be able to:

CO1: Understand and apply the international language of engineering graphics to convey technical ideas accurately.
CO2: Develop effective engineering communication skills through technical drawings.
CO3: Recognize the role of engineering design in real-life and industrial applications.
CO4: Use CAD tools to create accurate 2D working drawings.
CO5: Create and interpret isometric and orthographic projections of engineering components.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	–	–	–	–	–	–	–	–	–	–	–
CO2	1	–	–	–	–	1	–	1	–	–	–	–
CO3	1	–	–	–	–	1	–	1	1	–	–	–
CO4	1	–	–	–	–	–	2	–	2	1	1	–
CO5	1	–	–	–	–	–	2	–	2	1	1	1
AVG.	1	0	0	0	0	0.4	0.8	0.4	1	0.4	0.4	0.2

Course Name: Introduction to the Computer Programming using C	**Credit Scheme**			
Course Code: CSE C106	**Total Credit**	**L**	**T**	**P**
	4	**3**	**0**	**2**
Prerequisite Course and code (if any):				

Course Objective: The "Programming in C" course aims to teach participants the fundamentals of C programming language. By the end of the course, students will be able to write, debug, and understand basic C programs. The course focuses on building problem-solving skills and providing a strong programming foundation.

Course Outcomes: On completion of this course, the students will be able to:

CO1: Develop and translate the Algorithms to Programs & Execution (in C Language).
CO2: Implement Conditional Branching, Iteration and Recursion.
CO3: Decompose a Problem into Functions and Synthesize a Complete Program Using Divide and Conquer Approach.
CO4: Use Arrays to Develop Algorithms and Programs.Use Pointers to Develop Algorithms and Programs. Use Structure and Union to Develop Algorithms and Programs
CO5: Concept of File Handling. Create User Defined Functions to Develop New Header Files. Exercise User Defined Functions to solve real time problems

Detailed Syllabus:

Unit I **10 Hours**

Introduction of Computer: - Number System, Types of Computer, Generation of Computer, Flowchart and Algorithms.**Introduction to Programming:** Concept of assembler, compiler, interpreter, loader and linker.**Idea of Algorithm:** Representation of Algorithm, Flowchart, Pseudo code with examples, From algorithms to programs, source code.**Programming Basics:** Structure of C program: writing and executing the first C program, Syntax and logical errors in compilation, object and executable code.
Components of C language: Standard I/O in C, Fundamental data types, Variables and memory locations, Storage classes

Unit II **8 Hours**

Arithmetic expressions & Conditional Branching: Arithmetic expressions and precedence: Operators and expression using numeric and relational operators, mixed operands, type conversion, logical operators, bit operations, assignment operator, operator precedence and associatively. **Conditional Branching:** Applying if and switch statements, nesting if and else, use of break and default with switch.

Unit III **9 Hours**

Loops & Functions: Iteration and loops: use of while, do while and for loops, multiple loop variables, use of break and continue statements.

Functions: Introduction, types of functions, functions with array,passing parameters to functions, call by value, call by reference, recursive functions.

Unit IV **8 Hours**

Arrays: Array notation and representation, manipulating array elements, using multi dimensional arrays. Character arrays and strings, Structure, union, enumerated data types, Array of structures, Passing arrays to functions.

Unit V **10 Hours Pointer&**

File Handling:Pointers: Introduction, declaration, applications, Introduction to dynamic memory allocation (malloc, calloc, realloc, free), Use of pointers in self-referential structures, notion of linked list (no implementation)

File handling: File I/O functions, Standard C preprocessors, defining and calling macros, command-line arguments.

Suggested Readings

1. Text Books

1. Programming in ANSI C – E. Balagurusamy, McGraw Hill
2. Let Us C – Yashavant Kanetkar, BPB Publications

2. Reference Books

1. The C Programming Language – Brian W. Kernighan & Dennis M. Ritchie, Pearson
2. C How to Program – Paul Deitel & Harvey Deitel, Pearson
3. Programming with C – Byron Gottfried, Schaum's Outline Series

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	2	-	-	-	-	-	-	-
CO2	2	3	3	2	2	-	-	-	-	-	-	-
CO3	2	-	3	1	-	-	-	-	-	-	-	1
CO4	2	-	2	2	3	-	-	-	-	-	-	2
CO5	2	2	-	2	3	-	-	-	-	-	-	1
AVG.	2.2	1.4	1.6	1.4	2	0	0	0	0	0	0	0.8

PROGRAMMING FOR PROBLEM SOLVING

PRACTICAL LIST

1. Write a program to calculate the area of triangle using formula area=√s(s-a)(s-b)(sc)

2. Basic salary of an employee is input through the keyboard. The DA is 25% of the basic salarywhile the HRA is 15% of the basic salary. Provident Fund is deducted at the rate of 10% of the gross salary (BS+DA+HRA). Program to calculate the Net Salary.

3. Write a program to find the largest of three numbers using nested if else.

4. Write a program to receive marks of physics, chemistry & maths from user & check its eligibility for course if

   a) Marks of physics > 40

   b) Marks of chemistry > 50

c) Marks of math's > 60

d) Total of physics & math's marks > 150 or

e) Total of three subjects marks > 200

5. Write a program to construct a Fibonacci series upto n terms.

6. Write a program to find whether the number is Armstrong number.

7. Write a program to generate sum of series 1!+2!+3!+---------- n!

8. Write a program to print the entire prime no between 1 and 300.

9. Write a program to print area of rectangle using function & return its value to main function.

10. Write a program to calculate the factorial for given number using function.

11. Write factorial function & use the function to find the sum of series S=1!+2!+-----n!.

12. Write a program to find the factorial of given number using recursion.

13. Write a program to convert decimal number in to binary number

14. Write a program to find the transpose of a given matrix & check whether it is symmetric or not.

15. Write a program in C to check whether the given string is a palindrome or not.

16. Define a structure that can describe a hotel. It should have the member that includes the name, address, grade, room charge and number of rooms. Write a function to print out hotel of given grade in order of room charges

17. Write a c program to copy & count the character content of one file says a.txt to another file b.txt.

18. Write the following C program using pointer: a) To sort the list of numbers through pointer b) To reverse the string through pointer.

19. Write a program to find the largest no among 20 integers array using dynamic memory allocation.

20. Write a program to find the factorial of given number using command line argument.

Course Outcomes (COs)

After completing this course, the student will be able to:

CO1: Identify the basic components of a computer system and operate the computing environment for program development.
CO2: Develop simple C programs using fundamental constructs such as data types, operators, and control statements.
CO3: Implement C programs using arrays, strings, functions, and pointers to solve basic computational problems.
CO4: Perform input–output operations and apply debugging techniques to detect and correct logical and syntactical errors.
CO5: Design, compile, execute, and test C programs to solve real-world engineering problems following good programming practices.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	2	2	–	–	–	–	–	–	–
CO2	2	2	–	3	3	–	–	–	–	–	–	1
CO3	2	3	3	2	2	–	–	–	–	–	–	1
CO4	2	3	3	2	2	–	–	–	–	–	–	1
CO5	3	2	–	2	2	–	–	–	–	–	–	2
AVG.	2.4	2.4	1.6	2.2	2.2	0	0	0	0	0	0	1

Course Name: Communication skills I	**Credit Scheme**				**Evaluation Components**			
Course Code: CS 001	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	2	0	0	50	0	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1.To improve communication skills across various professional contexts.

2.To develop understanding of ethical principles and their workplace applications.

3.To enhance interpersonal skills and teamwork abilities.

4.To equip students with essential skills for career success.

5.To foster responsibility, ethical decision-making, and a positive work environment.

Course Outcomes: After completing this course, the student will be able to

CO1: Demonstrate effective communication skills in professional situations.

CO2: Apply ethical principles to decision-making in workplace scenarios.

CO3: Work collaboratively and productively within diverse teams.

CO4: Utilize workplace skills that support career growth and professional success.

CO5: Contribute to creating and maintaining a positive and ethical work environment

Unit I **7 Hours**

Fundamentals of Industrial Communication – Introduction to workplace communication, communication models, and the role of effective communication in professional settings; Verbal communication – oral and written skills including presentations, report writing, and business correspondence; Non-verbal communication – body language, tone, and visual cues in professional interactions.

Unit II **8 Hours**

Interpersonal and Cross-Department Communication – Building workplace relationships, active listening, teamwork, and conflict resolution; Communication within and across departments – internal memos, reports, and upward communication with management; Communication with stakeholders including customers, shareholders, and government agencies; Drafting sales letters and handling customer complaints.

Unit III **8 Hours**

Global, Cultural, and Technological Communication – Cross-cultural communication and sensitivity to diverse perspectives; Adapting communication styles for global contexts; Effective use of digital tools – email, messaging platforms, and collaboration software; Public speaking and presentation skills for formal and informal contexts.

Unit IV **8 Hours**

Introduction to Workplace Ethics – Definition and importance of ethics in the workplace; Ethical dilemmas and decision-making frameworks; Core ethical principles – honesty, integrity, fairness, responsibility, and respect; Application of ethics in business functions – marketing, finance, and HR; Workplace conduct – policies, codes of conduct, and professional standards.

Unit V **8 Hour**

Advanced Workplace Ethics and Legal Framework – Social responsibility and its role in business; Promoting fairness, diversity, and inclusion; Whistleblowing and procedures for reporting unethical behavior; Ethical leadership and fostering an ethical organizational culture; Overview of legal frameworks and regulations related to workplace ethics.

Suggested Books:

Text Books:

1. Lesikar, R. V., Pettit, J. D., & Flately, M. E. *Business Communication: Making Connections in a Digital World*. McGraw-Hill Education.
2. Guffey, M. E., & Loewy, D. *Essentials of Business Communication*. Cengage Learning.

Reference Books:

1. Boatright, J. R., Smith, J. D., & Dorsey, E. *Ethics and the Conduct of Business*. Pearson Education.

2. Weiss, J. W. *Business Ethics: A Stakeholder and Issues Management Approach*. Cengage Learning.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	-	-	-	-	-	-	-	-	-	-	-
CO2	1	-	-	-	-	1	-	1	-	-	-	-
CO3	1	-	-	-	-	1	-	1	1	-	-	-
CO4	1	-	-	-	-	-	2	-	2	1	1	-
CO5	1	-	-	-	-	-	2	-	2	1	1	1
AVG.	1	0	0	0	0	0.4	0.8	0.4	1	0.4	0.4	0.2

Semester: 2nd								
Course Name: Chemistry I	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC 111	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To impart fundamental knowledge of atomic and molecular structure, bonding, and spectroscopy for understanding chemical behavior.
2. To develop an understanding of water chemistry, treatment methods, and its industrial as as well as domestic applications.
3. To provide insights into polymer chemistry, classification, and applications in engineering materials.
4. To familiarize students with the chemistry of fuels, including conventional and alternative energy sources.
5. To introduce the environmental and industrial significance of chemical processes, emphasizing sustainable technological applications.

Course Outcomes:

After completing this course the student will be able to

CO1: Analyze molecular structure and properties using bonding concepts and spectroscopic techniques.

CO2: Explain water chemistry and apply purification methods for real-world industrial and domestic purposes.

CO3: Identify, classify, and evaluate different types of polymers and their engineering applications.

CO4: Compare and differentiate conventional and non-conventional fuels with respect to efficiency and environmental impact.

CO5: Apply chemical principles to address environmental concerns and select suitable materials in engineering applications.

Detailed Syllabus:

Unit 1 **8 Hours**

MOLECULAR STRUCTURE AND SPECTROSCOPIC TECHNIQUES Molecular Orbital Theory, Formation of homo and heteronuclear diatomic molecules Hydrogen Bonding and its application Metallic Bonding (Band theory) and application to conductors, semiconductors and insulators Nanoscale Materials - Properties and applications Basic Principles of spectroscopy and its applications for molecular structure.

Unit 2 8 Hours

WATER TECHNOLOGY Hardness of water: Causes, Types, Measurement, Boiler troubles: Sludges, Scales and Caustic Embrittlement. Softening of water by L-S Process, Zeolite Process and Reverse Osmosis Process, Ion Exchange Process, Calgon Process Numerical Problems based on L-S Process, Zeolite Process and hardness of water. Introduction to the membrane concept for the treatment of microplastics from water.

Unit 3 8 Hours

POLYMERS : Definition, degree of polymerization, functionality of monomer, Classification of polymers with examples, Types of polymerizations – addition and condensation polymerization with examples. Mechanism of addition polymerization. Plastics: Definition and characteristics-thermoplastic and thermosetting plastics, preparation, properties, and applications of PVC and Bakelite Fibers: Characteristics of fibers – preparation, properties and applications of Nylon and Dacron. Conducting polymers: Characteristics and Classification of conducting polymers with examples. Biodegradable polymers: Concept and advantages – Preparation of Polylactic acid and poly vinyl alcohol and their applications. Liquid Crystalline Polymers: Characteristics, classification with examples and their applications.

Unit 4 8 Hours

FUELS AND RENEWABLE SOURCE OF ENERGY Fuels Definition, Classification and Characteristics of a good fuel, Calorific value and its determination by Bomb Calorimeter, Numerical problems on Bomb Calorimeter, Composition and uses of Natural gas, CNG, LPG. Renewable Energy Sources:Solar energy, wind energy, hydroelectric and geothermal. Biofuels as alternative sources of energy (biomass, biogas).

Unit 5 10 Hours

ELECTROCHEMISTRY & ITS APPLICATIONS Electrode potential, standard electrode potential, factors affecting the electrode potential of a cell. Nernst equation: Electrochemical series and its application, Electrochemical cell: Daniel cell, Concentration cells, electrolyte concentration cell Numerical problems based on electrode potential and emf of a cell. Fuel Cells: Introduction, Principles, Classification, and application Corrosion its causes and effects, Theories of corrosion – Chemical & Electrochemical corrosion.

Suggested Books:

Text Books:

1. Engineering Chemistry, Dhanpat Rai Publishing, Jain and Jain
2. Engineering Chemistry,Khanna Publishers, O.P. Agarwal

Reference Books:

1. Engineering Chemistry, Dhanpat Rai & Sons, Shashi Chawla
2. Understanding Chemistry, University Press, India’s leading chemist, C.N. R. Rao

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	3	2	-	-	-	-	-	-	2
CO2	3	3	2	3	-		-	-	-	-	-	2
CO3	3	2	3	-	-	-	-	-	-	-	-	2
CO4	3	3	-	-	-	2	3	-	-	-	-	2
CO5	3	2	2	3	-	3	3	-	-	-	-	3
AVG.	3	2.4	1.4	2. 25	0.4	1	1.2	-	-	-	-	2.2

List of Practicals:

1. To determine the alkalinity of the given water sample containing carbonate (CO 2-)ions and bicarbonate (HCO -)ions by titrating it against standard HCl solution [N/10] using phenolphthalein and methyl orange as indicators.
2. To determine the chloride ion (Cl-) content in the given water sample by Argentometric method (Mohr's method) using N/50 AgNO3 as a standard solution and potassium chromate (K2CrO4) as an internal indicator.
3. To determine the temporary and permanent hardness of a given water sample by titrating it against standard solution of M/100 Ethylene Diamine Tetracetic Acid (EDTA) using Eriochrome black-T (EBT) as an internal indicator.
4. To determine the coefficient of viscosity of the given sample solution by Ostwald's viscometer (Viscosity of water = 0.0101 Poise).
5. To determine the ferrous ion (Fe++) content in a given sample solution of Mohr's salt (FeSO4.(NH4)2SO4.6H2O) by titrating it against standard N/30 potassium dichromate (K2Cr2O7) solution by using potassium ferricyanide K3 [Fe (CN) 6] as an external indicator.
6. To determine the surface tension of the given sample solution by drop number method.
7. To determine the acid value of oil.
8. To determine the strength of unknown HCl solution by titrating it against N/10 NaOH solution with the help of the pH meter.
9. Synthesis of phenol-formaldehyde resin.
10. To determine the alkalinity of the given water sample containing carbonate (CO3)ions and hydroxide (OH) ions by titrating it against standard HCl solution [N/10] using phenolphthalein and methyl orange as indicators.
11. To determine the rate constant of a reaction To determine the Copper (Cu++) ion content in the given sample of copper ore (blue vitriol) by titrating it against standard N/30 sodium thiosulphate solution using starch as an indicator by Iodometric titration.

Course Outcomes

Course Name: Chemistry Laboratory

After completing this course, the student will be able to:

CO1: Identify and handle laboratory chemicals, glassware, and instruments by following proper safety practices.
CO2: Perform quantitative and qualitative chemical experiments using standard laboratory techniques.
CO3: Analyze experimental data and calculate results related to concentration, pH, hardness, and reaction parameters.
CO4: Interpret experimental observations and prepare structured laboratory reports with correct units and conclusions.
CO5: Apply basic principles of chemistry to materials, corrosion, water analysis, and environmentally relevant problems.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	3	2	-	-	-	-	-	-	2
CO2	3	3	2	3	-		-	-	-	-	-	2
CO3	3	2	3	-	-	-	-	-	-	-	-	2
CO4	3	3	-	-	-	2	3	-	-	-	-	2
CO5	3	2	2	3	-	2	2	-	-	-	-	2
AVG.	3	2.4	1.4	2.25	0.4	2	2.5	-	-	-	-	5

Course Name: Mathematics II (Mathematical Foundation of Data Science)	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC 112	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	1	0	50	-	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1. To develop the ability to solve higher-order differential equations and apply them to real engineering problems.
2. To enable students to analyze the convergence of sequences and series using standard tests and represent functions with Fourier series.
3. To introduce the concepts of functions of complex variables and techniques to identify and construct analytic functions.
4. To provide knowledge of complex integration, Cauchy's theorems, series expansion, and residue calculus for problem-solving.
5. To equip students with the application of Laplace transforms for solving ordinary and simultaneous differential equations in engineering contexts.

Course Outcomes: After completing this course the student will be able to

CO1: Solve higher-order linear differential equations and apply them to engineering problems.
CO2: Analyze convergence of sequences and series using various tests and represent functions using Fourier series.
CO3: Understand the concepts of analytic functions, conformal mappings, and transformations in complex variables.
CO4: Evaluate complex integrals using Cauchy's theorems, series expansions, and residue methods.
CO5: Apply Laplace transforms to solve ordinary and simultaneous differential equations in engineering applications.

Detailed Syllabus:

UNIT I **9 Hours**

Ordinary Differential Equation of Higher Order: Linear differential equation of nth order with constant coefficients, Simultaneous linear differential equations, Second order linear differential equations with variable coefficients, Solution by changing independent variable, Method of variation of parameters, Cauchy-Euler equation, Application of differential equations in solving engineering problems.

UNIT II **9 Hours**

Sequences and Series: Definition of Sequence and series with examples, Convergence of series, Tests for convergence of series, Ratio test, D' Alembert's test, Raabe's test, Comparison test. Fourier series, Half range Fourier sine and cosine series.

UNIT III **8 Hours**

Complex Variable–Differentiation Functions of complex variable, Limit, Continuity and differentiability, Analytic functions, Cauchy- Riemann equations (Cartesian and Polar form), Harmonic function, Method to find Analytic functions, Milne's Thompson Method, Conformal mapping, Mobius transformation and their properties**.**

UNIT IV **8 Hours**

Complex Variable –Integration Complex integration, Cauchy- Integral theorem, Cauchy integral formula, Taylor's and Laurent's series, singularities and its classification, zeros of analytic functions, Residues, Cauchy's Residue theorem and its application.

UNIT V **8 Hours**

Laplace Transform: Laplace transform, Existence theorem, Properties of Laplace Transform, Laplace transform of derivatives and integrals, Unit step function, Laplace transform of periodic function, Inverse Laplace transform,Convolution theorem. Application of Laplace Transform to solve ordinary differential equations and simultaneous differential equations.

Suggested Books:

Text Books:

1. B. V. Ramana, Higher Engineering Mathematics, Tata McGraw-Hill Publishing Company Ltd. 2. B. S. Grewal, Higher Engineering Mathematics, Khanna Publisher.

Reference Books:

1. R. K. Jain & S. R. K. Iyenger, Advanced Engineering Mathematics , Narosa Publishing - House, 2002.
2. Erwin Kreyszig, *Advanced Engineering Mathematics*, Wiley India Pvt. Ltd.
3. H. K. Dass, *Advanced Engineering Mathematics*, S. Chand & Company Ltd.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	3	-	-	-	-	-	-	-	2
CO2	3	2	-	3	2	-	-	-	-	-	-	2
CO3	3	2	-	2	-	-	-	-	-	-	-	2
CO4	3	3	2	3	-	-	-	-	-	-	-	3
CO5	3	3	3	3	2	-	-	-	-	-	-	3
AVG.	3	2.6	1.4	2.8	0.8	-	-	-	-	-	-	2.4

Course Name: Data Structure using C	**Credit Scheme**				**Evaluation Components**			
Course Code: CSC 114	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	50	50	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1. To introduce the fundamental concepts of data structures and algorithm design.
2. To develop the ability to represent, store, and manipulate data efficiently using appropriate structures.
3. To understand the principles of algorithm analysis and evaluate their time and space complexities.
4. To explore various searching, sorting, and hashing techniques and their practical implementations.
5. To apply suitable data structures and algorithms for solving real-world computational problems effectively.

Course Outcomes:

CO1: Develop the ability to read, write, and analyze the time and space complexity of algorithms.
CO2: Describe the properties, behavior, and implementation of fundamental data structures such as Stacks, Queues, Linked Lists, Trees, and Graphs.
CO3: Compare and evaluate various searching and sorting techniques in terms of efficiency, memory usage, and time complexity.
CO4: Design and implement suitable hash functions, analyze collision effects, and explore different hash table implementations.
CO5: Apply data structures and algorithmic techniques to solve real-world computational problems effectively.

Detailed Syllabus

Unit I **8 Hours**

Introduction: Data types, Abstraction, Abstract Data Type (ADT), Concept of data structure, Types of data structures, Operations on Data Structures, Introduction to Algorithms, Writing Pseudocodes, Algorithm analysis, Complexity of algorithms and Time space trade-off, Searching: Linear and Binary.

Unit II **8 Hours**

Arrays: Introduction to Array, Applications of Array, Operations on Arrays: Traverse, Insert, Delete etc. Stacks: Introduction to Stacks, Array representation of Stack, Operations on Stack: Push, Pop, etc. Applications of Stacks: Infix and Postfix Conversion, Evaluations of Infix and Postfix expressions. Queue: Introduction to Queue, Array representation and implementation of queues, Operations of Queue, Applications of Queue, Types of Queue: Circular Queue, Priority Queue, Double ended Queue. Operations on each type of Queue and their Applications.

Unit III **10 Hours**

Linked Lists: Introduction to Dynamic Memory Allocation, Representation and Implementation of Single, Double, and Circular Linked Lists, Operations on Linked List: Insert, Delete, Traverse etc. Applications of Linked List, Linked List representation of Stack and Queue. Trees: Basic Tree terminologies, Types of Trees: Binary Tree, Binary Search Tree (BST), AVL Tree, B-Tree, and Heap. Representation and Implementations of different types of trees, Tree Traversal algorithms, Operation on trees: Insert, Delete, etc., Applications of Trees.

Unit IV **8 Hours**

Graphs: Introduction to Graph and their Terminologies, Types of Graph, Representations of Graph, Graph traversal algorithms, Topological Sorting, Minimum Spanning Tree, Shortest Path Algorithms: Single Source Shortest Path like Bellman-Ford, Dijkstra and All Pair Shortest Path like Floyd-Warshall.

Unit V **8 Hours**

Sorting Algorithms and their Analysis: Selection Sort, Bubble sort, Insertion sort, Quick sort, Merge sort, Heap Sort. Performance Analysis and Comparison of all sorting techniques. Hashing: Hash Functions and its type, Hash Table construction, Collision Resolution, Universal Addressing, Open Hashing.

Suggested Readings

Text Book

1. Aaron M. Tenenbaum, YedidyahLangsam, Moshe J. Augenstein, Data Structures using C Pearson.1st Edition.2019
2. Schaum's outline series, Data structures with C, McGraw Hill Education; 1st edition (July 2017)

ReferenceBooks

1. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publication,2nd Edition. 2008.
2. Robert Kruse, Data Structures and Program Design in C PHI.2nd Edition.2006.
3. Willam J. Collins, Data Structure and the Standard Template library –2003, T.M.H.1st Edition.
4. Kyle Loudon, Mastering Algorithms with C, O'Reily Publication, 1st Edition

CO-PO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	2	-	-	-	2	2	-	2
CO2	3	3	2	2	2	-	-	-	2	2	-	2
CO3	3	3	3	2	3	-	-	-	2	2	-	2
CO4	3	3	3	2	3	-	-	-	2	2	-	2
CO5	3	3	3	2	3	-	-	-	2	2	-	2
AVG.	3	3	2.8	2	2.6	-	-	-	2	2	-	2

List of Practicals:

Write C Programs to illustrate the concept of the following:

1. Sorting Algorithms-Non-Recursive.
2. Sorting Algorithms-Recursive.
3. Searching Algorithms.
4. Implementation of Stack using Array.
5. Implementation of Queue using Array.
6. Implementation of Circular Queue using Array.
7. Implementation of Stack using Linked List.
8. Implementation of Queue using Linked List.
9. Implementation of Circular Queue using Linked List.
10. Implementation of Tree Structures, Binary Tree, Tree Traversal, Binary Search Tree, Insertion and Deletion in BST.
11. Graph Implementation, BFS, DFS, Minimum cost spanning tree, shortest path algorithm.

Course Outcomes

Course Name: Data Structures Using C Laboratory

After completing this course, the student will be able to:

CO1: Implement and analyze basic data structures such as arrays, stacks, and queues using C.
CO2: Develop C programs for linked data structures including singly, doubly, and circular linked lists.
CO3: Apply searching and sorting techniques to organize and retrieve data efficiently.
CO4: Implement non-linear data structures such as trees and graphs using appropriate algorithms.

CO5: Evaluate the time and space complexity of data structure operations and select suitable structures for given problems.

CO-PO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	2	-	-	-	2	2	-	2
CO2	3	3	2	2	2	-	-	-	2	2	-	2
CO3	3	3	3	2	3	-	-	-	2	2	-	2
CO4	3	3	3	2	3	-	-	-	2	2	-	2
CO5	3	3	3	2	3	-	-	-	2	2	-	2
AVG.	3	3	2.8	2	2.6	-	-	-	2	2	-	2

Course Name: Workshop / Manufacturing Practices	**Credit Scheme**				**Evaluation Components**			
Course Code: ESC 115	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	1	0	4	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To provide foundational knowledge of workshop safety, plant layout, and the use of different materials and tools in manufacturing practices.
2. To develop skills in carpentry including wood types, treatment, defects, joints, and basic exercises using hand tools and wood working machines.
3. To impart practical knowledge of fitting and foundry operations, including marking, drilling, tapping, filing, moulding, and basic casting procedures.
4. To introduce welding and sheet metal processes, including arc welding, gas welding, TIG/MIG/Spot welding, and fabrication of sheet metal components.
5. To familiarize students with machining operations, measuring tools, and CNC processes, including turning, threading, tapering, knurling, and profile cutting using AutoCAD software.

Course Outcomes: After completing this course the student will be able to

CO1: Demonstrate awareness of workshop safety, plant layout, and selection of materials and tools.
CO2: Perform basic carpentry operations, prepare joints, and operate wood working tools and machines.
CO3: Execute fitting and foundry exercises with correct dimensioning, marking, and basic casting techniques.
CO4: Apply welding and sheet metal techniques to fabricate joints, funnels, trays, and panels.
CO5: Operate machine tools, perform machining operations, and understand CNC processes and profile cutting.

Detailed Syllabus:

UNIT 1 **2 Hours**

Introduction of workshop manufacturing practice,1.To study the importance of safety in the work workshop,To study the plant shop layout, types of plant layout, quality of good plant shop layout, Introduction different types of material used in workshop tools and raw

material etc.

UNIT 2 **3 Hours**

Carpentry Shop: Types of wood and its treatment and defects, Procedure of exercises-marking, sawing, hand drilling etc,Study of tools operation and carpentry joints, Simple exercise using jack plain, Rasp file, Chiesel, Try square, Process of prepare T- lap joint, mortise and tenon joints, Simple exercise on wood working lathe.

UNIT 3 **4 Hours**

Fitting Bench Working Shop: 1. Study of tools and operations 2. Procedure of exercises preparation of U or V-Shape male female fitting, dimension tolerance as per fitting; Sawing, involving filling work. 3. Making the perfect male-female joint 4. Simple exercise involving Marking, Drilling, Tapping, Dyeing, filling, Job right angle Moulding and foundry Shop: Introduction of mould, types, dies core cavity, allowances, parting line, castings types,etc.

UNIT 4 **4 Hours**

Welding Shop : 1. Study of tools and operations of arc welding, electrodes. Welding machine working principal 2. Simple butt joint. 3. Lap joint. 4. Gas welding, oxy acetylene welding. Types of gas flame, TIG, MIG, SPOT Welding. Sheet metal shop : 1.Study of tools, material and operations .like. Marking ,shearing, bending ,blanking, etc. 2.Making funnel complete process with soldering. 3.Fabrication of Square Tray, electrical panel box etc.

UNIT 5 **4 Hours**

Machine Shop : Introduction of Lathe machine ,surface grinder ,milling machine ,shaper machine working principal, Study of tools and operations,Facing, Plane turning. Stepturning,Taperturning, Threading, Knurling, chamfering, parting-off, Boring measuring tools vernier caliper, micrometer, height gauge, dial gauge. CNC machine working principal. profile cutting process by Autocad software.

Suggested Books:

Textbooks

1. H.S. Bawa, *Workshop Manual*, Tata McGraw Hill, 2010.
2. K.C. John, *Workshop Technology (Manufacturing Processes)*, McGraw Hill, 2014.

Reference Books

1. P.K. Mishra, *A Course in Workshop Technology*, Vol. I & II, Dhanpat Rai & Sons. 2. R.K. Jain, *Production Technology*, Khanna Publishers, 2016.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2	-	-	-	2	2	-	2
CO2	3	3	3	2	2	-	-	-	2	2	-	2
CO3	3	3	3	2	3	-	-	-	2	2	-	2
CO4	3	3	3	2	3	-	-	-	2	2	-	2
CO5	3	3	3	2	3	-	-	-	2	2	-	2 2
AVG.	3	3	2.8	2	2.6	-	-	-	2	2	-	2

List of Practicals:

1. To study the importance of safety in the workshop.
2. To study the plant and shop layout.
3. Carpentry Shop
4. Fitting Bench Working Shop
5. Black Smithy Or Forging Shop
6. Welding Shop
7. Sheet metal shop
8. Machine Shop
9. CNC Machine shop.
10. Electrical and electronics shop.
11. Foundry Shop.
12. Paint Shop.
13. Plumbing Shop.
14. Machine Shop (The lathe machine Parts, Accessories, Diagram and Explanation)

Course Outcomes

Course Name: Workshop / Manufacturing Practices Laboratory

After completing this course, the student will be able to:

CO1: Identify various workshop tools, machines, and materials used in basic manufacturing processes.
CO2: Perform fundamental operations in carpentry, fitting, welding, smithy, and sheet-metal workshops by following safety norms.

CO3: Interpret simple engineering drawings and fabricate basic components as per given specifications.
CO4: Apply appropriate manufacturing practices and process selection for producing simple mechanical parts.
CO5: Demonstrate teamwork, discipline, and professional ethics while working in a workshop environment.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	3	2	-	-	-	-	-	-	2
CO2	3	3	2	3	-		-	-	-	-	-	2
CO3	3	2	3	-	-	-	-	-	-	-	-	2
CO4	3	3	-	-	-	2	3	-	-	-	-	2
CO5	2	1	1	3	-	2	2	-	-	-	-	2
AVG.	7	5.5	3	4.5	1	2	2.5	-	-	-	-	5

Course Name: Python Programming	**Credit Scheme**				**Evaluation Components**			
Course Code: CSC 116	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	1	0	4	50	50	50	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To introduce students to Python programming, its IDE, basic syntax, expressions, operators, and type conversions.
2. To develop problem-solving skills using conditional statements, loops, and nested structures in Python.
3. To impart knowledge of functions, strings, and core Python data structures including lists, tuples, sets, and dictionaries.
4. To enable students to understand file I/O, exception handling, modules, abstract data types, and object-oriented programming concepts like classes and inheritance.
5. To provide understanding of recursion, iterators, and algorithms for searching and sorting using Python.

Course Outcomes: After completing this course the student will be able to

CO1: Write and execute basic Python programs using expressions, operators, and type conversions.
CO2: Apply conditional statements and loops to solve programming problems efficiently.
CO3: Implement functions, manipulate strings, and use Python data structures for real-world applications.
CO4: Handle files, exceptions, and modules; design classes and implement object-oriented programming concepts in Python.
CO5: Apply recursion, iterators, and implement searching and sorting algorithms using Python.

Detailed Syllabus:

UNIT 1 **7 Hours**

Introduction: The Programming Cycle for Python, Python IDE, Interacting with Python Programs, Elements of Python, Type Conversion. Basics: Expressions, Assignment Statement, Arithmetic Operators, Operator Precedence, Boolean Expression.

UNIT 2 **8 Hours**

Conditionals: Conditional statement in Python (if-else statement, its working and execution), Nested-if statement and Else if statement in Python, Expression Evaluation & Float Representation. Loops: Purpose and working of loops, while loop including its working, For Loop, Nested Loops, Break and Continue.

UNIT 3 **10 Hours**

Function: Parts of A Function, Execution of A Function, Keyword and Default Arguments, Scope Rules Strings: Length of the string and perform Concatenation and Repeat operations in it. Indexing and Slicing of Strings. Python Data Structure: Tuples, Unpacking Sequences, Lists, Mutable Sequences, List Comprehension, Sets, Dictionaries Higher Order Functions: Treat functions as first-class Objects, Lambda Expressions. File I/O: File input and output operations in Python Programming

UNIT 4 **12 Hours**

Exceptions and Assertions Modules: Introduction, Importing Modules using PIP,Numpy Abstract Data Types: Abstract data types and ADT interface in Python Programming. Classes: Class definition and other operations in the classes, Special Methods (such as _in it_, _str_, comparison methods and Arithmetic methods etc.), Class Example, Inheritance, Inheritance and OOP.

UNIT 5 **8 Hours**

Iterators & Recursion: Recursive Fibonacci, Tower Of Hanoi Search: Simple Search and Estimating Search Time, Binary Search and Estimating Binary Search Time. Sorting & Merging: Selection Sort, Merge List, Merge Sort, Higher Order Sort.

Suggested Books:

Text Books:

1. R. Nageswara Rao, *Core Python Programming*, Dreamtech Press.
2. Reema Thareja, *Python Programming*, Oxford University Press, 2017.

Reference Books:

1. Allen B. Downey, *Think Python*, 2nd Edition, O'Reilly Media.
2. Swaroop C H, *A Byte of Python*, 3rd Edition, Python Software Foundation

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2	-	-	-	2	2	-	2
CO2	3	3	3	2	2	-	-	-	2	2	-	2
CO3	3	3	3	2	3	-	-	-	2	2	-	2
CO4	3	3	3	2	3	-	-	-	2	2	-	2
CO5	3	3	3	2	3	-	-	-	2	2	-	2 2
AVG.	3	3	2.8	2	2.6	-	-	-	2	2	-	2

List of Practicals:

1. Write a program to perform arithmetic operations, type conversions, and use basic input/output statements.
2. Write a program using if, if-else, nested if, and elif for decision-making scenarios.
3. Implement while loops, for loops, nested loops, and use break/continue statements in programs. 4. Define and call functions with parameters, default and keyword arguments, practice recursion. 5. Write a program for string concatenation, slicing, indexing, and string operations.
6. Write a program to create, access, modify, and perform operations on lists and tuples. 7. Write a program for adding, deleting, and accessing elements in dictionaries and sets. 8. Write a program to generate lists using list comprehension with conditions.
9. Read from and write to text files, count words/lines, and copy contents between files.
10. Exception Handling:Write a program demonstrating try, except, finally, and raising exceptions.
11. Modules & Packages: Use Python standard libraries and custom modules in programs.
12. Object-Oriented Programming – Create classes, objects, inheritance, and implement methods including special methods like __init__.
13. Recursion & Iterators: Write a program for recursive problems like Fibonacci series, Tower of Hanoi, and implementing iterators.
14. Write a program to implement linear search and binary search.
15. Write a program to implement selection sort
16.Write a program to implement selection sort merge sort

Course Outcomes

Course Name: Python Programming Laboratory

After completing this course, the student will be able to:

CO1: Write and execute Python programs using basic syntax, data types, and control structures.
CO2: Develop Python programs using functions, modules, and built-in data structures such as lists, tuples, sets, and dictionaries.
CO3: Implement file handling and exception handling techniques for robust program execution.
CO4: Apply object-oriented programming concepts such as classes, objects, inheritance, and polymorphism in Python.
CO5: Solve real-world and engineering problems using Python by following good programming and debugging practices.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	-	3	2	-	2	-	-	-	-	2
CO2	2	3	2	3	-		-	-	-	-	-	2
CO3	2	2	3	-	-	-	-	-	-	-	-	2
CO4	3	2	-	-	-	2	3	-	-	-	-	2
CO5	2	1	1	3	3	2	2	-	-	-	-	2
AVG.	6	4.5	3	4.5	2.5	2	3.5	-	-	-	-	5

Course Name: Environmental Science	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC 117	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	1	1	0	0	50	0	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1. Provide foundational knowledge of environmental science and engineering principles, highlighting their differences.
2. Enable understanding of natural resources, biodiversity, and conservation strategies.
3. Explain causes, effects, and control measures of various types of environmental pollution.
4. Introduce environmental policies, legislation, and sustainable development guidelines.
5. Promote awareness of human impact on the environment and encourage sustainable practices and eco-friendly technologies.

Course Outcomes: After completing this course the student will be able to

CO1: Understand the differences between environmental science and environmental engineering and basic ecological concepts.
CO2: Analyze natural resources, biodiversity, and conservation strategies.
CO3: Identify sources, impacts, and control methods of air, water, soil, noise, and radioactive pollution.
CO4: Explain national and international environmental policies, legislation, and assessment methods.
CO5: Apply sustainable practices, renewable energy concepts, and eco-friendly technologies to reduce human environmental impact.

Detailed Syllabus:

Unit I **8 Hours**

Introduction to Environmental Science: Environmental Science, Environmental Engineering and their differences, Scope and importance of environmental studies, Basic ecological concepts, Ecosystems, Biomes, Environmental ethics.

Unit II **8 Hours**

Natural Resources and Biodiversity: Classification of natural resources, Water, Soil, Forests, Minerals, Energy resources, Renewable and non-renewable resources, Biodiversity, Threats to biodiversity, Conservation strategies, Ecological balance.

Unit III **8 Hours**

Environmental Pollution: Air pollution, Water pollution, Soil pollution, Noise pollution, Radioactive pollution, Causes and effects of pollution, Pollution monitoring techniques, Environmental standards, Impact on human heealth and ecosystems.

Unit IV **8 Hours**

Environmental Policies and Legislation: National and international environmental laws, Treaties and acts, Guidelines for sustainable development, Environmental impact assessment, Environmental audits, Compliance mechanisms.

Unit V **8 Hours**

Sustainable Practices and Human Impact: Human interactions with environment, Climate change, Global warming, Renewable energy, Waste management, Eco-friendly technologies, Sustainable urban and rural planning.

Suggested Books:

Text Books:

1. Erach Bharucha, *Textbook of Environmental Studies*, Universities Press, 2020. 2. Benny Joseph, *Environmental Science and Engineering*, Tata McGraw-Hill, 2019. 3. R. Rajagopalan, *Environmental Studies*, Oxford University Press, 2019.

Reference Books:

1. *Environmental Science* by Daniel D. Chiras, Jones & Bartlett Learning, 2018. 2. *Introduction to Environmental Science* by Cunningham & Cunningham, McGraw-Hill Education, 2017.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2	-	-	-	2	2	-	2
CO2	3	3	3	2	2	-	-	-	2	2	-	2
CO3	3	3	3	2	3	-	-	-	2	2	-	2
CO4	3	3	3	2	3	-	-	-	2	2	-	2
CO5	3	3	3	2	3	-	-	-	2	2	-	2 2
AVG.	3	3	2.8	2	2.6	-	-	-	2	2	-	2

Course Name: Communication skills II	**Credit Scheme**				**Evaluation Components**			
Course Code: CS 002	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	1	0	0	50	0	50	100
Prerequisite Course and code (if any):								

Course objectives:

1. Enhance professional communication competence in English for academic and workplace contexts.
2. Develop effective listening and speaking skills for discussions, presentations, and interviews.
3. Improve reading and writing abilities with emphasis on technical and professional documents.
4. Build confidence in group communication through group discussions, role plays, and teamwork activities.
5. Strengthen employability skills such as interview performance, resume writing, and workplace etiquette.

Course Outcomes

CO1: Communicate effectively in academic and professional environments

CO2: Write clear technical and professional documents

CO3: Participate confidently in group discussions and interviews

CO4: Deliver effective presentations

CO5: Demonstrate improved interpersonal and workplace communication skills

Syllabus

Unit 1 **8 Hours**

Importance of effective listening, Types of listening: active, passive, selective, Barriers to listening and how to overcome them, Pronunciation and accent neutralization, Stress, intonation, and rhythm, Conversational skills and role plays

Unit II **8 Hours**

Reading for information and inference, Skimming and scanning techniques, Understanding technical and non-technical texts, Vocabulary building, Use of dictionaries and reference materials

Unit III **8 Hours**

Sentence structure and paragraph development, Technical writing basics, Email writing (formal and informal), Report writing, Resume and cover letter writing, Common grammatical errors

Unit IV **8 Hours**

Group discussion: concepts and practice, Public speaking and presentations, Seminar and conference communication, Telephone and video-conference etiquette, Workplace communication and ethics

Unit V **8 Hours**

Interview skills (technical & HR interviews), Personality development, Teamwork and leadership skills, Time management, Emotional intelligence, Cross-cultural communication

Suggested books:

Text Books:

1. Technical Communication – Meenakshi Raman & Sangeeta Sharma

2. Business Communication – K.K. Sinha

Reference books:

3.Communication Skills – Sanjay Kumar & Pushp Lata

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	2	2	-	-	-	2	2	-	2
CO2	3	3	2	2	2	-	2	-	2	2	-	2
CO3 **CO4**	3 3	3 3	2 3	1 1	3 3	- -	2 3	- -	2 2	2 2	- -	2 2
CO5	3	3	2	2	3	-	2	-	2	2	-	2 2
AVG.	3	3	5	8	6.5	-	4.5	-	2	2	-	2

Semester: 3rd								
Course Name: Design & analysis of algorithm	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE 201	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	-	2	50	50	50	150
Prerequisite Course and code (if any):								

Course Objectives

1. Understand the principles of algorithm design and analysis.
2. Analyze the time and space complexity of algorithms.
3. Apply different algorithmic paradigms to solve real-world problems.
4. Develop efficient algorithms using appropriate data structures.
5. Enhance problem-solving and analytical skills essential for software engineering and research.

Course Outcomes: After completing this course the student will be able to

CO1: Analyze algorithms using asymptotic notations.

CO2: Design algorithms using divide and conquer techniques.

CO3: Apply greedy strategies to optimization problems.

CO4: Solve problems using dynamic programming techniques.

CO5: Design and analyze graph algorithms. Understand NP-completeness and computational complexity

Detailed Syllabus:

UNIT I: Introduction & Algorithm Analysis **7 Hours**

Introduction to algorithms, definition, and characteristics of algorithms. Algorithm specification and

performance analysis. Analysis of time and space complexity. Asymptotic notations: Big-O (O), Big-Ω (Omega), and Big-Θ (Theta). Best-case, average-case, and worst-case analysis of algorithms. Recurrence relations and their solutions using substitution method, recursion tree method, and Master's Theorem..

UNIT II: Divide and Conquer Algorithms (8–10Hours)

General methodology of divide and conquer approach. Algorithms based on divide and conquer technique including Binary Search, Merge Sort, and Quick Sort. Strassen's Matrix Multiplication algorithm. Time complexity analysis of divide and conquer algorithms. Advantages, limitations, and applications of divide and conquer techniques.

UNIT III: Greedy Algorithms (8–10 Hours)

Introduction to greedy method and its characteristics. Application of greedy strategies to optimization problems including Activity Selection Problem, Fractional Knapsack Problem, and Job Sequencing with Deadlines. Minimum Cost Spanning Tree algorithms using Prim's and Kruskal's algorithms. Optimal Merge Pattern and Huffman Coding. Analysis and correctness of greedy algorithms.

UNIT IV: Dynamic Programming (8–10 Hours)

Concept of dynamic programming and comparison with greedy method. Principles of optimal substructure and overlapping subproblems. Dynamic programming solutions to Matrix Chain Multiplication, Longest Common Subsequence (LCS), and 0/1 Knapsack Problem. All-Pairs Shortest Path problem using Floyd–Warshall Algorithm. Reliability Design Problem.

UNIT V: Graph Algorithms & **NP-Completeness** (8–10 Hours)

Graph representations and basic terminology. Graph traversal techniques including Breadth First Search (BFS) and Depth First Search (DFS). Shortest path algorithms such as Dijkstra's Algorithm and Bellman-Ford Algorithm. Topological sorting of directed acyclic graphs. Identification of Strongly Connected Component, Introduction to computational complexity and complexity classes P and NP. NP-Complete and NP-Hard problems. Overview of Cook's Theorem. Classical NP-Complete problems including Traveling Salesman Problem and Boolean Satisfiability Problem (SAT). Introduction to approximation algorithms. Overview of backtracking and branch and bound techniques.

Suggested Books:

Text Books

1. **T. H. Cormen et al.**, *Introduction to Algorithms*, MIT Press
2. **Ellis Horowitz, Sartaj Sahni**, *Fundamentals of Computer Algorithms*, Galgotia

Reference Books

1. **Aho, Hopcroft, Ullman**, *The Design and Analysis of Computer Algorithms*
2. **S. Dasgupta, C. Papadimitriou**, *Algorithms*, McGraw-Hill
3. **Anany Levitin**, *Introduction to the Design & Analysis of Algorithms*

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	-	-	-	-	1	-	2
CO2	3	3	3	1	1	2	1	-	-	1	1	2
CO3	3	3	3	3	2	-	2	-	1	1	2	2
CO4	3	3	3	2	2	-	-	2	1	1	1	2
CO5	3	3	3	2	3	3	-	1	2	1	1	3
Average	3.0	3.0	2.8	2.2	2.0	–	–	–	1.0	1.0	1.0	2.2

List of Practical

1. Study and analysis of algorithm performance (time & space complexity)
2. Implementation of Binary Search algorithm
3. Implementation of Merge Sort algorithm
4. Implementation of Quick Sort algorithm
5. Implementation of Strassen’s Matrix Multiplication

6. Activity Selection Problem using Greedy method

7, Fractional Knapsack Problem using Greedy approach

8. Job Sequencing with Deadlines

9. Minimum Spanning Tree using Prim's Algorithm

10. Minimum Spanning Tree using Kruskal's Algorithm

11. Matrix Chain Multiplication using Dynamic Programming

12. Longest Common Subsequence (LCS) using Dynamic Programming

13. 0/1 Knapsack Problem using Dynamic Programming

14. All-Pairs Shortest Path using Floyd–Warshall Algorithm

15. Graph Traversal using BFS and DFS

Course Outcomes: Design and Analysis of Algorithms Laboratory

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

CO1: Implement fundamental algorithms for sorting, searching, and basic data processing using a programming language.

CO2: Analyze the time and space complexity of implemented algorithms through experimental evaluation.

CO3: Design and implement algorithms using paradigms such as divide and conquer, greedy method, dynamic programming, and backtracking.

CO4: Implement graph algorithms including traversal, shortest path, and minimum spanning tree techniques.

CO5: Compare the performance of multiple algorithms solving the same problem and justify the

selection of an efficient solution.

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	-	-	-	-	1	-	2
CO2	3	1	2	1	1	2	1	-	-	1	1	2
CO3	3	3	3	3	2	-	2	-	1	1	2	2
CO4	3	2	3	2	1	-	-	2	1	1	1	2
CO5	3	3	3	2	2	3	-	1	2	1	1	3
Average	5	6	6.5	5	3.5	2.5	1.5	1.5	2	2.5	2.5	5.5

Object Oriented Programming with java	Credit Scheme	Evaluation Components						
Course Code: CSE 202	Total Credits	L	T	P	I	P	E	Total
	4	3	0	2	50	50	50	150
Prerequisite Course and code (if any):								

Course Objectives

1. To understand the fundamentals of Object-Oriented Programming (OOP) concepts.
2. To develop problem-solving skills using object-oriented approach.
3. To design and implement programs using Java and Python.
4. To understand exception handling, file handling, and multithreading concepts.
5. To build reusable, maintainable, and scalable software systems using OOP principles

Course Outcomes:

After completing this course the student will be able to:

CO1: Understand and apply core object-oriented concepts such as classes, objects, inheritance, and polymorphism.

CO2: Design programs using encapsulation, abstraction, and modular programming.

CO3: Implement exception handling and file handling mechanisms in Java/Python.

CO4: Develop applications using advanced OOP features such as interfaces, packages, and multithreading.

CO5: Apply object-oriented principles to solve real-world problems and develop efficient applications

Unit I **8 Hour**

Programming paradigms: Procedural vs Object-Oriented, Basic concepts of OOP: Object, Class, Abstraction, Encapsulation, Inheritance, Polymorphism, Features and benefits of OOP,Java overview and history, Java Virtual Machine (JVM), JRE, JDK, Structure of a Java program, Compilation and execution process

Unit II **8 Hour**

Tokens, keywords, identifiers, literals, Data types and variables, Operators and express

Control statements:, Selection (if, if–else, switch), Iteration (for, while, do-while),Type conversion casting,Command line arguments

Unit III **8 Hour**

Exception handling concepts: types of errors and exceptions. Exception hierarchy in Java/Python. Try-catch-finally blocks. User-defined exceptions. File handling: file streams, reading and writing text and binary files. Serialization and deserialization (Java). File operations using Python file objects.

Unit IV **8 Hour**

Abstract classes and interfaces. Difference between abstract class and interface. Implementation of interfaces. Multiple inheritance using interfaces. Multithreading concepts: thread lifecycle, creating threads, thread synchronization, inter-thread communication. Thread handling in Java and Python.

Unit V **8 Hour**

Collections framework (Java) and built-in data structures (Python). Generics (Java) and dynamic typing (Python). Lambda expressions and functional programming concepts. GUI programming basics (Java Swing / Python Tkinter). Introduction to design principles and simple design patterns. Case studies and real-world application development.

Suggested Books

Text Books

1. E. Balagurusamy, *Programming with Java*, McGraw Hill
2. Guido van Rossum, *Python Programming*, Pearson

Reference Books

1. Herbert Schildt, *Java: The Complete Reference*, McGraw Hill

2. Timothy Budd, *An Introduction to Object-Oriented Programming*, Pearson

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	-	-	-	-	1	-	2
CO2	3	3	3	1	1	2	1	-	-	1	1	2
CO3	3	3	3	3	2	-	2	-	1	1	2	2
CO4	3	3	3	2	2	-	-	2	1	1	1	2
CO5	3	3	3	2	3	3	-	1	2	1	1	3
Average	3.0	3.0	2.8	2.2	2.0	–	–	–	1.0	1.0	1.0	2.2

List of Practical

1. Study of OOP concepts and program structure
2. Program to create classes and objects
3. Implementation of inheritance and method overriding
4. Program demonstrating polymorphism

5. Implementation of abstract classes and interfaces

6. Program for exception handling

7. User-defined exception program

8. File handling – read and write operations

9. Program using collections framework / data structures

10. Multithreading program

11. Thread synchronization program

12. GUI application development

13. Case study-based application

14. Program using lambda expressions

15. Mini project using Java/Python

Course Outcomes: Object Oriented Programming with Java Laboratory

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

CO1: Write and execute Java programs using basic language constructs, classes, and objects.

CO2: Implement object-oriented principles such as encapsulation, inheritance, and polymorphism in Java programs.

CO3: Develop and test programs using interfaces, abstract classes, and packages.

CO4: Handle runtime errors using exception handling and perform file input/output operations.

CO5: Develop multithreaded programs and use Java Collections Framework for efficient data management.

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	2	1	-	-	-	-	1	-	2
CO2	3	3	3	1	1	2	1	-	-	1	1	2
CO3	3	2	3	3	2	-	2	-	1	1	2	2
CO4	2	3	1	2	2	-	-	2	1	1	1	2
CO5	3	3	1	2	3	3	-	1	2	1	1	3
Average	3.0	3.0	2.8	2.2	2.0	–	–	–	1.0	1.0	1.0	2.2

Course Name: Relational Database Management System	Credit Scheme				Evaluation Components			
Course Code: CSEC211	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1. To introduce the fundamental concepts and architecture of database management systems.
2. To understand advanced topics such as distributed databases, data fragmentation, replication, and query processing.
3. To study transaction management, concurrency control, and database recovery mechanisms.
4. To develop skills in database administration, including user management, security, backup, and performance tuning.
5. To apply database design principles such as ER modeling, normalization, and schema refinement for building efficient databases.

Course Outcomes:

CO1: Understand database models, architecture, and fundamental concepts.
CO2: Develop and manage relational databases using SQL.
CO3: Apply normalization techniques to improve database efficiency.
CO4: Implement indexing, query optimization, and transaction management for performance.
CO5: Analyze database security, recovery mechanisms, and concurrency control strategies

Detailed Syllabus:

Unit I **8 Hours**

Introduction: An overview of database management system, Database System Vs File System, Database system concepts and architecture, data models schema and instances, data independence and database language and interfaces, Data definitions language, DMI, Overall Database structure. Data modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping

constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of ER diagrams to tables, extended ER model, relationships of higher degree.

Unit II **10 Hours**

Relational Data Model and Language: Relational data model concepts, integrity constraints: entity integrity, referential integrity, Keys constraints, Domain constraints, relational algebra, relational calculus, tuple and domain calculus.

Unit III **8 Hours**

Introduction to SQL: Characteristics of SQL, Advantages of SQL, SQL data types and literals, Types of SQL commands, SQL operators and their procedure, Tables, views and indexes Queries and sub queries, Aggregate functions, Insert, update and delete operations, Joints, Unions, Intersection, Minus, Cursors in SQL. PL/SQL, Triggers and clusters.

Unit IV **8 Hours**

Database Design & Normalization: Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependencies, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design.

Unit V **8 Hours**

Transaction Processing Concepts: Transaction system, Testing of seralizability, Seralizability of schedules, conflict and view serializable schedule, recoverability, Recovery form transaction failures, deadlock handling. Concurrency Control.

Suggestive Readings:

Text Books:

1. Introduction to Database Systems by Bipin Desai
2. Database System Concepts by Henry F.Korth

Reference Books:

1. Introduction to SQL by Ivan Bayross
2. Database Management Systems by SCHAUM's Outlines
3. Database Systems by Navathe.

CO-PO Mapping:

PO □ CO □	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	3	3	3	2	3	-	-	-	2	-	-	2
CO3	3	3	2	2	2	-	-	-	-	-	-	2
CO4	3	3	3	3	3	-	-	-	-	-	-	2
CO5	2	3	2	2	-	-	-	2	-	-	-	2
Avg	2.8	2.8	2.4	2.1	2.6	0	0	2	2	0	0	2

List of Practicals:

1) Create a table “Student” contains 5 attributes:-
   a. Rollno [Primary Key]
   b. SName
   c. Course
   d. Age
   e. City
2) a. Insert 5 records in the Student’s table:
   b. Delete the records whose roll no. is equal to 101 or 102
   c. Retrieve only those records whose course is equal to “B.Tech”
   d. Retrieve only those records whose age is greater than 18
3) Create a table “Employee” contains 5 attributes:-
   a. Empid [Primary Key]
   b. Ename
   c. Salary
   d. Age
   e. Department
4) a. Insert 5 records in the above table
   b. Update the salary of all the employees 10%
   c. Retrieve only those records who has maximum or minimum salary
5) WAQ to perform all the Aggregate Functions used in SQL
6) What is Primary Key and Foreign Key with the help of an example.
7) What is Join ? Explain the types of Join with the help of example
8) What is a function? How to create function in PL/SQL with example
9) What is the procedure? How to create procedure in PL/SQL with example
10) What is the trigger? How to create trigger in PL/SQL with example
11) What is the package? How to create a package in PL/SQL with example
12) Perform Union, Intersection and Minus in 2 tables with examples.

13) What is a cursor? Explain the types of cursor? How to create explicit cursor in

PL/SQL
14) How to create an implicit cursor in PL/SQL.

Course Outcomes: Relational Database Management System Laboratory

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

CO1: Create and manage relational database schemas using SQL Data Definition Language (DDL) commands.

CO2: Perform data insertion, retrieval, update, and deletion using SQL DML and query operations.

CO3: Apply integrity constraints, keys, joins, views, and aggregate functions to ensure data consistency and accuracy.

CO4: Design efficient databases by applying normalization techniques and schema refinement.

CO5: Develop database programs using PL/SQL constructs such as procedures, functions, packages, cursors, and triggers.

CO-PO Mapping:

PO □ CO □	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	–	1	–	–	–	–	–	–	1
CO2	3	3	3	–	2	–	–	–	–	–	–	1
CO3	2	3	3	2	1	–	–	–	–	–	–	1
CO4	2	3	3	2	2	–	–	–	–	–	–	1
CO5	2	3	2	2	1	–	–	–	–	–	–	2
Avg	2.4	2.8	2.7	2	1.4	0	0	0	0	0	0	1.2

Course Name: CBNST	**Credit Scheme**				**Evaluation Components**			
Course Code: BSC204	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	2	0	90	0	60	150
Prerequisite Course and code (if any):								

Course Objectives:

1.With the current deployment of computer technology and tools.

2. It is very important to develop efficient algorithms for solving problems in science, engineering, technology, insurance & banking.

3. The objective of this course is to enable students to obtain an intuitive and working understanding of numerical methods for the basic problems of numerical analysis and gain experience in the implementation of numerical methods using a computer.

4. They would also gain an appreciation of the concept of error in these methods and the need to analyze and predict it.

5. Prerequisites: Basic knowledge of functions, logarithmic, trigonometric and exponential functions, graph of a function, polynomials, and roots of a polynomial, differentiation and integration.

Course Outcomes (COs)

Based on your syllabus, the Course Outcomes can be framed as:

CO1: Apply floating point arithmetic concepts and iterative numerical methods to find roots of equations.
CO2: Solve systems of linear equations and interpolation problems using direct and iterative numerical techniques.
CO3: Use advanced interpolation, approximation techniques, and polynomial approximations for unequal data.

CO4: Apply numerical differentiation, integration, and methods for solving ordinary differential equations with error analysis.
CO5: Apply curve fitting, regression analysis, and spline methods to real-world engineering and statistical data.

Unit-I Floating point Arithmetic: Representation of floating point numbers, Operations, Normalization, Pitfalls of floating point representation, Errors in numerical computation Iterative Methods: Zeros of a single transcendental equation and zeros of polynomial using Bisection Method, Iteration Method, Regula-Falsi method, Newton Raphson method, Secant method, Rate of convergence of iterative methods.

Unit-II Simultaneous Linear Equations: Solutions of system of Linear equations, Gauss Elimination direct method and pivoting, Ill Conditioned system of equations, Refinement of solution. Gauss Seidal iterative method, Rate of Convergence Interpolation and approximation: Finite Differences, Difference tables,Polynomial Interpolation: Newton's forward and backward formula.

Unit-III Central Difference Formulae: Gauss forward and backward formula, Stirling's, Bessel's, Everett's formula. Interpolation with unequal intervals: Langrange's Interpolation, Newton Divided difference formula, Hermite's Interpolation,Approximation of function by Taylor's series and Chebyshev polynomial

Unit-IV Numerical Differentiation and Integration: Introduction, Numerical Differentiation, Numerical Integration, Trapezoidal rule, Simpson's rules, Boole's Rule, Weddle's Rule Euler-Maclaurin Formula Solution of differential equations: Picard's Method, Euler's Method, Taylor's Method, Runge-Kutta methods, Predictor-corrector method, Automatic error monitoring, stability of solution.

Unit-V Curve fitting, Cubic Spline and Approximation: Method of least squares, fitting of straight lines, polynomials, exponential curves etc Frequency Chart: Different frequency chart like Histogram, Frequency curve, Pi-chart. Regression analysis: Linear and Non-linear regression, multiple regressions

References: 1. Rajaraman V., "Computer Oriented Numerical Methods", PHI

2. Gerald & Wheatley, “Applied Numerical Analyses”, AW

3. Jain, Iyengar and Jain, “Numerical Methods for Scientific and Engineering Computations”, New Age Int.

4. Grewal B. S., “Numerical methods in Engineering and Science”, Khanna Publishers, Delhi

5. T. Veerarajan, T Ramachandran, “Theory and Problems in Numerical”.

CO-PO Matrix

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	–	–	3	–	–	–	–	–	–	1
CO2	3	3	–	1	3	–	–	–	–	–	–	1
CO3	3	2	–	1	3	–	–	–	–	–	–	1
CO4	3	3	–	2	3	–	–	–	–	–	–	2
CO5	2	3	2	2	3	–	–	–	–	–	–	2
Avg	1.8	1.6	2	1.5	3	0	0	0	0	0	0	1.4

Course Name: ESSENCE OF INDIAN TRADITIONAL KNOWLEDGE	**Credit Scheme**				**Evaluation Components**			
Course Code: ASC206	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	NC	NC	0	0	NC	0	0	NC
Prerequisite Course and code (if any):								

COURSE OBJECTIVES: The objectives of this course are to

1. To facilitate the students with the concepts of Indian traditional knowledge and to make them understand the Importance of roots of knowledge system.

2. To make the students understand the traditional knowledge and analyses it and apply it to their day to day life.

3. To make the students know the need and importance of protecting traditional knowledge.

4. To make the students understand the concepts of Intellectual property to protect the traditional knowledge.

5. This course is also concentrating on various acts in protecting the environment and knowledge management impact on various sectors in the economic development of the country.

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

CO1. Understand the concept of Traditional knowledge and its importance.

CO 2. Know the need and importance of protecting traditional knowledge.

CO3. Know the various enactments related to the protection of traditional knowledge.

CO 4. Understand the concepts of Intellectual property to protect the traditional knowledge.

CO5. Know the contribution of scientists of different areas.

Unit – 1 Introduction to Traditional and Culture Knowledge Define culture, traditional, civilization and heritage knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK). Indigenous traditional knowledge Vs western traditional knowledge vis-à-vis formal knowledge. the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK). Indigenous traditional knowledge Vs western traditional knowledge vis-à-vis formal knowledge.

Unit-2 Protection of Traditional Knowledge Protection of traditional knowledge: The need for protecting traditional knowledge Significance of traditional knowledge Protection, value of traditional knowledge in global economy, Role of Government to harness traditional knowledge. Unit – 3 Traditional Knowledge and Intellectual Property Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, Global legal forums for increasing protection of Indian Traditional Knowledge.

Unit – 4 Traditional Knowledge in Different Sectors Traditional knowledge in engineering, biotechnology and agriculture, traditional medicine system, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of traditional knowledge.

Unit – 5 Education System in India Education in ancient, medieval and modern India, aims of education, subjects, languages, Science and Scientists of Ancient India, Scientists of Medieval

India, Scientists of Modern India. The role Gurukulas in Education System, Value based Education.

Text/Reference Books: 1. Traditional Knowledge System in India by Amit Jha Atlantic publishers,

2002.

2. "Knowledge Traditions and Practices of India" Kapil Kapoor1, Michel Danino2.

3. Traditional Knowledge System in India, by Amit Jha, 2009.

4. Satya Prakash, “Founders of Sciences in Ancient India”, Vijay Kumar Publisher, 1989

5. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and Vipin Kumar Singh Pratibha Prakashan 2012.

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	–	–	–	–	3	2	2	–	1	–	2
CO2	–	2	–	–	–	3	3	3	–	1	–	2
CO3	–	2	–	1	–	3	2	3	–	1	–	2
CO4	1	2	2	1	–	3	2	3	–	1	1	2
CO5	2	–	–	–	–	2	–	1	–	2	–	3
Avg	1.6	2	2	1	0	1.8	2.2	2.4	0	1.2	1	2.2

Course Name: Soft Skill Enhancement I	**Credit Scheme**				**Evaluation Components**			
Course Code: SSE 001	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	2	0	0	2	0	0	100
Prerequisite Course and code (if any):								

Course Objectives

1. To enhance students' communication skills for academic and professional environments.
2. To develop interpersonal skills, team work abilities, and leadership qualities.
3. To improve presentation skills, public speaking, and confidence building.
4. To cultivate problem-solving, critical thinking, and decision-making abilities.
5. To prepare students for placement processes and professional career challenges.

Course Outcomes (COs)

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

- **CO1:** Demonstrate effective verbal and non-verbal communication skills.
  CO2: Apply interpersonal skills and emotional intelligence in professional interactions.

- **CO3:** Work effectively in teams and demonstrate leadership qualities.
- **CO4:** Deliver structured presentations and participate confidently in discussions.
- **CO5:** Apply soft skills in interviews, group discussions, and workplace scenarios.

Detailed Syllabus

Unit – I: Communication Skills

Importance of communication, process of communication, barriers to communication, verbal and non-verbal communication, listening skills, reading and writing skills, professional etiquette in communication.

Unit – II: Interpersonal Skills & Emotional Intelligence

Self-awareness, self-management, motivation, empathy, emotional intelligence, attitude, personality development, body language, time management, stress management.

Unit – III: Team Work and Leadership

Team dynamics, roles in a team, leadership styles, conflict management, problem-solving techniques, decision-making skills, ethics in professional life.

Unit – IV: Presentation and Public Speaking Skills

Presentation planning, structuring presentations, use of visual aids, public speaking techniques, overcoming stage fear, group discussions, debates.

Unit – V: Career Skills and Professional Readiness

Resume writing, cover letter, interview skills (technical & HR), mock interviews, aptitude basics, workplace ethics, adaptability and lifelong learning.

References

1. Bhatnagar Nitin & Mamta Bhatnagar, *Effective Communication and Soft Skills*, Pearson.
2. Barun K. Mitra, *Personality Development and Soft Skills*, Oxford University Press.

3. Dale Carnegie, *How to Win Friends and Influence People*, Simon & Schuster.

4. Meenakshi Raman & Sangeeta Sharma, *Technical Communication*, Oxford University Press.

5. Shiv Khera, *You Can Win*, Bloomsbury.

CO-PO Matrices

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	–	–	–	–	–	2	–	2	–	3	–	2
CO2	–	2	–	–	–	3	–	3	–	2	–	2
CO3	–	2	2	–	–	3	–	2	3	2	–	2
CO4	–	–	–	–	–	2	–	2	–	3	–	2
CO5	–	2	–	–	–	3	–	3	–	3	1	3
Avg	0	2	2	0	0	2.6	0	2.4	3	2.6	1	2.2

Course Name:Engineering Innovation-I (conceive)	**Credit Scheme**				**Evaluation Components**			
Course Code: E1001	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	1	1	0	0	1	0	0	100
Prerequisite Course and code (if any):								

Course Objectives

1. To introduce students to the fundamentals of engineering innovation and creative thinking.
2. To develop problem identification and solution design skills using innovative approaches.
3. To promote design thinking, entrepreneurship, and multidisciplinary collaboration.
4. To encourage application of modern tools and technologies in developing innovative solutions.
5. To foster ethical, sustainable, and socially responsible innovation.

Course Outcomes (COs)

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

- **CO1:** Explain the concepts and importance of innovation in engineering and society.

- **CO2:** Identify real-world engineering problems and analyze them using creative thinking tools.
- **CO3:** Apply design thinking methodologies to develop innovative engineering solutions.
- **CO4:** Use modern engineering tools and techniques to model and validate innovative ideas.
- **CO5:** Demonstrate teamwork, communication, and ethical responsibility in innovation projects.

Detailed Syllabus:

Unit – I: Introduction to Engineering Innovation

Innovation vs invention, need for innovation, types of innovation, innovation ecosystem, role of engineers in innovation, case studies of successful engineering innovations.

Unit – II: Creativity and Design Thinking

Creativity techniques (brainstorming, mind mapping, SCAMPER), design thinking process, empathy mapping, problem definition, ideation, prototyping and testing.

Unit – III: Problem Identification and Concept Development

Identifying societal and industrial problems, user-centered design, feasibility analysis, concept generation, selection methods, risk analysis.

Unit – IV: Innovation Tools and Technologies

Use of CAD tools, simulation tools, rapid prototyping, digital fabrication, IoT basics for innovation, data-driven decision making, sustainability considerations.

Unit – V: Entrepreneurship and Project Implementation

Basics of entrepreneurship, innovation to startup journey, intellectual property basics, patents and copyrights, business model canvas, ethics in innovation, team-based mini project.

References

1. Peter F. Drucker, *Innovation and Entrepreneurship*, Harper & Row.
2. Tom Kelley & David Kelley, *Creative Confidence*, Crown Publishing.
3. IDEO, *The Field Guide to Human-Centered Design*.

4. R. Srinivasan, *Engineering Innovation and Design*, McGraw Hill.

5. Tidd, Bessant & Pavitt, *Managing Innovation*, Wiley.

CO_PO MATRIX

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	–	–	–	–	3	2	2	–	1	–	2
CO2	–	3	–	2	–	3	2	–	–	1	–	2
CO3	–	2	3	2	–	3	2	2	–	1	–	2
CO4	2	2	3	2	3	–	2	–	–	–	–	2
CO5	–	–	2	–	–	3	2	3	3	2	1	3
Avg	2	2.3	2.6	2	3	3	2	2.3	3	1.2	1	2.2

Semester: 4th								
Course Name: Theory of Computation	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC204	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	-	-	40	-	60	100
Prerequisite Course and code (if any):								

Course Objectives:

1. Introduce the fundamental concepts of formal languages, automata theory, and grammars.
2. Explain deterministic and non-deterministic finite automata, their equivalence, and applications.
3. Develop understanding of regular expressions, context-free grammars, and related theorems (Arden, Myhill-Nerode, Pumping Lemma).
4. Explain the working of Push Down Automata and their relation with context-free grammars.
5. Introduce Turing machines, undecidability, recursive functions, and the theoretical limits of computation.

Course Outcomes:

CO1: Describe formal languages, Kleene closures, arithmetic expressions, grammar definitions, Chomsky hierarchy, and finite automata (FA).

CO2: Construct, analyze, and optimize deterministic and non-deterministic finite automata, Moore and Mealy machines, and evaluate their applications and limitations.

CO3: Apply Arden's Theorem, Pumping Lemma, and Myhill-Nerode theorem; simplify context-free grammars and understand normal forms and decidability.

CO4: Design and analyze Push Down Automata (PDA), understand acceptance of strings, and relate

PDA to context-free grammars.

CO5: Understand Turing machines (TM), their language acceptance, universal TM, undecidability, recursive functions, and limitations of computation.

etailed Syllabus:

Unit I 6 Hours

Introduction to defining language, Kleene closures, Arithmetic expressions, defining grammar, Chomsky hierarchy, Finite Automata (FA), Transition graph, generalized transition graph.

Unit II 8 Hours

Nondeterministic finite automata (NFA) ,Deterministic finite automata (DFA), Construction of DFA from NFA and optimization, FA with output Moore machine, Mealy machine and Equivalence, Applications and Limitation of FA.

Unit III 8 Hours

Arden Theorem, Pumping Lemma for regular expressions, Myhill-Nerode theorem, Context free grammar: Ambiguity, Simplification of CFGs, Normal forms for CFGs, Pumping lemma for CFLs, Decidability of CFGs, Ambiguous to Unambiguous CFG.

Unit IV 8 Hours

Push Down Automata (PDA): Description and definition, Working of PDA, Acceptance of a string by PDA, PDA and CFG, Introduction to auxiliary PDA and Two stack PDA.

Unit V 10 Hours

Turing machines (TM): Basic model, definition and representation, Language acceptance by TM, TM and Type – 0 grammar, Halting problem of TM,Modifications in TM, Universal TM, Properties of recursive and recursively enumerable languages, unsolvable decision problem, undecidability of Post correspondence problem, Church‟s Thesis, Recursive function theory.

Suggested Books:

Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2007), Introduction to Automata Theory Languages and Computation, 3rd edition, Pearson Education, India.

Reference Books:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	-	-	-	-	1	-	2
CO2	3	3	3	2	2	-	-	-	-	1	1	2
CO3	3	3	3	3	2	-	-	-	1	1	2	2
CO4	3	3	3	2	2	-	-	-	1	1	1	2
CO5	3	3	3	2	3	-	-	-	2	1	1	3
Average	3.0	3.0	2.8	2.2	2.0	–	–	–	1.0	1.0	1.0	2.2

Course Name: Operating System	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC212	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	4	1	2	50	50	50	150
Prerequisite Course and code (if any):								

Course Objective:

This course ensures that the students understand how:

1. Demonstrate understanding of the concepts, structure and design of operating Systems.
2. Demonstrate understanding of operating system design and its impact on application system design and performance.
3. Demonstrate competence in recognizing and using operating system features.
4. To interpret the issues and challenges of memory management.
5. To Encourage students to use operating system tools and utilities for diagnostics and performance monitoring.

Course Outcomes:

Towards the end of the course, the students will be able to:

CO1: Explain the fundamental concepts of an operating system and its role in managing computer resources like CPU, memory, I/O devices, and files.

CO2: Assess the main functionalities of an OS such as process management, memory management, file systems, device management, and user interfaces.

CO3: Understand and implement process scheduling algorithms (e.g., FCFS, SJF, Round Robin, etc.) and evaluate their performance.

CO4: Recognize synchronization techniques such as semaphores, mutexes, and monitors to solve problems related to race conditions, deadlock, and critical sections.

CO5: Analyze and apply memory allocation strategies (e.g., contiguous allocation, dynamic partitioning, and paging).

Detailed Syllabus:

Unit I **6** **Hours**

Importance of Operating Systems, Basic concepts and terminology, Types of Operating Systems, Functions and Structure, Batch and Multiprogramming systems, Timesharing and Real-time systems, Multiprocessor and Distributed systems, Operating System as a Resource Manager, Booting process, and POST.

Unit II **7 Hours**

Functions and concepts of Process, Process states, State transition, Process Control Block (PCB), Events related to process, Process scheduling, Scheduling objectives and levels, Preemptive and Non-preemptive scheduling algorithms, Concurrent processes, Process synchronization, Mutual exclusion, Critical section, Classical problems of mutual exclusion, and Deadlock handling, prevention, avoidance detection and recovery.

Unit III **4 Hours**

Memory management functions, Contiguous and Dynamic memory allocation, Non-contiguous memory management through Segmentation and Paging, Virtual memory, Demand paging, Page replacement policies, and Working set principle.

Unit IV **5 Hours**

Information management and File systems, File system functions, File directories, File system structure and design, Logical and Physical file system layers, File organization, File allocation, and Free space management, File protection and security.

Unit V **5 Hours**

Disk scheduling and scheduling policies, Device management functions and techniques, Dedicated, Shared, and Virtual devices, Spooling, Channels and Control unit, and Case Study: Introduction to Linux/UNIX.

Suggestive Readings:

Text Books:

1. Operating System Concepts, Silber Achatz and Galvin, John Wiley & Sons, 7th Ed., 2005.
2. Operating System, Haldar, Aravind, Pearson Education, 2nd Edition, 2014.

Reference Books:

1. Modern Operating System, Tannenbaum, PHI, 4th Edition, 2016.
2. Operating Systems, Nutt, Pearson Education, 3rd Edition, 2009.

CO-PO Mapping

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	1	1	1	1	1	1	1	2
CO2	3	3	3	2	2	1	1	1	2	1	1	2
CO3	3	3	2	2	1	1	1	1	3	2	1	2
CO4	3	3	2	2	2	1	1	1	2	1	1	2
CO5	3	3	3	2	3	1	1	1	2	1	2	3
Avg	3	2.8	2.2	1.8	1.8	1	1	1	2	1.2	1.2	1.2

List of practical

1. To study of various UNIX editors such as vi, ed, ex and EMACS.
2. Write a Shell program to check the given number is even or odd/Write a Shell program to check the given year is leap year or not
3. To write a C program for implementation of Priority scheduling algorithms.
4. To write a C program for implementation of Round Robin scheduling algorithms.
5. To write a C program for implementation of FCFS and SJF scheduling algorithms.
6. To write a C-program to implement the producer – consumer problem using semaphores.
7. To write a C program to implement banker‟s algorithm for deadlock avoidance.
8. To write a C program to implement algorithm for deadlock detection

9. To write a c program to implement Threading and Synchronization Applications.

10. To write a C program for implementation memory allocation methods for fixed partition using first fit.

11. To write a c program to implement Paging technique for memory management

12. To write a C program for implementation of FIFO page replacement algorithm./To write a C program to organize the file using single level directory

Course Outcomes – Operating System Laboratory

CO1: Use UNIX/Linux commands, editors, and shell scripting to perform basic operating system tasks.

CO2: Implement CPU scheduling algorithms and analyze their performance.

CO3: Implement process synchronization and deadlock handling techniques using system programming concepts.

CO4: Apply memory management techniques such as partitioning, paging, and page replacement algorithms.

CO5: Develop C programs to simulate core operating system functionalities like file organization and resource management.

CO-PO Mapping

PO → CO ↓	PO1	PO 2	PO3	PO 4	PO 5	PO 6	PO 7	PO8	PO 9	PO1 0	PO1 1	PO12
CO1	2	1	1	–	3	–	–	–	–	1	–	1
CO2	3	3	2	2	2	–	–	–	2	1	–	1
CO3	3	3	3	2	2	–	–	–	2	1	–	1
CO4	3	3	3	2	2	–	–	–	1	–	–	1
CO5	2	2	3	2	3	–	–	–	2	1	–	2

Avg	2.6	2.4	2.4	2.4	2.4	0	0	0	1.7	1	0	1.2

Course Name: Computer Organization and Architecture	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC213	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	50	-	50	100
Prerequisite Course and code (if any):								

Course Objective:

1. To impart basic concepts of computer architecture and organization.
2. To explain key skills of constructing cost-effective computer systems.
3. To familiarize the basic CPU organization.
4. To help students in understanding various memory devices.
5. To facilitate students in learning IO communication.

Course Outcomes:

At the end of the course students will be able to:

1. Identify various components of computer and their interconnection
2. Identify basic components and design of the CPU: the ALU and control unit.
3. Compare and select various Memory devices as per requirement.
4. Compare various types of IO mapping techniques
5. Critique the performance issues of cache memory and virtual memory

Detailed Syllabus:

Unit I

STRUCTURE OF COMPUTERS: Computer types, Functional units, Basic operational concepts, Von-Neumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Data representation, Fixed and Floating point, Error detection and correction codes. COMPUTER

ARITHMETIC: Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations.

Unit II
BASIC COMPUTER ORGANIZATION AND DESIGN: Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC

Unit III
REGISTER TRANSFER AND MICRO-OPERATIONS: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit. MICRO-PROGRAMMED CONTROL: Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit.

Unit IV
MEMORY SYSTEM: Memory Hierarchy, Semiconductor Memories, RAM(Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID.

Unit V
INPUT OUTPUT: I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. MULTIPROCESSORS: Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Inter processor Communication and Synchronization, Cache Coherence.

Suggestive Readings:

Text Books:

T1. M Moris Mano, Computer System Architecture, 3rd edition, Pearson/PHI, India

T2. William Stallings, Computer Organization and Architecture- designing for performance, 8th edition, Prentice Hall, New Jersey.

Reference books:

R1. Anrew S. Tanenbaum, Structured Computer Organization, 5th edition, Pearson Education Inc.

R2. N. Carter-Computer Architecture, Schaums Outline Series, TMH, New Delhi, 2006, Pearson Higher Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), Computer Organization, 5th Edition, McGraw Hill. New Delhi. India.

CO-PO Mapping:

PO → CO ↓	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	2				3	3	3		2	2
CO2	3	2	2	2								3
CO3	2	3	3				2	2	2		3	3
CO4					2	2				3		
CO5		2	3				3		2	3	3	
Avg	2.6	1.8	2.5	2	2	2	2.6	2.5	3.5	3	2.6	2.6

Course Name: Discrete Structure	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE218	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objectives:

1. Introduce students to the fundamental concepts of discrete mathematics that form the mathematical foundation of computer science.
2. Develop logical reasoning and problem-solving skills using propositional and predicate logic.
3. Provide knowledge of sets, relations, and functions and their applications in computing.
4. Explain combinatorics, counting techniques, and recurrence relations for algorithm analysis.
5. Familiarize students with graph theory and trees and their applications in networks and data structures.

Course Outcomes:

Towards the end of the course, the students will be able to:

CO1. Apply principles of logic to construct valid arguments and solve computational problems
CO2. Analyze and manipulate sets, relations, and functions relevant to computer science applications.
CO3. Solve counting and combinatorial problems using permutations, combinations, and recurrence relations.
CO4.Use Boolean algebra and algebraic structures to design and analyze digital circuits and logical systems..
CO5.Demonstrate mathematical reasoning skills necessary for algorithm design, analysis, and advanced computing studies.

Unit I **8 Hour**

Set Theory: Introduction, Combination of sets, Multiset Ordered pairs, Set Identities.

Relations: Definition, Operations on relations, Properties of relations, Composite Relations, Equality of relations, Order of relations.

Functions: Definition, Classification of functions, Operations on functions, Recursively defined functions. Natural Numbers: Introduction, Mathematical Induction, Variants of Induction, Induction with Non zero Basecases

Unit II **8 Hour**

Partial order sets: Definition, Partial order sets, Combination of partial order sets, Hasse diagram.Lattices: Definition, Properties of lattices – Bounded, Complemented, Modular Complete Lattice, Morphisms of lattices.Boolean Algebra: Introduction, Axioms and Theorems of Boolean algebra.

Unit III **8 Hour**

Algebraic Structures: Definition, Groups, Subgroups and order Cyclic Groups, Cossets, Lagrange's theorem, Normal Subgroups, Permutation and Symmetric groups, Group Homeomorphisms Definition and elementary properties of Rings and Fields.

Unit IV **8 Hour**

Trees Definition, Binary tree, Binary tree traversal ,Binary search tree.

Graphs: Definition and terminology, Representation of graphs, Multi graphs, Bipartite graphs, Planar graphs, Isomorphism and Homeomorphism of graphs, Euler and Hamiltonian paths, Graph coloring.

Recurrence Relation & Generating function: Recursive definition of functions, Recursive algorithms, Method of solving recurrences.

Combinatory: Introduction, Counting Techniques, Pigeonhole Principle

Unit V **8 Hour**

Propositional Logic: Proposition, well formed formula, Truth tables, Tautology, Satisfiability Contradiction, Algebra of proposition, Theory of Inference, Natural Deduction. Predicate Logic: First order predicate, well formed formula of predicate, quantifiers, Inference theory of predicate logic.

Suggestive Readings:

Text Books:

1.Liuand Mohapatra,“Elements of Discrete Mathematics”,McGrawHill

2.Jean Paul Trembley, R Manohar, Discrete Mathematical Structures with Application to Computer Science, McGraw-Hill

Reference Books:

1. Y.N.Singh,“DiscreteMathematicalStructures”,WileyIndia,NewDelhi,FirstEdition,August2010***.***

2.B.Kolman,R.C.Busby,andS.C.Ross,Discrete Mathematical Structures, PHILearning Private Limited, Delhi India.

CO-PO Mapping:

PO → CO ↓	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	2				3	3	3		2	2
CO2	3	2	2	2								3
CO3	2	3	3				2	2	2		3	3
CO4					2	2				3		
CO5		2	3				3		2	3	3	
Avg	2.6	1.8	2.5	2	2	2	2.6	2.5	3.5	3	2.6	2.6

Course Name: Engineering Innovation II(Design)	Credit Scheme				Evaluation Components			
Course Code: E1002	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	1	1	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objectives

- Introduce students to the **engineering design process** with emphasis on creativity and innovation.
- Develop skills in **problem identification, need analysis, and conceptual design**.
- Enable students to apply **design thinking, prototyping, and modeling tools**.
- Encourage **team-based project work** and interdisciplinary collaboration.
- Strengthen communication skills through **design documentation and presentations**.

Course Outcomes (CO)

Towards the end of the course, the students will be able to:

CO1. Identify real-world engineering problems and formulate clear design requirements.
CO2. Apply design thinking methodologies to generate innovative solutions.
CO3. Develop conceptual and detailed designs using appropriate engineering tools.
CO4. Build and test prototypes considering constraints such as cost, safety, and sustainability.
CO5. Demonstrate teamwork, project management, and technical communication skills.

DETAILED SYLLABUS

Unit I – Design Thinking & Problem Identification (8 Hours)
Introduction to Engineering Design, Design Thinking Process, Need Identification, Problem Definition, Stakeholder Analysis, Market and User Research, Design Constraints and Specifications.

Unit II – Concept Generation & Creativity Tools (8 Hours)
Brainstorming Techniques, Mind Mapping, TRIZ Basics, Morphological Analysis, Concept Screening and Selection, Feasibility Analysis.

Unit III – Engineering Design & Modeling (8 Hours)
System-Level Design, Functional Decomposition, Block Diagrams, Engineering Drawings, Introduction to CAD Tools, Design Standards and Tolerances.

Unit IV – Prototyping & Testing (8 Hours)
Types of Prototypes (Low-fidelity & High-fidelity), Materials and Manufacturing Considerations, Rapid Prototyping, Testing Methods, Design Iteration and Optimization.

Unit V – Project Implementation & Sustainability (8 Hours)
Project Planning and Scheduling, Risk Analysis, Cost Estimation, Sustainability and Ethics in Design, Documentation, Technical Report Writing, Design Presentation and Demonstration.

Suggested References

1. Karl T. Ulrich & Steven D. Eppinger, *Product Design and Development*, McGraw Hill.
2. David G. Ullman, *The Mechanical Design Process*, McGraw Hill.
   IDEO, *The Field Guide to Human-Centered Design*.
3. R. Srinivasan, *Engineering Innovation and Design*, McGraw Hill.
4. ISO and BIS Standards (relevant to design domain).

CO-PO- matrix

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	2	2	–	2	2	1	–	1	–	2
CO2	2	2	3	3	–	2	2	–	–	1	–	2
CO3	2	2	3	3	3	–	2	–	–	–	–	2
CO4	–	2	2	2	–	3	3	2	–	–	–	2
CO5	–	–	2	–	–	3	2	3	3	2	1	3
Avg	2	2.2	2.4	2.2.	3	2.2	2.2	2	2	1.3	1	2.2

Course Name: Soft Skill Enhancement II	**Credit Scheme**				**Evaluation Components**			
Course Code: SSE 002	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	2	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objectives

1. To strengthen advanced communication and interpersonal skills for academic and professional environments.
2. To enhance presentation, group discussion, and professional writing abilities.
3. To develop leadership, teamwork, and emotional intelligence skills.
4. To improve employability through interview skills, resume building, and workplace etiquette.
5. To foster ethical behavior, self-awareness, and lifelong learning attitudes.

Course Outcomes (COs)

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

- **CO1:** Communicate effectively in professional and academic contexts using appropriate verbal and non-verbal skills.
  CO2: Demonstrate effective interpersonal skills including teamwork, leadership, and conflict management.

- **CO3:** Prepare and deliver structured presentations and participate confidently in group discussions.
- **CO4:** Develop professional documents such as resumes, emails, and reports following workplace standards.
  CO5: Exhibit ethical behavior, emotional intelligence, and self-management skills in personal and professional life.

Detailed Syllabus

Unit I: Advanced Communication Skills

Professional communication, verbal and non-verbal cues, listening skills, assertive communication, barriers to communication, cross-cultural communication.

Unit II: Interpersonal Skills and Emotional Intelligence

Self-awareness, emotional intelligence, empathy, personality development, leadership styles, teamwork, conflict resolution.

Unit III: Group Discussion and Presentation Skills

Group discussion techniques, role plays, public speaking, presentation structure, visual aids, handling questions and feedback.

Unit IV: Professional Writing and Workplace Etiquette

Resume and cover letter writing, professional emails, report writing basics, workplace ethics, time

management, corporate etiquette.

Unit V: Employability and Career Readiness

Interview skills, mock interviews, aptitude for placements, stress management, goal setting, lifelong learning and adaptability.

Suggested Readings

1. Barun K. Mitra, *Personality Development & Soft Skills*, Oxford University Press.
2. Dale Carnegie, *How to Win Friends and Influence People*.
3. Daniel Goleman, *Emotional Intelligence*, Bantam Books.
4. Meenakshi Raman & Sangeeta Sharma, *Technical Communication*, Oxford.

CO-PO Matrix

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	–	–	–	–	–	2	–	2	3	3	–	2
CO2	–	–	–	–	–	2	–	3	3	2	1	2
CO3	–	–	1	–	–	–	–	2	3	3	–	2
CO4	–	–	–	–	–	–	–	2	2	3	1	2
CO5	–	–	–	–	–	3	–	3	2	2	1	3
Avg	0	0	1	0	0	2.3	0	2.4	2.2	2.2	1	2.2

5.

Semester: 5th								
Course Name: Data Communication and Computer Networks	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC301	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objective:

1. To understand fundamental concepts and architectures of computer networks.
2. To study data transmission, modulation, and communication techniques.
3. To analyze switching, multiplexing, and error control mechanisms.
4. To learn routing algorithms and congestion control strategies.
5. To explore internetworking, TCP/IP model, and network services.

Course Outcomes:

CO1: Understand the OSI model, transmission media, and modulation techniques.
CO2: Implement data link protocols and error correction methods.
CO3: Analyze routing algorithms and WAN technologies.
CO4: Apply TCP/IP concepts for addressing, fragmentation, and communication.
CO5: Demonstrate understanding of various network services and applications.

Detailed Syllabus:

Unit I 6 Hours

Introduction: Motivation, OSI model, Signals and media, Bits over signals, Synchronous communication, Modulation and modems, Bandwidth, Throughput, and noise, Time division and Frequency division multiplexing,Standards, Switching methods, ISDN.

Unit II **8 Hours**

Packet Transmission:Multiplexing, Frames, Error correction techniques, LAN/WAN/MAN, Topology, CSMA/CD, LANprotocol, Elementary Data link protocol- Sliding window protocols, Token passing rings, FDDI, IEEE802.3, 802.5.

Unit III **6 Hours**

Routing Algorithms: Distance-Vector, Link-State, Shortest path computation, Dijkstra's algorithm, Congestion control, WANtechnologies including frame relay, X.25, ATM.

Unit IV **8 Hours**

Internetworking: Motivation, Concept, Goals, TCP/IP model, IP addressing with sub netting, Address binding with ARP,IP Datagram, Encapsulation IP fragmentation and reassembly, ICMP, IGMP, TCP.

Unit V **6 Hours**

Network Services: Electronic mail, File transfer, Access and management, Virtual terminals, Remote procedure call.

Suggested Books:

Text Books:

1. Data communication and Networking, Forouzan, B.A, McGraw Hill, 4thEdition.
2. Computer Networks, Tanenbaum, A.S., Prentice Hall,4th Edition.

Reference Books:

1. Internetworking with TCP/IP Vol. 1 Principles, Comer, D.E., Prentice Hall of India, 5th Edition.
2. Computer Networking with Internet Protocols and Tech, Stallings, W., Prentice Hall of India.

CO-PO Mapping:

CO → / PO ↓	PO1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	1	1	–	–	–	–	–	–	–	–	–	–
CO2	1	1	1	–	–	–	–	–	–	–	–	–
CO3	1	1	1	1	–	–	–	–	–	–	–	–

CO4	1	1	1	1	1	–	–	–	–	–	–	–
CO5	1	1	1	–	1	–	–	–	1	–	–	–
Average	1.0	1.0	1.0	0.6	0.6	0	0	0	0.2	0	0	0

List of Practical

1. **Study of networking devices and media** Study of hub, switch, router, modem, cables, and connectors.
2. **Study of network topologies** Bus, star, ring, mesh, and hybrid topologies.
3. **Data transmission modes and media** Study of simplex, half-duplex, full-duplex communication and transmission media.
4. **Implementation of error detection techniques** Programs for: Parity Check,Checksum, Cyclic Redundancy Check (CRC)
5. **Implementation of error correction techniques** Program for Hamming code.
6. **Implementation of flow control protocols** Stop-and-Wait protocol.
7. **Implementation of sliding window protocols** Go-Back-N and Selective Repeat ARQ.
8. **Study and simulation of OSI model** Layer-wise functionality and data flow.
9. **Study and simulation of TCP/IP model** Comparison of OSI and TCP/IP models.
10. **Socket programming – Client and Server** Implement simple client-server communication using TCP.
11. **UDP socket programming** Implement communication using UDP protocol.
12. **Network configuration and troubleshooting** Use of networking commands: `ping`, `tracert`, `ipconfig/ifconfig`, `netstat`, `arp`.
13. **Study of routing algorithms** Shortest path routing (e.g., Dijkstra's algorithm).
14. **Study of congestion control mechanisms** TCP congestion control and avoidance techniques.
15. **Study of network security basics** Firewalls, encryption, and authentication concepts.

Course Outcomes: Data Communication and Computer Networks

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

CO1: Explain the fundamentals of data communication, transmission media, and network topologies.

CO2: Describe layered network architectures such as the OSI and TCP/IP models and their functions.

CO3: Analyze data link layer mechanisms including error detection, error correction, and flow control techniques.

CO4: Apply network layer concepts such as IP addressing, routing algorithms, and congestion control mechanisms.

CO5: Implement and analyze transport layer protocols including TCP and UDP for reliable and unreliable communication.

CO-PO Mapping:

CO → / PO ↓	PO1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	–	–	1	–	–	–	–	–	–	1
CO2	3	2	1	–	1	–	–	–	–	–	–	1
CO3	3	3	2	2	1	–	–	–	–	–	–	1
CO4	3	3	3	2	2	–	–	–	–	–	–	1
CO5	2	2	3	2	3	–	–	–	2	1	–	2
Average	2.8	2.4	1.8	1.2	1.6	–	–	–	0.4	0.2	–	1.2

Course Name: Compiler Design	Credit Scheme				Evaluation Components			
Course Code: CSEC302	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. This course ensures that the students understand how:
2. Understand the structure and components of a compiler.
3. Apply lexical and syntax analysis techniques to design a compiler front-end.
4. Optimize intermediate code for better performance.
5. Generate efficient target code and manage runtime environments.

Course Outcomes:

Towards the end of the course, the students will be able to:
CO1: Describe the different phases of the compiler and their functionalities.
CO2: Apply lexical analysis and parsing techniques to extract language constructs from source code.
CO3: Construct syntax trees, symbol tables, and perform type checking.
CO4: Generate intermediate code and apply code optimization techniques.
CO5: Translate intermediate representation into target code and manage memory allocation.

Detailed Syllabus:

UNIT I 9 Hours

Introduction: Phases of a Compiler, Structure of a Compiler, Lexical Analysis & Role of Lexical Analyzer, Input Buffering, Specification & Recognition of Tokens, Lex Tool: Intro and Use Cases, Finite Automata – NFA & DFA, Converting Regex to Automata, Minimization of DFA.

UNIT II 10 Hours

Role of Parser, Context-Free Grammars (CFGs), Writing a Grammar, Error Handling in Parsing, Top-Down Parsing Strategies, Recursive Descent Parser, Predictive Parser – LL(1) Table, Bottom-Up Parsing – Shift Reduce, LR(0) Items and Parsing Table, SLR & LALR Parsing, Error Recovery & YACC.

UNIT III 9 Hours

Syntax-Directed Definitions (SDDs), Evaluation Orders of SDD, Syntax-Directed Translation Schemes, L-Attributed SDDs Implementation, Intermediate Code Overview, Three Address Code (TAC), Types, Declarations, Type Checking, Control Flow & Switch Statements, Intermediate Code for Procedures.

UNIT IV 10 Hours

Runtime Environment: Overview, Stack Allocation, Access to Nonlocal Data, Heap Management & Garbage Collection, Trace-based Collection, Code Generator: Issues & Target Code, Basic Blocks, Flow Graphs, Code Optimization: Peephole, Register Allocation & Dynamic Programming.

UNIT V 5 Hours

Optimization: Sources & Overview, Data Flow Analysis Foundations, Constant Propagation & PRE, Loop Optimizations, Peephole Optimization & Review.

Suggestive Readings:

Text Books:

1. Aho, Lam, Sethi, Ullman, *Compilers: Principles, Techniques, and Tools*, 2nd Edition, Pearson Education (Dragon Book)
2. Alfred V. Aho and Jeffrey D. Ullman, *Principles of Compiler Design*, Addison-Wesley

Reference Books:

1. Keith D. Cooper and Linda Torczon, *Engineering a Compiler*, Elsevier
2. Andrew W. Appel, *Modern Compiler Implementation in C*, Cambridge University Press

CO-PO Mapping

PO → CO ↓	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	–	1	–	–	–	–	–	–	1	1
CO2	3	3	3	2	2	–	–	–	–	–	2	2
CO3	3	3	3	2	2	–	–	–	–	–	2	2
CO4	3	3	3	3	2	–	–	–	–	–	2	2
CO5	3	3	3	2	2	–	–	–	–	–	2	2
Avg	3	2.8	3	2	2	0	0	0	0	0	1.8	1.8

Course Name: Software Engineering	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC214	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	-	-	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To introduce the fundamental principles, concepts, and methodologies of software engineering.
2. To develop the ability to design and implement software using appropriate models and best practices.
3. To equip students with knowledge of project management, software quality assurance, and testing strategies.
4. To create awareness about ethical issues, legal standards, and the role of emerging technologies in software development.
5. To enable students to build scalable, secure, and high-performance software solutions for real-world applications.

Course Outcomes:

CO1: Understand fundamental concepts, principles, and methodologies of software engineering.
CO2: Apply software development models, design patterns, and best practices in real-world applications.
CO3: Analyze software requirements, architecture, and project management strategies.
CO4: Evaluate software quality, testing techniques, and maintenance strategies for reliability.
CO5: Implement secure, scalable, and efficient software solutions using modern tools and technologies.

Detailed Syllabus:

Unit I **12 Hours**

What is Software Engineering and its history, software crisis, Evolution of a Programming System Product, Characteristics of Software, Brooks' No Silver Bullet, Software Myths. Software Development Life Cycles: Software Development Process, The Code-and-Fix model, The Waterfall model, The Evolutionary Model, The Incremental Implementation, Prototyping, The Spiral Model, Software Reuse, Critical Comparisons of SDLC models, An Introduction to Non-Traditional Software Development Process: Rational Unified Process, Rapid Application Development, Agile Development Process.

Unit II **8 Hours**

Requirements: Importance of Requirement Analysis, User Needs, Software Features and Software Requirements, Classes of User Requirements: Enduring and Volatile, Sub phases of Requirement Analysis, Functional and Non-functional requirements, Barriers to Eliciting User requirements, The software requirements document and SRS standards, Requirements Engineering, Case Study of SRS for a Real Time System. Tools for Requirements Gathering: Document Flow Chart, Decision Table, Decision Tree, Structured Analysis: DFD, Data Dictionary, Introduction to non-traditional Requirements.

Unit III **6 Hours**

Software Design: Goals of good software design, Design strategies and methodologies, Data oriented software design, Structured Design: Structure chart, Coupling, Cohesion, Modular structure, Packaging, Object oriented design, Top-down and bottom-up approach, Design patterns.

Unit IV **8 Hours**

Software Measurement and Metrics: Various Size Oriented Measures: Halestead's software science, Function Point(FP) based measures, Cyclomatic Complexity Measures: Control flow graphs. Development: Selecting a language, Coding guidelines, Writing code, Code documentation.

Unit V **10 Hours**

Software Testing: Testing process, Design of test cases, Functional Testing: Boundary value analysis, Equivalence class testing, Decision table testing, Cause effect graphing, Structural testing, Path testing, Data flow and mutation testing, Unit testing, Integration and system testing, Debugging, Alpha & beta testing, testing tools & standards. Software Maintenance: Management of maintenance, Maintenance process, Maintenance models, Regression testing, Reverse engineering, Software re-engineering, Configuration management, documentation.

Suggestive Readings:

Text Books:

1. R. S. Pressman, Software Engineering: A Practitioner's Approach, McGrawHill, Seventh Edition, 2010.
2. P. K. J. Mohapatra, Software Engineering (A Life cycle Approach), New Age International Publishers, First Edition, 2009.

Reference Books:

1. K.K. Aggarwal, Yogesh Singh, Software Engineering (3rd Edition), New Age International Publishers, Third Edition, 2007.
2. Ian Sommerville, Software Engineering, Addison Wesley, Ninth Edition, 2010.

CO-PO MTRAIX

PO → CO ↓	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	3	3	3	2	2	-	-	-	2	-	-	2
CO3	3	3	2	2	-	-	-	-	3	2	-	2
CO4	3	3	2	2	2	-	-	-	2	-	-	2
CO5	3	3	3	2	3	-	-	-	2	-	2	3
Avg	3	2.8	2.5	2	2.3	0	0	0	2.2	2	2	2.2

Course Name: Human Values	**Credit Scheme**				**Evaluation Components**			
Course Code: HSC304	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	2	-	-	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To help students understand the importance of human values in personal and professional life.
2. To develop clarity about ethical conduct, moral responsibilities, and social harmony.
3. To promote self-awareness, emotional intelligence, and responsible citizenship.
4. To nurture universal values aligned with sustainable development and national integrity.
5. To encourage ethical decision-making in engineering and technology practice.

Course Outcomes:

CO1: Understand the concept of human values and their significance in individual and societal life.
CO2: Develop ethical reasoning and decision-making abilities.
CO3: Recognize responsibilities toward family, society, nation, and environment.
CO4: Apply professional ethics in engineering practices.
CO5: Promote harmony, sustainability, and social responsibility in personal and professional domains.

Detailed Syllabus:

Unit I: Introduction to Human Values and Self-Exploration (6 Hours)

Meaning and importance of Human Values; Self-exploration – understanding the self (I) and body; Needs of self and body; Happiness and prosperity; Continuous happiness and sustainable living; Right understanding vs. assumptions; Basic human aspirations; Harmony in the self.

Unit II: Understanding Harmony in the Family and Society (6 Hours)

Harmony in family – trust and respect; Relationship values – affection, care, guidance, reverence, gratitude; Role of family in value formation; Harmony in society; Concept of justice; Mutual fulfillment in human relationships; Social responsibilities and citizenship values.

Unit III: Professional Ethics and Engineering Values (6 Hours)

Introduction to professional ethics; Ethical dilemmas in engineering; Integrity, honesty, accountability, and transparency; Code of conduct for engineers; Corporate social responsibility (CSR); Ethical use of technology; Digital ethics and cyber responsibility; Intellectual property rights (IPR).

Unit IV: Environmental Ethics and Sustainable Development (6 Hours)

Relationship between human beings and nature; Environmental degradation and its impact; Sustainable development principles; SDGs (Sustainable Development Goals); Conservation of natural resources; Climate change awareness; Eco-friendly technologies and green engineering practices.

Unit V: Human Values in National and Global Context (6 Hours)

Constitutional values; Fundamental rights and duties; National integration; Cultural diversity and unity; Global citizenship; Peace, tolerance, and inclusiveness; Gender equality; Leadership qualities and value-based decision making.

Suggestive Readings:

Text Books:

1. R.R. Gaur, R. Sangal & G.P. Bagaria – *A Foundation Course in Human Values and Professional Ethics*, Excel Books.
2. A.N. Tripathi – *Human Values*, New Age International Publishers.

Reference Books:

1. M. Govindarajan, S. Natarajan & V.S. Senthil Kumar – *Professional Ethics and Human Values*, PHI Learning.
2. B.L. Bajpai – *Indian Ethos and Modern Management*, New Royal Book Co.
3. Universal Human Values Manual (AICTE recommended).

CO-PO MTRAIX

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	–	–	–	2	2	3	2	2	–	2
CO2	1	2	–	–	–	2	2	3	2	2	–	2
CO3	–	1	–	–	–	3	3	3	2	2	–	2

CO4	1	2	1	–	1	2	2	3	2	2	–	2
CO5	–	1	–	–	–	3	3	3	2	2	–	3
Avg	1	1.4	1	-	1	2.4	2.4	3	2	2	-	2.2

Course Name: Engineering Innovation – III (Implement)	**Credit Scheme**				**Evaluation Components**			
Course Code: EI 003	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	1	0	-	2	-	50		50
Prerequisite Course and code (if any):								

Course Objective:

To enable students to implement innovative solutions identified in earlier stages.

To develop prototype development, testing, and validation skills.

To enhance teamwork, leadership, and project management skills.

To promote entrepreneurial thinking and value creation.

To prepare students for industry-ready innovation and technology deployment.

Course Outcomes:

CO1: Translate innovative ideas into working prototypes or functional systems.
CO2: Apply engineering tools and techniques for implementation and testing.
CO3: Demonstrate teamwork, leadership, and project coordination skills.
CO4: Validate solutions based on performance, feasibility, and user feedback.
CO5: Present and document the implemented solution professionally.

Detailed Syllabus:

Module I: Project Planning & Implementation Strategy (4 Hours)

Problem refinement and solution finalization, Feasibility analysis (technical, financial, operational) Resource planning and timeline preparation, Risk identification and mitigation strategy

Module II: Prototype Development / System Implementation (12 Hours)

Hardware/Software implementation, Use of modern engineering tools and platforms, Integration of components, Version control and documentation, Debugging and troubleshooting

Module III: Testing, Validation & Optimization (6 Hours)

Functional testing, Performance testing, User feedback collection, Refinement and optimization

Cost-benefit analysis

Module IV: Entrepreneurship & Deployment Aspects (4 Hours)

Market viability analysis, Intellectual Property awareness, Business model basics, Sustainable and scalable solutions

Module V: Final Demonstration & Documentation (4 Hours)

Prototype demonstration, Technical report writing, Poster presentation / Model display, Viva-voce examination

List of Practical Activities

1. Identification and refinement of innovation problem statement.
2. reparation of project implementation plan.
3. Development of working prototype/software model.
4. Testing and debugging sessions.
5. Performance validation and documentation.
6. Business feasibility discussion.
7. Final demonstration and presentation..

Suggestive Readings:
Text Books:

Reference Books:

CO-PO MTRAIX

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	3	1	1	–	2	2	2	3
CO2	3	3	3	3	3	1	1	–	2	2	2	3
CO3	2	2	2	2	2	1	1	2	3	3	2	3
CO4	3	3	3	3	3	2	2	2	2	2	3	3
CO5	2	2	2	2	2	1	1	2	3	3	2	3

Avg	2.6	2.6	2.6	2.4	2.6	1.2	1.2	2	2.4	2.4	2.2	3

Course Name: Employability Skills	**Credit Scheme**				**Evaluation Components**			
Course Code: ES 001	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	1	-	2	-	50	50	100
Prerequisite Course and code (if any):								

Course Objective:
To develop verbal and non-verbal communication skills.
To enhance quantitative aptitude and logical reasoning abilities.
To strengthen professional etiquette and workplace behavior.
To prepare students for interviews, group discussions, and corporate recruitment processes.
To build confidence, leadership qualities, and lifelong learning attitude.

Course Outcomes:
CO1: Communicate effectively in professional and workplace environments.
CO2: Apply quantitative aptitude and logical reasoning for competitive examinations.
CO3: Demonstrate professional behavior, teamwork, and leadership skills.
CO4: Prepare effective resumes and perform confidently in interviews and group discussions.
CO5: Develop career planning strategies and continuous self-improvement practices.

Detailed Syllabus:

Unit I: Communication Skills (Lecture + Practical) 8 Hours

Verbal and non-verbal communication, Listening skills and presentation skills, Public speaking techniques. Business communication basics, Email and professional correspondence

Unit II: Quantitative Aptitude & Logical Reasoning – 10 Hours

Number systems, Percentage, Profit & Loss, Ratio & Proportion, Time & Work, Simple & Compound Interest, Logical reasoning and analytical puzzles

Unit III: Soft Skills & Professional Etiquette – 8 Hours

Personality development, Emotional intelligence, Teamwork and leadership, Conflict management, Workplace ethics.

Unit IV: Resume Writing & Interview Skills – 10 Hours

Resume and CV preparation, Cover letter writing, Interview types (HR, Technical, Behavioral) Group Discussion techniques, Corporate expectations

Unit V: Career Planning & Entrepreneurship Orientation – 9 Hours

Career goal setting, SWOT analysis, Introduction to entrepreneurship, Workplace digital skills Lifelong learning and upskilling.

List of practical

1. **Role play, group discussions, presentations, mock conversations**
2. **Problem-solving sessions, timed aptitude practice tests.**
3. Case studies, leadership activities, situational exercises.
4. Mock interviews, GD practice, resume evaluation.
5. Personal career plan preparation, business idea presentation.

Suggestive Readings:

Text Books:

Reference Books:

CO-PO MTRAIX

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	–	–	–	–	1	–	2	3	3	–	2
CO2	2	3	1	–	–	–	–	–	–	–	–	2
CO3	1	1	1	–	–	2	–	2	3	3	1	2
CO4	1	–	–	–	–	1	–	2	3	3	–	2
CO5	–	1	–	–	–	2	1	2	2	2	1	3
Avg	1.3	1.7	1.0	-	-	1.5	1.0	2.0	2.8	2.8	1.0	2.2

Semester: 6th								
Course Name: Web Technology with PHP	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC316	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2				150
Prerequisite Course and code (if any):								

Course Objective:
The course aims to:

1. Understand the architecture and working principles of the Internet, web browsers, servers, and web protocols.
2. Learn to design and develop static, responsive, and visually appealing web pages using HTML and CSS.
3. Acquire skills to implement interactivity in web pages through client-side scripting using JavaScript.
4. Develop proficiency in creating dynamic and event-driven web applications using client-side and server-side technologies.
5. Gain practical knowledge of database connectivity and backend development using PHP for building complete web applications.

Course Outcomes:
On completion of this course, the students will be able to:
CO1: Apply the knowledge of the internet and client side and server side concepts in understanding and developing various applications and Analyze and develop static interactive web pages using HTML and CSS
CO2: Create scripts using JavaScript in a web page and Integrate JavaScript in a web page.
CO3: Design forms and check for data accuracy and Design and develop web applications.

CO4: Understand, analyze and develop event-driven programs using Client side programming like JavaScript
CO5: Analyze and develop database connectivity programs and Understand, analyze and develop web applications using Server side programming.

Detailed Syllabus:

Unit I **10 Hours**
Introduction: Concept of WWW, Internet and WWW, HTTP Protocol Request and Response, Web browser and Web servers, Features of latest version of Web.
Coding Basics: Introduction to HTML Tags. The lang attribute The meta tag & the unicode character set Coding Links: Absolute & Relative URLs Anchor tags & hrefs Linking to other websites Linking to pages within a website Opening a link in a new browser window/tab Adding Images The break tag The image tag & source attribute.

Unit II **8 Hours**
Cascading Style Sheet: Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS.

Unit III **8 Hours**
Javascript: What is Javascript, How to develop Javascript, simple Javascript, variables, functions, conditions, loops and repetition.Advance script, Javascript and objects, Javascript own object.

Unit IV **9 Hours**
Introduction to PHP: PHP introduction, inventions and versions, important tools and software requirements (like Web Server, Database, Editors etc.), PHP with other technologies, scope of PHP, Basic Syntax, PHP variables and constants, Types of data in PHP , Expressions, scopes of a variable (local, global), PHP Operators : Arithmetic, Assignment, Relational , Logical operators, Bitwise , ternary and MOD operator. PHP operator Precedence and associatively.

Unit V **10 Hours**
PHP conditional events and Loops: PHP IF Else conditional statements (Nested IF and Else), Switch case, while ,For and Do While Loop, Goto , Break ,Continue and exit.
PHP Functions: Function, Need of Function , declaration and calling of a function, PHP Function with arguments.Function argument with call by value, call by reference, Scope of Function Global and Local S*tring* Manipulation and Regular Expression: Creating and accessing String , Searching & Replacing String, Formatting, joining and splitting String , String Related Library functions, Use and advantage of regular expression over inbuilt function.Array: Anatomy of an Array ,Creating index based and Associative array.

Suggested Books:
Text Books:

1. "Web Enable Commercial Application Using HTML, DHTML", Ivan Bay Ross- BPB Publication.

Reference Books:

1. "Web Technology: A developer perspective", Gopalan&Akilandeswari, PHI
2. Programming the World Wide Web, Robert W Sebesta, 7ed, Pearson.

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	1	-	-	-	-	1	1	2
CO2	3	2	3	2	2	-	-	-	1	1	1	2
CO3	3	3	3	2	2	-	-	-	1	1	2	2
CO4	3	3	3	3	2	-	-	-	1	2	2	3
CO5	3	3	3	3	3	-	-	-	2	2	3	3
Average	**3.0**	**2.6**	**2.8**	**2.2**	**2.0**	**0.0**	**0.0**	**0.0**	**1.0**	**1.4**	**1.8**	**2.4**

List of practical

HTML & CSS

1. **Basic HTML Pages:** Create a simple web page with headings, paragraphs, images, and hyperlinks.

2. **Lists and Tables:** Design web pages with ordered, unordered, and nested lists; create tables with rowspan and colspan.

3. **Forms in HTML:** Design forms with input fields, checkboxes, radio buttons, dropdowns, and submit/reset buttons.

4. **CSS Styling:** Apply CSS for colors, fonts, borders, and backgrounds. Use inline, internal, and external CSS.

5. **CSS Layouts:** Create responsive web pages using float, flexbox, and grid layouts.

JavaScript / Client-Side Scripting

6. **JavaScript Basics:** Write scripts for variables, operators, loops, and conditions.
7. **Event Handling:** Implement JavaScript events like `onclick`, `onmouseover`, `onchange`.
8. **Form Validation:** Validate form inputs for mandatory fields, email, phone numbers, and numeric fields.
9. **DOM Manipulation:** Dynamically change content and styles using JavaScript.

PHP / Server-Side Scripting

10. **PHP Basics:** Create PHP scripts to display text, variables, and constants.
11. **Conditional Statements and Loops:** Write programs using `if-else`, `switch-case`, `for`, `while`, and `do-while` loops.
12. **PHP Functions:** Create functions with parameters, return values, and demonstrate call by value and call by reference.
13. **String Manipulation:** Perform operations like string concatenation, searching, replacing, splitting, and formatting.
14. **Arrays in PHP:** Create indexed, associative, and multidimensional arrays and manipulate data.
15. **Form Handling with PHP:** Retrieve form data using `$_GET` and `$_POST` and display it dynamically.

Course Outcomes – Web Technology with PHP Lab

After successful completion of the lab, students will be able to:

CO1: Design and develop static and interactive web pages using HTML and CSS.

CO2: Implement client-side scripts using JavaScript to add interactivity and validate user inputs.

CO3: Develop web forms and ensure data accuracy through form validation techniques.

CO4: Create event-driven web programs using JavaScript to handle user actions.

CO5: Build dynamic web applications using PHP and integrate them with a backend database for complete web solutions.

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	1	–	–	–	–	1	1	2
CO2	3	2	3	2	2	–	–	–	1	1	1	2
CO3	3	3	3	2	2	–	–	–	1	1	2	2
CO4	3	3	3	3	2	–	–	–	1	2	2	3
CO5	3	3	3	3	3	–	–	–	2	2	3	3
Average	3	2.6	2.8	2.2	2	0	0	0	1	1.4	1.8	2.4

Course Name: Artificial Intelligence	Credit Scheme				Evaluation Components			
Course Code: CSEC317	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	2				150
Prerequisite Course and code (if any):								

Course Objective:
This course aims to

1. To introduce the fundamental principles and concepts of Artificial Intelligence.
2. To develop an understanding of search strategies and their role in problem-solving.
3. To provide knowledge of knowledge representation techniques and logical reasoning methods.
4. To expose students to various learning approaches, including statistical and reinforcement learning.
5. To enable the design and application of intelligent systems for solving real-world problems.

Course Outcomes:
CO1: Describe the nature of AI problems, the evolution of AI, and the structure and behavior of intelligent agents in various environments.
CO2: Apply uninformed and heuristic search strategies, including A*, AO*, and adversarial search techniques like minimax and alpha-beta pruning, to solve goal-oriented and game-based problems.
CO3: Represent knowledge using logic, semantic networks, and rules, and perform reasoning under uncertainty using Bayesian inference and Dempster-Shafer theory.
CO4: Use first-order logic and inference techniques such as forward/backward chaining, resolution, and apply machine learning methods including decision trees, statistical learning and reinforcement learning.
CO5: Explain the architecture and functions of expert systems, and apply reasoning approaches including rule-based, frame-based and case-based reasoning, addressing uncertainty and knowledge acquisition challenges.

Detailed Syllabus:

Unit I

AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

Unit II

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A*, AO* Algorithms, Problem reduction, Game Playing-Adversal search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

Unit III

Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Baye's probabilistic interferences and dempstershafer theory.

Unit IV

First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation. Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

Unit V

Expert systems:- Introduction, basic concepts, structure of expert systems, the human element in expert systems how expert systems works, problem areas addressed by expert systems, expert systems success factors, types of expert systems, expert systems and the internet interacts web, knowledge engineering, scope of knowledge, difficulties, in knowledge acquisition methods of knowledge acquisition, machine learning, intelligent agents, selecting an appropriate knowledge acquisition method, societal impacts reasoning in artificial intelligence, inference with rules, with frames: model based reasoning, case based reasoning, explanation & meta knowledge inference with uncertainty representing uncertainty.

Suggestive Readings:

Text Books:

1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", Second Edition, Pearson Education.

2. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence : a logical approach", Oxford University Press.

Reference Books:

1. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problemsolving", Fourth Edition, Pearson Education.
2. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.

CO – PO Mapping

PO → CO ↓	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	3	3	3	2	3	-	-	-	-	-	-	2
CO3	3	3	2	2	2	-	-	-	-	-	-	2
CO4	3	3	3	3	3	-	-	-	-	-	-	2
CO5	2	3	2	2	-	-	-	2	-	-	-	2
Avg	2.8	2.8	2.2	2.2	2.6	0	0	2	0	0	0	2

Course Name: Server Side Programming Language	Credit Scheme				Evaluation Components			
Course Code: CSEC318C	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	1	0	40	0	60	100
Prerequisite Course and code (if any):								

Course Objective:
This course aims to

1. Understand how servers, databases, and front-end interfaces interact to create full-stack applications.
2. Develop proficiency in server-side programming languages such as PHP, Python, Ruby, or JavaScript (Node.js).
3. Learn to design, query, and manage relational (SQL) and non-relational (NoSQL) databases.
4. Use server-side logic to generate dynamic content and handle user interactions.
5. Understand the principles of securing web applications against threats such as SQL injection, cross-site scripting (XSS), and session hijacking.

Course Outcomes:
CO1: Students will be able to **Remember** the fundamental concepts of server-side scripting, including its role in web development and interaction with databases and client-side technologies.
CO2: Students will to **able** to Explain the workings of HTTP protocols, request/response cycles, and server-client communication in dynamic web applications.
CO3:Students will able to **Apply** basic server-side scripts to handle form submissions, process user inputs, and generate dynamic content.
CO4:Students will able to **Analyze** web application requirements to design appropriate server-side solutions that meet performance and scalability needs.
CO5:Students will able to **Evaluate** the security vulnerabilities of server-side applications and apply suitable countermeasures to ensure data integrity and user privacy.

Detailed Syllabus:

Unit I

Introduction to PHP: PHP introduction, inventions and versions, important tools and software requirements (like Web Server, Database, Editors etc.), PHP with other technologies, scope of PHP, Basic Syntax, PHP variables and constants, Types of data in PHP.

Unit II

Expressions, scopes of a variable (local, global), PHP Operators : Arithmetic, Assignment, Relational , Logical operators, Bitwise , ternary and MOD operator. PHP operator Precedence and associatively. PHP conditional events and Loops: PHP IF Else conditional statements (Nested IF and Else), Switch case, while ,For and Do While Loop, Goto , Break ,Continue and exit.

Unit III

Handling HTML form with PHP: Capturing Form Data, GET and POST form methods, Dealing with multi value fields, Redirecting a form after submission. PHP Functions: Function, Need of Function , declaration and calling of a function, PHP Function with arguments, Default Arguments in Function, Function argument with call by value, call by reference, Scope of Function Global and Local

Unit IV

String Manipulation and Regular Expression: Creating and accessing String , Searching & Replacing String, Formatting, joining and splitting String , String Related Library functions, Use and advantage of regular expression over inbuilt function, Use of preg_match(), preg_replace(), preg_split() functions in regular expression.

Unit V

Array: Anatomy of an Array ,Creating index based and Associative array ,Accessing array, Looping with Index based array, with associative array using each() and foreach(), Some useful Library function.

Connectivity of PHP with MySQL .

Suggestive Readings:

Text Books:

1. **"Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5"** by Robin Nixon
2. **"Node.js Design Patterns"** by Mario Casciaro and Luciano Mammino
3. **"Flask Web Development: Developing Web Applications with Python"** by Miguel Grinberg
4. **"PHP and MySQL Web Development"** by Luke Welling and Laura Thomson

References:

1. PHP: PHP Manual Comprehensive guide for PHP functions, classes, and best practices.
2. **Node.js**: Node.js Documentation In-depth references for Node.js APIs, modules, and features.
3. Python (Django): Django Documentation Detailed documentation for Django, including tutorials and API references.
4. **Python (Flask)**: Flask Documentation Covers Flask features, extensions, and examples.

CO – PO Mapping

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	2
CO2	2	3	3	2	3	-	-	-	-	-	-	2
CO3	3	2	2	2	2	-	-	-	-	-	-	2
CO4	3	3	2	3	3	-	-	-	-	-	-	3
CO5	3	3	2	2	-	-	-	2	-	-	-	2
Avg	2.8	2.8	2.2	2.2	2.6	0	0	2	0	0	0	2

Semester: 7th

Course Name: Advance Web Technology	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE 401	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4							
Prerequisite Course and code (if any):								

Course Objective:

1. Understand the fundamental concepts, characteristics, and challenges of Big Data and its impact across industries.
2. Explain NoSQL databases and their significance in large-scale data storage and management.
3. Explore the architecture and functioning of the Hadoop framework and its core components (HDFS, MapReduce).
4. Develop the ability to design and implement MapReduce-based applications for data analysis.
5. Familiarize with advanced Big Data tools such as HBase, Cassandra, Pig, and Hive for large-scale data processing and analytics.

Course Outcomes:

CO1: Describe the concepts of Big Data, its characteristics, technologies, and applications across various domains.

CO2: Analyze the structure, functioning, and data models of NoSQL databases for Big Data management.

CO3: Demonstrate understanding of the Hadoop ecosystem, its components, and the data flow in HDFS.
CO4: Develop and test MapReduce applications for distributed data processing and analysis.
CO5: Apply Big Data analytical tools like HBase, Cassandra, Pig, and Hive for data manipulation and querying.

Detailed Syllabus:

Unit I **8 Hours**

Introduction to Big Data: Big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data ,credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open source technologies, cloud and big data mobile business intelligence, Crowd sourcing analytics ,inter and trans firewall analytics.

Unit II **8 Hours**

NOSQL Data Management: Introduction to NoSQL , aggregate data models ,aggregates , key-value and document data models, relationships, graph databases, schema less databases ,materialized views,distribution models ,sharding , Masterslave replication , peer-peer replication , sharding and replication , consistency , relaxing consistency , version stamps , mapreduce , partitioning and combining , composing map-reduce calculations.

Unit III **8 Hours**

Basics of HADOOP: Data format , analyzing data with Hadoop , scaling out , Hadoop streaming , Hadoop pipes , design of Hadoop distributed file system (HDFS) , HDFS concepts , Java interface , data flow ,Hadoop I/O , data integrity , compression , serialization , Avro file-based data structures.

Unit IV **8 Hours**

MapReduce Applications: Map Reduce workflows , unit tests with MRUnit , test data and local tests – anatomy of Map Reduce job run , classic Map-reduce , YARN , failures in classic Map-reduce and YARN , job scheduling , shuffle and sort , task execution , MapReduce types , input formats , output formats.

Unit V **8 Hours**

HADOOP Related Tools: Hbase, data model and implementations, Hbase clients ,Hbase examples – praxis.Cassandra ,cassandra data model , cassandra examples , cassandra clients , Hadoop integration. Pig , Grunt , pig data model , Pig Latin , developing and testing Pig Latin scripts. Hive , data types and file formats , HiveQL data definition , HiveQL data manipulation – HiveQL queries.

Suggested Books:
Text Books:

1. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley.
2. DT Editorial Services, Big-Data Black Book, Wiley.

Reference Books:

1. Dirk deRoos, Chris Eaton, George Lapis, Paul Zikopoulos, Tom Deutsch, "Understanding Big Data Analytics for Enterprise Class Hadoop and Streaming Data", McGraw Hill.
2. Thomas Erl, Wajid Khattak, Paul Buhler, "Big Data Fundamentals: Concepts, Drivers and Techniques", Prentice Hall.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	2	2	–	–	–	–	2	–	2
CO2	3	3	2	2	3	–	–	–	–	2	–	2
CO3	3	3	3	3	3	–	–	–	–	2	–	2
CO4	3	3	3	3	3	–	–	–	–	3	2	3
CO5	3	3	3	3	3	–	–	–	–	3	3	3

List of Practicals – Advanced Web Technology Lab

HTML5 & CSS3

1. **Semantic HTML5:** Create web pages using semantic tags (`<header>`, `<footer>`, `<article>`, `<section>`, `<nav>`).

2. **Multimedia Integration:** Embed audio, video, and canvas elements in web pages.

3. **Responsive Design:** Use CSS3 media queries to make pages responsive for desktop, tablet, and mobile.

4. **Advanced CSS3 Effects:** Implement transitions, animations, gradients, shadows, and transformations.

JavaScript / Client-Side Programming

5. **DOM Manipulation:** Create dynamic content by manipulating DOM elements.
6. **Event Handling:** Implement mouse, keyboard, and form events to enhance user interaction.
7. **Form Validation:** Validate form data using JavaScript and display custom error messages.
8. **AJAX Requests:** Use AJAX to fetch data from a server without reloading the web page.

Front-End Frameworks

9. **Bootstrap Framework:** Build responsive layouts using the Bootstrap grid system and components.
10. **jQuery / DOM Scripting:** Implement dynamic effects and event handling using jQuery.
11. **JSON Data Handling:** Fetch and parse JSON data to dynamically display on web pages.

PHP / Server-Side Programming

12. **Advanced PHP Functions:** Use built-in PHP functions for arrays, strings, and date/time operations.
13. **Form Handling with PHP:** Process and validate HTML form data on the server.
14. **Session and Cookie Management:** Maintain user sessions and cookies across web pages.
15. **File Handling:** Upload, read, write, and manage files on the server using PHP

Course Outcomes – Advanced Web Technology Lab

After successful completion of the lab, students will be able to:

CO1: Design and develop responsive and interactive web pages using HTML5, CSS3, and client-side scripting.

CO2: Implement dynamic content and user interactions using JavaScript, DOM manipulation, events,

and AJAX.

CO3: Integrate PHP server-side programming to handle forms, sessions, cookies, and file management for dynamic web applications.

CO4: Connect web applications with MySQL databases to perform CRUD operations and implement secure data handling techniques.

CO5: Develop a complete, advanced web application combining front-end, back-end, and database connectivity, adhering to modern web standards and best practices.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	2	2	–	–	–	–	2	–	2
CO2	3	3	2	2	3	–	–	–	–	2	–	2
CO3	3	3	3	3	3	–	–	–	–	2	–	2
CO4	3	3	3	3	3	–	–	–	–	3	2	3
CO5	3	3	3	3	3	–	–	–	–	3	3	3
Avg	3	2.8	2.4	2.6	2.8	0	0	0	0	2.4	1	2.4

Course Name: Advanced Computer Architecture	Credit Scheme				Evaluation Components			
Course Code: CSE 403	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3							
Prerequisite Course and code (if any):								

Course Objective:

1. Understand the evolution of parallel computer architectures and models of parallel computation.
2. Explore performance laws, program partitioning, and scheduling techniques for achieving efficient parallel execution.
3. Analyze advanced processor architectures including RISC, CISC, superscalar, VLIW, and SIMD systems.
4. Examine the design and organization of multiprocessor and multicomputer systems, including interconnection and communication mechanisms.
5. Study data flow and VLSI computing architectures for high-performance and reconfigurable computing applications.

Course Outcomes: On successful completion of this course, students will be able to:

CO1: Explain the fundamentals and models of parallel computer architecture, parallelism, and performance evaluation laws.

CO2: Analyze memory hierarchy, cache design, shared memory organization, and bus systems for parallel computers.

CO3: Compare different processor architectures such as RISC, CISC, superscalar, VLIW, and SIMD systems.

CO4: Evaluate multiprocessor interconnects, pipelining mechanisms, and message-passing techniques for performance optimization.
CO5: Illustrate data flow and VLSI computation models, systolic array designs, and matrix arithmetic processors for high-performance computing.

Detailed Syllabus

UNIT- I **9 Hours**
PARALLEL COMPUTER MODELS Evolution of Computer architecture, system attributes to performance, Multi processors and multi computers, Multi-vector and SIMD computers, PRAM and VLSI models-Parallelism in Programming, conditions for Parallelism-Program Partitioning and Scheduling-program flow Mechanisms-Speed up performance laws-Amdahl‘s law, Gustafson‘s law-Memory bounded speedup Model.

UNIT- II **9 Hours**
MEMORY SYSTEMS AND BUSES Memory hierarchy-cache and shared memory concepts-Cache memory organization-cache addressing models, Aliasing problem in cache, cache memory mapping techniques-Shared memory organization-Interleaved memory organization, Lower order interleaving, Higher order interleaving. Back plane bus systems-Bus addressing, arbitration and transaction.

UNIT -III **9 Hours**
ADVANCED PROCESSORS Instruction set architectures-CISC and RISC scalar processors-Super scalar processors-VLIW architecture- Multivector and SIMD computers-Vector processing principles-Cray Y-MP 816 system-Inter processor communication

UNIT- IV **9 Hours**
MULTI PROCESSOR AND MULTI COMPUTERS Multiprocessor system interconnects- Cross bar switch, Multiport memory-Hot spot problem, Message passing mechanisms-Pipelined processors-Linear pipeline, on linear pipelineInstruction pipeline design-Arithmetic pipeline design.

UNIT- V **9 Hours**
DATA FLOW COMPUTERS AND VLSI COMPUTATIONS Data flow computer architectures-Static, Dynamic-VLSI Computing Structures-Systolic array architecture, mapping algorithms into systolic arrays, Reconfigurable processor array-VLSI matrix arithmetic processors-VLSI arithmetic models, partitioned matrix algorithms, matrix arithmetic pipelines.

Suggested Books:

Text Books:
1. J.L.Hennessy, D.A.Patterson, Computer Architecture: a quantitative approach, Morgan Kaufmann, 5th edition, 2011, ISBN: 978-1558605961.
2. William Stallings, Computer Organization and Architecture, Prentice Hall, 10th edition, 2015, ISBN-10: 013293633X, ISBN-13: 978-0132936330.

Reference Books:

1. Andrew S. Tanenbaum, Structured Computer Organization, Prentice Hall, 6th edition, 2012, ISBN: 978-0132916523.
2. Patterson, J.L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Morgan Kaufmann, 5th edition, 2013, ISBN-13:9780124078864
3. C. Hamacher, Z. Vranesic and S. Zaky, Computer Organization, McGraw-Hill, 5th edition,2002, ISBN: 0072320869.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	2	–	–	–	–	2	–	2
CO2	3	3	3	3	2	–	–	–	–	2	–	2
CO3	3	3	3	3	3	–	–	–	–	3	–	2
CO4	3	3	3	3	3	–	–	–	–	3	2	3
CO5	3	3	3	3	3	–	–	–	–	3	3	3
Avg	3	3	2.8	2.8	2.6	0	0	0	0	2.6	2.5	2.4

Course Name: Mobile Computing	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE 406	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2	2	0	0	50	-	50	100
Prerequisite Course and code (if any):								

Course Objectives:

1. Understand the fundamentals of mobile computing and wireless communication.
2. Explore the architecture and functioning of mobile devices and networks (e.g., 3G, 4G, 5G, Wi-Fi, Bluetooth).
3. Analyze mobile application platforms and operating systems (such as Android and iOS).
4. Design and develop basic mobile applications.
5. Examine security challenges and solutions in mobile computing.

Course Outcomes: After completing this course the student will be able to

CO1: To elaborate the next generation Mobile Communication System.
CO2: To understand network and transport layers of Mobile Communication
CO3: Analyze various protocols of all layers for mobile and ad hoc wireless communication networks.
CO4: To Understand IP and TCP layers of Mobile Communication.
CO5. To Compose Pointers and Structures to Develop Algorithms and Programs.

Detailed Syllabus:

Unit I **5 Hours**

Evolution of wireless networks; Wire line data networks; Wireless data networks; Networks, middleware, and gateways; Applications, services, and security; Evolution of mobile communication systems and paging systems.

Unit II **5 Hours**

Three-tier architecture; Design considerations for mobile computing; Multiple access techniques – FDMA, TDMA, CDMA, and SDMA – principles and comparison of efficiency and application scenarios.

Unit III **5 Hours**

GSM principles and architecture; GPRS architecture; EDGE technology; 2G cellular network architecture and functionalities; Data services and evolution of packet-switched mobile communication systems.

Unit IV **4 Hours**

Frequency reuse concept; Channel allocation strategies and handoff techniques; Methods to improve coverage and capacity in cellular systems; Parameters for mobile multipath channels and causes of small-scale fading.

Unit V **6 Hours**

Wireless systems and standards: integration between wireless and wired telephone networks; Bluetooth and RFID technologies; Mobile IP and IPv6; JAVA Card and CDMA digital cellular standard; Wireless Application Protocol (WAP); MMS and GPRS-based applications; Operating systems for mobile devices, design constraints in handheld applications; Palm and Symbian OS features and architectures.

Suggested Readings

1. **Text Book**
   T1. Mobile Computing: Technology, Applications and service creation by Asoke K. Talukdarand Roopa R. Yavagal, Tata Mc Graw Hill Education Pvt Ltd, New Delhi.

2. **Reference Books**

1.Raj kamal "Mobile computing" 3rd edition(2019)

2.Asoke K. Talukder & Roopa R. Yavagal "Mobile computing: Technology application & Service creation" 2nd Edition

CO-PO Mapping:

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	–	–	–	–	–	–	1	–	2
CO2	3	3	2	–	1	–	–	–	–	1	–	2

CO3	2	3	2	–	2	–	–	–	–	2	–	2
CO4	2	3	3	2	1	–	–	–	–	2	–	3
CO5	3	2	3	2	3	–	–	–	–	2	–	3
Avg	2.6	2.6	2.2	2	1.7	0	0	0	0	1.6	0	2.4

Course Name: .Net technology using VB.net	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE 407	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4							
Prerequisite Course and code (if any):								

Course Objective:

1. Features of Visual Basic .NET
2. Concepts of Object Oriented Programming Language
3. Concept of Class, Abstract Class, Interface, Inheritance
4. Learn about controls, events available in Toolbox
5. How to perform Database Connectivity

Course Outcome:

CO1: Describing the basics of design, create, build, and debug Visual Basic applications.
CO2: Implementing one and two dimensional arrays for sorting, calculating, and displaying of data.
CO3: Handle exceptions and debug errors to ensure the reliability and stability of applications.
CO4: Structuring decision structures for determining different operations and structures to perform repetitive tasks. Also organizing procedures, sub procedures and functions.
CO5: Create interactive user interfaces using forms and controls, and manage user interactions effectively and database connectivity.

Detailed Syllabus

Unit I **9 Hours**

Evolution of Computer Architecture – System Attributes to Performance – Multiprocessors and Multicomputers – Multivector and SIMD Computers – PRAM and VLSI Models – Parallelism in Programming – Conditions for Parallelism – Program Partitioning and Scheduling – Program Flow Mechanisms – Speedup Performance Laws – Amdahl's Law – Gustafson's Law – Memory Bounded Speedup Model.

Unit II **9 Hours**

Memory Hierarchy – Cache and Shared Memory Concepts – Cache Memory Organization – Cache Addressing Models – Aliasing Problem in Cache – Cache Memory Mapping Techniques – Shared Memory Organization – Interleaved Memory Organization – Lower Order Interleaving – Higher Order Interleaving – Backplane Bus Systems – Bus Addressing – Arbitration and Transaction.

Unit III **9 Hours**

Instruction Set Architectures – CISC and RISC Scalar Processors – Superscalar Processors – VLIW Architecture – Multivector and SIMD Computers – Vector Processing Principles – Cray Y-MP 816 System – Interprocessor Communication.

Unit IV **9 Hours**

Multiprocessor System Interconnects – Crossbar Switch – Multiport Memory – Hot Spot Problem – Message Passing Mechanisms – Pipelined Processors – Linear Pipeline – Nonlinear Pipeline – Instruction Pipeline Design – Arithmetic Pipeline Design.

Unit V **9 Hours**

Data Flow Computer Architectures – Static and Dynamic Models – VLSI Computing Structures – Systolic Array Architecture – Mapping Algorithms into Systolic Arrays – Reconfigurable Processor Array – VLSI Matrix Arithmetic Processors – VLSI Arithmetic Models – Partitioned Matrix Algorithms – Matrix Arithmetic Pipelines.

Suggested Books:

Text Books:

T1: Visual Basic .net Comprehensive Concepts and Techniques Author:- Shelly, cashman, Quasney.
T2: Visual Basic .net Author:- Steven Holzner Dream Tech Press Latest Edition.

References:

R1: Murach's Beginning Visual Basic .NET Author:- Anne Prince
R2: Programming in Visual Basic. NET Author:- Julia Case Bradley, Anita C. Millspaugh

CO-PO Mapping

COs / POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	-	-	-	-	-	-	-	-	1

CO2	3	3	2	2	-	-	-	-	-	1	-	1
CO3	2	3	3	2	1	-	-	-	-	1	-	1
CO4	2	3	3	3	2	-	-	-	-	1	-	1
CO5	2	2	3	3	2	-	-	-	-	1	1	1
Average	**2.4**	**2.6**	**2.4**	**2.0**	**1.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.8**	**0.2**	**1.0**

ELECTIVE

(Artificial Intelligence & Machine Learning)

Course Name: AI & ML Foundations	**Credit Scheme**				**Evaluation Components**			
Course Code:CSEA205	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To understand the basic concepts and history of Artificial Intelligence and Machine Learning.

2. To learn different types of learning paradigms used in ML.

3. To analyze problem-solving techniques and intelligent agents.

4. To gain foundational knowledge of data, models, and algorithms used in ML.

5. To apply AI & ML concepts to simple real-world scenarios.

Course Outcomes:

1. Explain core concepts, definitions, and applications of Artificial Intelligence and Machine Learning.
2. Classify different types of machine learning approaches such as supervised, unsupervised, and reinforcement learning.
3. Apply basic AI problem-solving techniques and search strategies.
4. Understand fundamental ML algorithms and model evaluation methods.
5. Identify ethical issues, limitations, and future trends in AI and ML.

Detailed Syllabus:

UNIT I **Introduction to Artificial Intelligence** 8 Hours

Definition and scope of Artificial Intelligence , History and evolution of AI, Branches and applications of AI Intelligent systems and environments , Intelligent agents: types, architecture, and characteristics, AI techniques: advantages and limitations

UNIT II **Problem Solving and Search Techniques** 8 Hours

Problem formulation and state space representation, Uninformed search strategies: Breadth First Search (BFS), Depth First Search (DFS), Uniform Cost Search, Informed search strategies:Best First Search A* algorithm Heuristic functions and their design ,Constraint Satisfaction Problems (CSP)

UNIT III **Fundamentals of Machine Learning** 8 Hours

Introduction to Machine Learning, Difference between AI, ML, and Deep Learning, Types of data: structured and unstructured, Data preprocessing and feature selection, Training, testing, and validation datasets, Bias–variance tradeoff, Overfitting and underfitting

UNIT IV **Machine Learning Algorithms** 8 Hours

Supervised learning: Linear Regression, Logistic Regression, k-Nearest Neighbors (k-NN), Decision Trees, Unsupervised learning: Clustering (K-means), Hierarchical clustering Introduction to Reinforcement Learning, Basic concepts of neural networks

UNIT V **Model Evaluation, Ethics, and Applications** 8 Hours

Model evaluation metrics: Accuracy, Precision, Recall, F1-Score, Confusion matrixCross-validation techniques , Ethical issues in AI and ML , Privacy, bias, and fairness Limitations and future trends in AI & ML , Case studies and real-world applications

Suggestive Readings:

Text Books:

1. Stuart Russell & Peter Norvig, *Artificial Intelligence: A Modern Approach*, Pearson Education.
2. Tom M. Mitchell, *Machine Learning*, McGraw-Hill Education.

Reference Books:

1. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer.
2. Sebastian Raschka & Vahid Mirjalili, *Python Machine Learning*, Packt Publishing.

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	2	2	1	-	-	1	-	-
CO2	2	3	2	2	3	-	-	-	1	-	-
CO3	2	2	3		1	-	-	-	–	-	-
CO4	3	3	3	2	2	1	-	-	2	2	1
CO5	3	3	3	3	3	1	-	-	2	2	1
Avg	2.6	2.6	2.6	2.2	2.2	1	0	0	1.2	2	1

Course Name: Fuzzy Logic & Neural Networks	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEA219	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To introduce the fundamental concepts of fuzzy sets, fuzzy logic, and soft computing techniques.
2. To understand the structure and working principles of artificial neural networks.
3. To study different learning algorithms and neural network architectures.
4. To analyze the role of fuzzy inference systems in decision-making and control applications.
5. To develop the ability to design and apply fuzzy and neural models to real-world problems.

Course Outcomes:

1. Explain the concepts of fuzzy sets, membership functions, and fuzzy reasoning.
2. Design and implement basic fuzzy logic systems for control and decision-making problems.
3. Describe the architecture and functioning of artificial neural networks.

4. Apply neural network learning algorithms such as perceptron and backpropagation.

5. Analyze and solve real-world problems using neural networks, fuzzy systems, or hybrid neuro-fuzzy models.

Detailed Syllabus:

UNIT I **Introduction to Soft Computing** 8 Hours

Limitations of conventional (hard) computing, Concept of Soft Computing , Components of Soft Computing: Fuzzy Logic, Neural Networks, Genetic Algorithms (overview), Comparison between Soft Computing and Hard Computing, Applications of Soft Computing in engineering and real-world systems

UNIT II **Fuzzy Logic Fundamentals** 8 Hours

Classical sets vs Fuzzy sets, Fuzzy set operations, Membership functions: types and properties Fuzzy relations and composition, Fuzzy propositions and fuzzy reasoning, Fuzzy inference systems: Mamdani and Sugeno models, Fuzzification and Defuzzification methods, Applications of Fuzzy Logic in control systems and decision making

UNIT III **Artificial Neural Networks – Basics** 8 Hours

Biological neuron vs Artificial neuron, Neural network models, Network architectures: single layer and multilayer networks, Activation functions, Learning paradigms: supervised, unsupervised, and reinforcement learning, Perceptron model and learning rule, Limitations of perceptron

UNIT IV **Neural Network Learning Algorithms** 8 Hours

Multilayer Perceptron (MLP), Backpropagation algorithm, Error surfaces and convergence issues Radial Basis Function (RBF) networks, Self-Organizing Maps (SOM), Associative memory models, Applications of neural networks in pattern recognition and classification

UNIT V **Neuro-Fuzzy Systems and Applications** 8 Hours

Introduction to Neuro-Fuzzy systems, Adaptive Neuro-Fuzzy Inference System (ANFIS), Architecture and learning of ANFIS, Comparison of Fuzzy Systems and Neural Networks, Hybrid Soft Computing systems, Case studies and real-world applicationsFuture trends in intelligent systems

Suggestive Readings:

Text Books:

1. Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Wiley India
2. Simon Haykin, *Neural Networks and Learning Machines*, Pearson Education

Reference Books:

1. Kosko, *Neural Networks and Fuzzy Systems*
2. S. Rajasekaran and G.A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic*

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2	2	3	2	1	-	-	1	-	1
CO2	3	3	2	2	3	-	-	-	1	-	-
CO3	1	3	2		2	-	-	-	–	-	2
CO4	3	3	3	2	3	2	-	-	2	2	1
CO5	3	3	3	3	3	1	-	-	2	2	1
Avg	2.4	2.8	2.4	2.2	2.6	1.3	0	0	1.4	2	1.2

Course Name: Machine Learning using Python	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEA305	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

To introduce students to the fundamental concepts, principles, and types of machine learning and their applications across various domains.

To develop proficiency in Python programming for implementing machine learning algorithms and handling real-world datasets.

To strengthen students' understanding of statistical and mathematical foundations required for model building and evaluation.

To provide knowledge of supervised, unsupervised, and neural network-based learning techniques for solving practical problems.

To cultivate the ability to design, analyze, and optimize end-to-end machine learning projects, preparing students for advanced studies and industry applications in AI and ML.

Course Outcomes:

6. Identify appropriate machine learning algorithms for various types of learning tasks
7. across domains.
8. Implement machine learning algorithms using Python on real-world datasets.
9. Explain the fundamental concepts of neural networks and deep learning.
10. Develop end-to-end machine learning projects using Python for practical applications.

11. Evaluate and optimize machine learning models using performance metrics and critical analysis.

Detailed Syllabus:

UNIT I 8 Hours

Introduction to Data Science and Digital Data – Structured, Unstructured, Semi-structured; Machine Learning – Definition, Need and Concept of Learning; Types of Machine Learning – Supervised, Unsupervised, Semi-supervised, Reinforcement Learning; Applications of Machine Learning in Healthcare, Finance, and Retail.

UNIT II 8 Hours

Python for Machine Learning – Python Objects Overview, Vectors, Lists, Factors, Matrices, Arrays, Data Frames, Manipulating Objects, Functions and Looping Constructs, Input/Output Operations, Data Handling – Import and Export, Mini Lab Review.

UNIT III 8 Hours

Descriptive Statistics – Central Tendency (Mean, Median, Mode), Dispersion (Variance, Standard Deviation), Shape (Skewness, Kurtosis), Percentiles and Five-Point Summary; Data Visualization – Boxplot, Histogram, Barplot, Pie Chart, Scatter Plot; Two-way Tables, Covariance, Correlation, Chi-Square Test for Association.

UNIT IV 8 Hours

Clustering – Introduction, Applications, Similarity Measures, K-means Clustering; Regression – Concept and Applications, Simple Linear Regression, Multiple Linear Regression; Classification – Overview, Logistic Regression.

UNIT V 8 Hours

Decision Trees, k-Nearest Neighbors, Support Vector Machines; Neural Networks – Introduction, Activation Functions, Learning Rate, Stochastic Gradient Descent (SGD), Feedforward and Backpropagation; Deep Learning – Basic Concepts and Recent Trends.

Suggestive Readings:

Text Books:

3. Stephen P. Robbins & Mary Coulter, “Management”, 15e, Pearson Education, Harlow, 2021
4. Stephen P. Robbins, Timothy A. Judge, Neharika Vohra, “Organizational Behaviour”, 18e, Pearson Education, India, 2019

Reference Books:

3. *Introduction to Data Mining* – Pang-Ning Tan, Michael Steinbach, Vipin Kumar, 2016
4. *An Introduction to Statistical Learning* – Trevor Hastie, Robert Tibshirani, 2016

5. *Applied Predictive Modeling* – Max Kuhn, Kjell Johnson, 2013.

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	2	2	1	-	-	1	-	-
CO2	3	3	2	2	3	-	-	-	1	-	-
CO3	2	2	2		2	-	-	-	–	-	-
CO4	3	3	3	2	3	1	-	-	2	2	1
CO5	3	3	3	3	3	1	-	-	2	2	1
Avg	3	2.6	2.4	2.2	2.6	1	0	0	1.2	2	1

Course Name: Deep Learning using R	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEA319	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. Understand the fundamentals of deep learning and neural network architectures.
2. Apply deep learning concepts using R programming for data analysis tasks.
3. Design and implement deep neural network models using appropriate R libraries.
4. Analyze and evaluate model performance using suitable metrics and validation techniques.
5. Explore applications of deep learning in real-world domains such as healthcare, finance, and image processing.

Course Outcomes:

1. Explain the principles and working of artificial neural networks and deep learning models.
2. Perform data preprocessing and visualization for deep learning applications using R.
3. Develop and train deep learning models using R libraries such as Keras and TensorFlow.
4. Evaluate, optimize, and fine-tune deep learning models to improve performance.
5. Apply deep learning techniques using R to solve real-world problems and interpret results effectively.

Detailed Syllabus:

UNIT I **Introduction to Deep Learning** 8 Hours

Overview of Artificial Intelligence, Machine Learning, and Deep Learning, Biological neuron vs Artificial neuron , Basics of Artificial Neural Networks (ANN), Types of neural networks Activation functions: Sigmoid, Tanh, ReLU, Softmax, Loss functions and optimization basics Introduction to R programming for deep learning

UNIT II **Deep Neural Networks** 8 Hours

Architecture of Deep Neural Networks (DNN), Forward propagation and Backpropagation, Gradient descent and optimization algorithms (SGD, Adam, RMSprop), Weight initialization and learning rate tuning, Regularization techniques: L1, L2, Dropout, Overfitting and underfitting

UNIT III **Deep Learning with R** 8 Hours

R environment setup for deep learning, Introduction to TensorFlow and Keras in R, Data preprocessing and normalization in R, Building, compiling, and training models using Keras Model evaluation and performance metrics, Visualization of results using R

UNIT IV **Advanced Deep Learning Models** 8 Hours

Convolutional Neural Networks (CNN): architecture and applications, Pooling techniques and convolution operations, Recurrent Neural Networks (RNN) ,Long Short-Term Memory (LSTM) and GRU. Sequence modeling and time-series analysis, Transfer learning concepts

UNIT V **Applications and Case Studies** 8 Hours

Convolutional Neural Networks (CNN): architecture and applications, Pooling techniques and convolution operations, Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) and GRU Sequence modeling and time-series analysis, Transfer learning concepts

Suggestive Readings:

Text Books:

1. François Chollet, *Deep Learning with R*, Manning Publications, 2nd Edition.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press.

Reference Books:

1. J. Han, M. Kamber, J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufmann.
2. Brett Lantz, *Machine Learning with R*, Packt Publishing.

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	1	2	2	2	2	1	-	-	2	-	2

CO2	3	3	1	2	1	-	-	-	1	-	-
CO3	2	2	2		2	-	-	-	–	-	-
CO4	2	3	3	2	3	1	-	-	2	2	1
CO5	3	3	3	3	3	1	-	-	1	2	1
Avg	2.4	2.6	2.4	2.2	2.4	1	0	0	1.2	2	1.3

Course Name: Computer Vision theory	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEA404	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To introduce the fundamental concepts and principles of Computer Vision and image formation.
2. To explain various image representation, enhancement, and preprocessing techniques.
3. To study feature extraction, image segmentation, and object detection methods.
4. To understand the role of machine learning and deep learning in Computer Vision applications.
5. To familiarize students with real-world applications and challenges in Computer Vision systems.

Course Outcomes:

1. Explain the basic concepts of Computer Vision and the process of digital image formation.
2. Apply image preprocessing and enhancement techniques to improve image quality.
3. Analyze and implement feature extraction and image segmentation techniques for visual data.
4. Demonstrate understanding of object detection and recognition using traditional and deep learning approaches.
5. Identify and evaluate Computer Vision applications and challenges in real-world scenarios.

Detailed Syllabus:

UNIT I **Introduction to Computer Vision** 8 Hours

Overview of Computer Vision – Human Vision vs Computer Vision – Applications of Computer Vision – Image formation process – Digital image fundamentals – Sampling and quantization – Types of images – Image acquisition – Basic image operations.

UNIT II **Image Preprocessing and Enhancement** 8 Hours

Image enhancement techniques – Spatial domain methods – Point processing – Histogram processing – Image smoothing and sharpening – Noise models – Noise reduction techniques – Frequency domain filtering – Introduction to morphological operations.

UNIT III **Feature Extraction and Image Segmentation** 8 Hours

Edge detection techniques (Sobel, Prewitt, Canny) – Corner detection – Texture analysis – Feature descriptors – Image segmentation concepts – Thresholding techniques – Region-based segmentation – Edge-based segmentation – Clustering methods (K-means) – Watershed algorithm.

UNIT IV **Object Detection and Recognition** 8 Hours

Object detection concepts – Template matching – Feature-based recognition – Traditional approaches (HOG, SVM) – Introduction to machine learning in Computer Vision – Object tracking – Motion analysis – Face detection and recognition.

UNIT V **Deep Learning and Applications of Computer Vision** 8 Hours

Introduction to deep learning – Convolutional Neural Networks (CNN) – CNN architecture – Training and optimization – Popular CNN models – Object detection using deep learning (YOLO, SSD, Faster R-CNN) – Applications in medical imaging, autonomous vehicles, surveillance, robotics – Challenges and future trends in Computer Vision.

Suggestive Readings:

Text Books:

1. Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Pearson Education.
2. Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer.

Reference Books:

1. Milan Sonka, Vaclav Hlavac, Roger Boyle, *Image Processing, Analysis, and Machine Vision*, Cengage Learning.

2. Emanuele Trucco and Alessandro Verri, *Introductory Techniques for 3D Computer Vision*, Prentice Hall.

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	1	2	2	2	2	1	-	-	2	-	2
CO2	3	3	1	2	1	-	-	-	1	-	-
CO3	2	2	2		2	-	-	-	–	-	-
CO4	2	3	3	2	3	1	-	-	2	2	1
CO5	3	3	3	3	3	1	-	-	1	2	1
Avg	2.4	2.6	2.4	2.2	2.4	1	0	0	1.2	2	1.3

ELECTIVE

(Cyber Security)

Course Name: Information Security Fundamentals	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE C216	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2				150
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Understand fundamental concepts of information and cyber security
2. Learn common threats, vulnerabilities, and attacks
3. Gain knowledge of security mechanisms, policies, and standards
4. Understand cryptography basics and system security
5. Introduce ethical, legal, and professional issues in information security

COURSE OUTCOMES: On completion of this course, the students will be able to:
CO1: Explain security principles and threat models
CO2: Identify and analyze security risks and attacks
CO3: Apply basic cryptographic techniques
CO4: Understand network, system, and application security
CO5: Recognize legal, ethical, and compliance requirements

Syllabus

Unit I

Information Security: Definition, Need, and Objectives, CIA Triad: Confidentiality, Integrity, Availability, Information Security vs Cyber Security, Threats, Vulnerabilities, Attacks, and Risks
Security Models and Principles, Overview of Security Architecture

Unit II

Passive and Active Attacks, Insider vs Outsider Attacks, Malware:(Viruses, Worms, Trojans, Ransomware,

Spyware), Social Engineering Attacks, Phishing, Spoofing, and DoS/DDoS Attacks

Unit III

Introduction to Cryptography, Encryption and Decryption, Symmetric Key Cryptography (DES, AES – overview), Asymmetric Key Cryptography (RSA – overview), Hash Functions (MD5, SHA – basics), Digital Signatures, Key Management Concepts

Unit IV

Network Security Concepts, Firewalls: Types and Functions, Intrusion Detection and Prevention Systems (IDS/IPS), Secure Network Protocols (HTTPS, SSL/TLS – overview), Operating System Security, Authentication and Authorization, Access Control Models (DAC, MAC, RBAC)

Unit V

Web Application Vulnerabilities, SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Secure Software Development Lifecycle (SSDLC), Security Testing Basics

Suggested books

Textbooks

1. William Stallings, *Cryptography and Network Security*
2. Michael E. Whitman & Herbert J. Mattord, *Principles of Information Security*

Reference Books

1. Charles P. Pfleeger, *Security in Computing*
2. Nina Godbole, *Information Systems Security*

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2

CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.6	2.4	2	2	2	0	3	3	2	2.2	2	2.2

List of Practicals

1. Understand CIA Triad (Confidentiality, Integrity, Availability)
2. Password hashing using SHA algorithms
3. Implement encryption and decryption using: AES or DES (basic level)
4. Implement public-key cryptography using RSA
5. Implement hashing using:MD5 / SHA-256
6. Create and verify digital signatures
7. Capture and analyze network packets using Wireshark
8. Configure basic firewall rules (Linux / Windows)
9. Demonstration of Snort or similar IDS tools
10. Implement secure communication using SSL/TLS concepts

Course Outcomes

Course Name: Information Security Fundamentals Lab

At the end of the course, the students will be able to:

CO1: Demonstrate fundamental information security concepts, including the CIA triad, security principles, and threat models through practical experiments.

CO2: Identify, analyze, and simulate common security threats, vulnerabilities, and cyber-attacks such as malware, phishing, and DoS attacks.

CO3: Implement basic cryptographic techniques including symmetric encryption, hashing, and digital signatures using suitable tools or programming environments.

CO4: Configure and evaluate basic network, system, and application security mechanisms such as firewalls, authentication, and access control.

CO5: Examine legal, ethical, and compliance issues in information security and apply secure practices during laboratory activities.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.6	2.4	2	2	2	0	3	3	2	2.2	2	2.2

Course Name: Cryptography and Network Security	**Credit Scheme**				**Evaluation Components**			
Course Code:CSEC219	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	0	40	-	60	100
Prerequisite Course and code (if any):								

Course Objective:

1. To understand the fundamental concepts of cryptography and secure communication.
2. To study classical and modern encryption techniques and their applications.
3. To analyze authentication, integrity, and key management mechanisms.
4. To understand network security protocols and secure system design.
5. To expose students to real-world security threats, attacks, and countermeasures.

Course Outcomes:

1. Explain the principles and goals of cryptography and network security.
2. Apply symmetric and asymmetric cryptographic algorithms for secure data transmission.
3. Analyze cryptographic hash functions, digital signatures, and authentication protocols.
4. Understand and evaluate network security protocols and firewalls.
5. Identify common security attacks and design appropriate defense mechanisms.

Detailed Syllabus:

Unit I: Introduction to Cryptography and Security Concepts

Introduction to computer and network security. Security goals: confidentiality, integrity, authentication, authorization, and non-repudiation. Security attacks, services, and mechanisms. Classical cryptographic techniques: substitution ciphers, transposition ciphers, and product ciphers. Cryptanalysis and brute-force attacks. Overview of modern cryptography and network security architecture.

Unit II: Symmetric Key Cryptography

Principles of symmetric encryption. Stream ciphers and block ciphers. Data Encryption Standard (DES) algorithm and its structure. Triple DES. Advanced Encryption Standard (AES). Modes of operation of block ciphers. Key distribution and key management. Security of symmetric encryption algorithms.

Unit III: Asymmetric Key Cryptography and Hash Functions

Public key cryptography concepts and applications. RSA algorithm. Diffie–Hellman key exchange. Elliptic Curve Cryptography (ECC). Cryptographic hash functions and requirements. Secure Hash Algorithm (SHA) family. Message Authentication Codes (MAC). Digital signatures and their applications.

Unit IV: Authentication and Network Security Protocols

Authentication mechanisms and protocols. Kerberos authentication system. X.509 certificates and Public Key Infrastructure (PKI). Secure Socket Layer (SSL) and Transport Layer Security (TLS). Email security: Pretty Good Privacy (PGP) and S/MIME. IP Security (IPSec) architecture and protocols.

Unit V: System and Network Security

Firewall concepts and types. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). Malware threats: viruses, worms, Trojans, and ransomware. Network attacks: spoofing, sniffing, denial-of-service attacks. Web security issues and countermeasures. Security policies, risk management, and case studies in network security.

Suggestive Readings:

Text Books:

1. William Stallings, *Cryptography and Network Security: Principles and Practice*, Pearson Education.
2. Behrouz A. Forouzan, *Cryptography and Network Security*, McGraw-Hill Education.

Reference Books:

1. Douglas R. Stinson, *Cryptography: Theory and Practice*, CRC Press.

2. Charlie Kaufman, Radia Perlman, Mike Speciner, *Network Security: Private Communication in a Public World*, Pearson.

CO – PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	2	2	1	-	-	1	-	-
CO2	2	3	2	2	3	-	-	-	2	-	-
CO3	1	2	2		1	-	-	-	–	-	-
CO4	3	3	3	2	2	1	-	-	2	2	1
CO5	2	3	3	3	3	1	-	-	2	2	1
Avg	2.4	2.6	2.4	2.2	2.4	1	0	0	1.7	2	1

Course Name: Security and Privacy incloud	Credit Scheme				Evaluation Components			
Course Code: CSE C305	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	50	50	50	150
Prerequisite Course and code (if any):								

COURSE OBJECTIVES:

The objectives of the course are to

1. Understand the fundamentals of cloud security and cryptographic techniques.
2. Learn the architectural and design principles for securing cloud environments.
3. Analyze access control mechanisms and identity management solutions in cloud platforms.
4. Apply cloud security design patterns and best practices.
5. Develop strategies for monitoring, auditing, and managing security in cloud environments

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

CO1: Explain core security concepts such as confidentiality, integrity, authentication, and cryptography in the cloud**.**
CO2: Design secure cloud architectures addressing common threats and applying best practices in isolation and data protection
CO3: Implement identity and access control mechanisms including RBAC, MFA, and identity federation
CO4: Apply security design patterns to real-world cloud scenarios for robust security architecture
CO5: Perform proactive monitoring and auditing for incident response and compliance in cloud systems

Syllabus

Unit I

Overview of cloud security- Security Services - Confidentiality, Integrity, Authentication, Nonrepudiation, Access Control - Basic of cryptography - Conventional and public-key cryptography, hash functions, authentication, and digital signatures

Unit II

Security design principles for Cloud Computing - Comprehensive data protection - End-to-end access control - Common attack vectors and threats - Network and Storage - Secure Isolation Strategies - Virtualization strategies - Inter-tenant network segmentation strategies - Data Protection strategies: Data retention, deletion and archiving procedures for tenant data, Encryption, Data Redaction, Tokenization, Obfuscation, PKI and Key.

Unit III

Access control requirements for Cloud infrastructure - User Identification - Authentication and Authorization - Roles-based Access Control - Multi- factor authentication - Single Sign-on, Identity Federation - Identity providers and service consumers - Storage and network access control options - OS Hardening and minimization - Verified and measured boot - Intruder Detection and prevention.

Unit IV

Introduction to Design Patterns, Cloud bursting, Geo- tagging, Secure Cloud Interfaces, Cloud Resource Access Control, Secure On-Premise Internet Access, Secure External Cloud.

Unit V

Proactive activity monitoring - Incident Response, Monitoring for unauthorized access, malicious traffic, abuse of system privileges - Events and alerts - Auditing – Record generation, Reporting and Management, Tamper-proofing audit logs, Quality of Services, Secure Management, User management, Identity management, Security Information and Event Management

Suggested Book

Textbooks:

1. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance,: Tim Mather, Subra Kumaraswamy, and Shahed Latif, Publisher: O'Reilly Media, ISBN: 978-0596802769

Reference Book

1. Cloud Computing Security: Foundations and Challenges, Author: John R. Vacca, Publisher: CRC Press, ISBN: 978-1498782733

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.6	2.4	2	2	2	0	3	3	2	2.2	2	2.2

List of Practicals

1. Create and explore a basic cloud account (AWS / Azure / GCP – free tier)
2. Compare security and privacy issues in each model
3. Upload data to cloud storage (S3 / Blob / Cloud Storage)

4. Configure security groups / network rules
5. Configure virtual networks (VPC / VNet)
6. Encrypt and decrypt data using KMS
7. Enable cloud logging and monitoring services
8. Enable HTTPS using SSL/TLS certificates
9. Perform basic security assessment of cloud resources
10. Identify security and privacy failures

Course Outcomes

Course Name: Security and Privacy in Cloud Lab

At the end of the course, the students will be able to:

CO1: Demonstrate fundamental cloud security and privacy concepts, including cloud service models, deployment models, and shared responsibility frameworks.
CO2: Identify and analyze security threats, vulnerabilities, and risks specific to cloud computing environments.
CO3: Implement authentication, authorization, encryption, and key management mechanisms to secure cloud-based resources and data.
CO4: Configure and evaluate security controls for cloud infrastructure, applications, and storage using industry-standard cloud platforms and tools.
CO5: Examine privacy, compliance, and regulatory requirements in cloud environments and apply best practices for secure and privacy-preserving cloud deployments.

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.6	2.4	2	2	2	0	3	3	2	2.2	2	2.2

Course Name: Block Chain Technology	Credit Scheme				Evaluation Components			
Course Code: CSEC319	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	2	25	25	25	150
Prerequisite Course and code (if any):								

Course Objective:
The course aims to:

1. Provide fundamental understanding of blockchain concepts, structure, and cryptographic mechanisms used in distributed ledgers.
2. Explore blockchain protocols, consensus algorithms, and their role in ensuring security and trust.
3. Introduce various blockchain implementation frameworks, architectures, and development environments.
4. Examine real-world blockchain applications across multiple domains such as finance, logistics, and smart cities.
5. Develop hands-on experience with Hyperledger Fabric, Ethereum, and related blockchain development tools.

Course Outcomes:
On completion of this course, the students will be able to:
CO1: Understand the basics of Block chain
CO2: Learn Different protocols and consensus algorithms in Block chain
CO3: Learn the Blockchain implementation frameworks
CO4: Understand the Blockchain Applications
CO5: Experiment the Hyper ledger Fabric, Ethereum networks.

Detailed Syllabus:
Unit I
Introduction to Block chain: Block chain- Public Ledgers, Block chain as Public Ledgers - Block in a Block chain, Transactions The Chain and the Longest Chain - Permissioned Model of Block chain, Cryptographic -Hash Function, Properties of a hash function-Hash pointer and Merkle tree.

Unit II
Bit coin and Crypto currency: A basic crypto currency, Creation of coins, Payments and double spending,

FORTH – the precursor for Bit coin scripting, Bit coin Scripts, Bit coin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay.

Unit III

Bit coin Consensus: 6 Bit coin Consensus, Proof of Work (PoW)- Hash cash PoW, Bit coin PoW, Attacks on PoW, monopoly problem- Proof of Stake- Proof of Burn - Proof of Elapsed Time – Bitcoin Miner, Mining Difficulty, Mining Pool Permission model and use cases.

Unit IV

Hyper ledger Fabric & Ethereum: Architecture of Hyper ledger fabric v1.1- chain code- Ethereum: Ethereum network, EVM, Transaction fee, Mist Browser, Ether, Gas, Solidity.

Unit V

Block chain Applications: Smart contracts, Truffle Design and issue- D Apps- NFT. Block chain Applications in Supply Chain Management, Logistics, Smart Cities, Finance and Banking, Insurance, etc.- Case Study.

Suggested Books:

Text Books:

1. Christopher Kruegel, Fredrik Valeur, Giovanni Vigna: "Intrusion Detection and Correlation Challenges and Solutions", 1stEdition, Springer.
2. Carl End or f, Eugene Schultz and Jim Mellander "Intrusion Detection & Prevention", 1st Edition, Tata McGraw Hill.

Reference Books:

1. Stephen North cutt, Judy Novak: "Network Intrusion Detection", 3rd Edition, New Riders Publishing.
2. T. Fahringer, R. Prodan, "A Textbook on Grid Application Development and Computing Environment". 6th Edition, Khanna Publishers.

CO-PO Mapping

COs / POs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12
CO1	3	2	1	1	1	-	-	-	-	-	1	1
CO2	3	3	2	2	2	-	-	-	-	-	1	1
CO3	3	3	3	3	3	-	-	-	1	-	2	2
CO4	3	2	3	2	2	1	1	1	1	1	2	2
CO5	3	3	3	3	3	-	-	-	2	1	3	3
Average	**3.0**	**2.6**	**2.4**	**2.2**	**2.2**	**0.2**	**0.2**	**0.2**	**0.8**	**0.4**	**1.8**	**1.8**

List of Practicals:

1. Install and configure a blockchain development framework such as Ethereum or Hyperledger Fabric.
2. Set up a private blockchain network with multiple nodes using tools like Ganache or Geth.
3. Implement functions to store and retrieve data on the blockchain.
4. Test and debug the smart contract using unit testing frameworks like Truffle or Hyperledger Composer.
5. Implement a user interface for the DApp using web development technologies like HTML, CSS, and JavaScript.
6. Integrate the DApp with a blockchain network and demonstrate its functionality.
7. Explore different cryptographic algorithms used in blockchain technology, such as hashing, digital signatures, and encryption.
8. Implement cryptographic functions to secure transactions and data on the blockchain.
9. Analyze and propose security measures to protect against common blockchain vulnerabilities like double-spending and 51% attacks.
10. Simulate and compare different consensus algorithms like Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT).
11. Set up a private blockchain network with a specific consensus mechanism and validate its performance and fault tolerance.
12. Investigate the impact of consensus algorithms on the blockchain network's scalability and energy consumption.
13. Build a bridge between two different blockchain networks and facilitate cross-chain communication.
14. Test and evaluate the solution's effectiveness in terms of transparency, efficiency, and security.

15. Propose strategies to address governance and legal issues in blockchain implementations.

Course Name: Blockchain Technology Lab

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

CO1: Demonstrate understanding of fundamental blockchain concepts, including blocks, transactions, cryptographic hash functions, and Merkle trees.

CO2: Analyze and implement blockchain protocols and consensus mechanisms such as Proof of Work and Proof of Stake in a controlled lab environment.

CO3: Develop and deploy blockchain networks using implementation frameworks such as Hyperledger Fabric and Ethereum.

CO4: Design and execute smart contracts and decentralized applications (DApps) for real-world use cases.

CO5: Perform hands-on experimentation with blockchain development tools, wallets, miners, and test networks to evaluate performance, security, and scalability

CO-PO Mapping

COs / POs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	2	1	1	1	–	–	–	–	–	1	1
CO2	3	3	2	2	2	–	–	–	–	–	1	1
CO3	3	3	3	3	3	–	–	–	1	–	2	2
CO4	3	2	3	2	2	1	1	1	1	1	2	2
CO5	3	3	3	3	3	–	–	–	2	1	3	3
Average	**3**	**2.6**	**2.4**	**2.2**	**2.2**	**0.2**	**0.2**	**0.2**	**0.8**	**0.4**	**1.8**	**1.8**

Course Name: Ethical Hacking	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEC404	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	2	50	0	50	100
Prerequisite Course and code (if any):								

Course Objective:
The course aims to:

1. To introduce the fundamental concepts of ethical hacking and information security, including legal, ethical, and professional responsibilities.
2. To develop an understanding of hacking methodologies, threat models, and the stages of cyber attacks used by attackers and security professionals.
3. To familiarize students with various types of cyber threats, vulnerabilities, and exploits affecting computer systems, networks, and web applications.
4. To provide knowledge of tools and techniques used for reconnaissance, scanning, enumeration, and vulnerability assessment.
5. To enable learners to analyze security weaknesses and recommend appropriate countermeasures to protect systems and data.

Course Outcomes:

1. On completion of this course, the students will be able to:
2. Students will be able to explain the principles of ethical hacking, cybersecurity laws, and professional ethics in information security.
3. Students will be able to identify security vulnerabilities in computer systems, networks, and applications using standard assessment techniques.
4. Students will be able to apply ethical hacking tools and methodologies for reconnaissance, scanning, and vulnerability analysis.
5. Students will be able to analyze cyber-attack scenarios and propose suitable security controls and mitigation strategies.
6. Students will be able to demonstrate practical skills in securing systems and networks while adhering to

legal and ethical guidelines.

Detailed Syllabus:

Unit I: Introduction to Ethical Hacking and Cyber Security

Introduction to ethical hacking, need and scope of ethical hacking, hacking versus ethical hacking, types of hackers, information security concepts, confidentiality, integrity and availability, threat landscape, vulnerability and risk assessment, overview of cyber laws, IT Act, ethical and legal issues, professional responsibilities of an ethical hacker, penetration testing concepts and methodologies.

Unit II: Footprinting and Reconnaissance

Information gathering concepts, footprinting objectives, competitive intelligence, active and passive footprinting techniques, DNS footprinting, email footprinting, network footprinting, website footprinting, social engineering fundamentals, reconnaissance tools and techniques, countermeasures for footprinting and reconnaissance.

Unit III: Scanning, Enumeration and Vulnerability Analysis

Network scanning concepts, types of scanning, port scanning techniques, network mapping, OS fingerprinting, enumeration of services and users, vulnerability assessment concepts, vulnerability scanning tools, identification of system weaknesses, analyzing scan results, security assessment and mitigation strategies.

Unit IV: System Hacking, Malware and Web Application Attacks

System hacking methodology, password cracking techniques, privilege escalation, covering tracks, malware concepts, viruses, worms, trojans, ransomware, spyware, rootkits, malware analysis basics, web application architecture, common web vulnerabilities such as SQL injection, cross-site scripting, session hijacking, security flaws and preventive measures.

Unit V: Network Security, Wireless Hacking and Countermeasures

Network security threats and attacks, sniffing techniques, spoofing attacks, denial of service attacks, wireless network security, Wi-Fi encryption standards, wireless hacking techniques, intrusion detection and prevention systems, firewalls, cryptography basics, security policies, incident response, future trends in ethical hacking and cyber security.

Suggested Books:

Text Books:

1. Penetration Testing: A Hands-On Introduction to Hacking — Georgia Weidman
2. The Hacker Playbook 3: Practical Guide to Penetration Testing — Peter Kim

Reference Books:

1. EC-Council – Ethical Hacking and Countermeasures Attack Phases
2. Michael Simpson, Kent Backman & James Corley – Hands-On Ethical Hacking and Network Defense

CO-PO Mapping

COs / POs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	2	2	2	1	1	-	-	-	-	-	2	1
CO2	1	3	2	2	2	-	-	-	-	-	1	1
CO3	2	3	1	3	3	-	-	-	1	-	2	2
CO4	2	2	3	2	1	1	1	1	1	1	2	2
CO5	3	3	3	3	3	-	-	-	2	1	3	3
Avg	2	2.6	2.4	2.2	2	1	1	1	1.3	1	2	1.8

DEPARTMENT ELECTIVE

Course Name: Data Analytics and Visualization	**Credit Scheme**				**Evaluation Components**			
Course Code: CSED319	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Introduce students to the concepts, processes, and lifecycle of **data analytics** and its role in decision-making. Enable students to understand different types of data (structured, semi-structured, and unstructured) and apply appropriate **data preprocessing techniques**.
2. Familiarize learners with **statistical, regression, classification, and clustering methods** for analyzing diverse datasets.
3. Provide exposure to stream data mining techniques and real-time analytics applications.Develop skills in using modern data analytics frameworks and tools such as Hadoop, MapReduce, Hive, and NoSQL databases
4. Train students to apply **R programming** for performing data analysis, modeling, and visualization.
5. Cultivate the ability to create effective **visual representations** of data to communicate insights clearly and meaningfully.

Course Outcomes: On completion of this course, the students will be able to:
CO1:Discuss various concepts of data analytics pipeline
CO2: Apply classification and regression techniques
CO3:Explain and apply mining techniques on streaming data
CO4:Describe the concept of R programming and Implement analytics on Big data using R.
CO5:Create and interpret effective data visualizations using appropriate tools and techniques to communicate analytical results for decision-making.

Syllabus:

Unit 1:
Introduction to Data Analytics: Sources & nature of data, Classification of data: Structured, Semi-structured, Unstructured,Characteristics of data & Big Data platform,Need of Data Analytics & evolution of scalability,Data Analytics Process & Tools,Analysis vs Reporting,Modern tools & Applications of Data Analytics,Data Analytics Lifecycle & Phases,Discovery, Data Preparation, Model Planning,Model Building, Results Communication, Operationalization,Modern Data Analytics Tools (Tableau, Power BI, Google Data Studio)

Unit II:
Regression Modeling,Multivariate & Bayesian Analysis,Bayesian Networks & Inference,Support Vector & Kernel Methods,Time Series Analysis: Linear vs Nonlinear,Rule Induction & Neural Networks Basics,Neural Networks: Learning & Generalisation,PCA and Competitive Learning,Fuzzy Logic & Decision Trees,Stochastic Search Methods,Applications of Bayesian Networks in Real-World (Healthcare, Fraud Detection)

Unit III:
Introduction to Data Streams & Models,Stream Computing Architecture,Sampling & Filtering Data in Streams,Counting Distinct Elements in Streams,Estimating Moments & Decaying Windows,Real-time Analytics Platforms,Case Study: Sentiment Analysis,Case Study: Stock Market Prediction,Streaming Applications in Business,Emerging Trends in Streaming Data Analytics,Case Studies on Real-time Analytics (e.g., Twitter Sentiment, IoT Streaming)

Unit IV:
Mining Frequent Itemsets, Apriori Algorithm,Handling Large Data Sets,Counting Frequent Itemsets in Streams, Clustering Techniques: Hierarchical,K-Means Clustering & Visualization,Clustering High Dimensional Data,Advanced Clustering: CLIQUE & ProCLUS,Clustering for Streams,Parallelism in Clustering,Advanced Clustering Methods (DBSCAN, OPTICS),

Unit V:
Frameworks: MapReduce & Hadoop,Hive, Pig, HBase, NoSQL,Data Visualization Techniques,Interaction Techniques & Systems,R for Data Analytics: GUI, Import/Export, Visualization,NoSQL Databases (MongoDB, Cassandra) and Visualization with R Shiny

Suggested Books:

Text Books:

1. Michael Minelli, Michehe Chambers, “Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Business”, 1st Edition, AmbigaDhiraj, Wiely CIO Series, 2013
2. ArvindSathi, “Big Data Analytics: Disruptive Technologies for Changing the Game”, 1st Edition, IBM Corporation, 2012.1. Rajaraman, A., Ullman, J. D., Mining of Massive Datasets, Cambridge University Press, United Kingdom, 2012

Reference Books:

1. Berman, J.J., Principles of Big Data: Preparing, Sharing and Analyzing Complex Information, Morgan Kaufmann, 2014
2. Barlow, M., Real-Time Big Data Analytics: Emerging Architecture, O Reilly, 2013

CO-PO Mapping

COs	PO1	PO2	PO 3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.2	2	2	0	3	3	2	2.2	2	2.2

List of Practicals:

1.To get the input from user and perform numerical operation (MAX, MIN, AVG, SUM, SQRT, Round)using in R

2 To Perform data import/export (.CSV, .CLs, .TXT) operations using data frames in R

3 To get the input matrix from user and perform matrix addition, subtraction, multiplixation, inverse transpose and division operations using vector concept in R

4 To perform statistical operations (Mean, Medium, mode, and Standard deviation) using R

5 To perform data pre-processing operations i) Handling Missing data ii) Min-Max normalization

6 To Perform dimensionality reduction operation using PCA for houses data set

7 To Perform simple linear regression with R

8 To Perform K- Mean clustering operation and visualize for iris data set

9 Learn how to collect data via web-scraping, APIs and data connectors from suitable sources as specified by the instructor

10 Perform association analysis on a given data set and evaluate its accuracy

11 Build a recommendation system on a given data set and evaluate its accuracy
12 Build a time- series model on a given data set and evaluate its accuracy. involving various countries of the world states
13 Build cartographic visualization for multiple data set involving various and districts in india etc.
14 Perform text mining on a set of documents and visualize the most important words in a visualization such as word cloud

Course Outcomes

Course Name: Data Analytics and Visualization Lab

At the end of the course, the students will be able to:

CO1: Collect, clean, and preprocess structured and unstructured datasets using appropriate data analytics tools and techniques.

CO2: Perform exploratory data analysis (EDA) using statistical methods to identify patterns, trends, and relationships in data.

CO3: Apply data analytics techniques to analyze real-world datasets and extract meaningful insights for decision-making.

CO4: Design and develop effective data visualizations using charts, graphs, and dashboards to communicate analytical results clearly.

CO5: Interpret, evaluate, and present analytical findings using visualization tools and reports for technical and non-technical audiences.

CO-PO Mapping

COs	PO1	PO2	PO 3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	– –	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2		
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.2	2	2	0	3	3	2	2.2	2	2.2

Course Name: Cloud Computing	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE017	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	40	0	60	100
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. To introduce the fundamental concepts, characteristics, and evolution of cloud computing and explain its significance in modern IT infrastructure.
2. To familiarize students with various **cloud service models (SaaS, PaaS, IaaS)** and **deployment models (Public, Private, Hybrid, Community)**, along with their advantages and limitations.
3. To develop an understanding of **datacenter operations, resource management, virtualization, and cloud simulators**, highlighting efficiency, scalability, and cost trade-offs.
4. To expose students to **cloud platforms and tools** such as CloudSim, Green Cloud, VMware, Google Cloud, and Microsoft Azure for hands-on learning.
5. To enable students to design, implement, and deploy simple cloud-based applications and services, while addressing issues of security, compliance, and performance.

Course Outcomes: On completion of this course, the students will be able to:

1. Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing
2. Apply the fundamental concepts in datacenters to understand the tradeoffs in power, efficiency and cost.
3. Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.
4. Analyze various cloud programming models and apply them to solve problems on the cloud.
5. Utilize cloud platforms, simulators, and virtualization tools to design, implement, and deploy applications in the cloud environment.

Syllabus:

Unit I

Cloud Computing Fundamentals and Architecture: Definition and evolution of cloud computing, characteristics and benefits, challenges and risks, cloud service models (IaaS, PaaS, SaaS, FaaS), cloud deployment models (Public, Private, Hybrid, Multi-cloud), overview of cloud infrastructure components (VMs, storage, networking).

Unit II

Virtualization and Cloud Infrastructure: Introduction to virtualization, types of virtualization (hardware, OS, server, network), hypervisors, role of virtualization in cloud computing, virtual machines and containers, configuration and management of cloud infrastructure, scalability, load balancing, high availability, and disaster recovery.

Unit III

Cloud Platforms and Services: Overview of popular cloud platforms (AWS, Azure, GCP), key services (compute, storage, networking, databases), hands-on with platform-specific tools, cloud storage types (object, block, file), cloud databases (relational and NoSQL), introduction to cloud application deployment and architecture.

Unit IV

Cloud Security and Identity Management: Security threats in the cloud, shared responsibility model, data security (encryption, secure storage, compliance), Identity and Access Management (IAM), user roles, authentication and authorization, network security (firewalls, VPNs), secure cloud access and policies.

Unit V

Cloud Automation, Incident Response, and Emerging Trends: Automation tools (Terraform, Ansible, CloudFormation), orchestration and DevOps practices in cloud, incident detection and response planning, best practices for cloud monitoring and recovery, introduction to serverless computing, containerization (Docker, Kubernetes), AI/ML applications on cloud platforms.

Suggested Books

Text Book

1. Rajkumar Buyya, Christian Vecchiola, and S. Thamarai Selvi, *Mastering Cloud Computing: Foundations and Applications Programming*, McGraw Hill Education, 2013.
2. Toby Velte, Anthony Velte, and Robert Elsenpeter, *Cloud Computing: A Practical Approach*, McGraw Hill Education, 2009.

Reference Books

1. Gautam Shroff, *Enterprise Cloud Computing: Technology, Architecture, Applications*, Cambridge University Press, 2010.
2. Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Pearson Education, 2014.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.1	2	2	0	3	3	2	2.2	2	2.2

Course Name: Statistical Machine Learning & Modeling	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE D305	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	40	0	60	100
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Understand the statistical foundations of machine learning, including probability theory, random variables, statistical inference, and estimation techniques.
2. Develop and apply statistical models for supervised and unsupervised learning problems such as regression, classification, clustering, and dimensionality reduction.
3. Analyze and interpret data using statistical learning methods, emphasizing model assumptions, bias–variance trade-off, and uncertainty quantification.
4. Select, train, and evaluate machine learning models using appropriate performance metrics, cross-validation, and model selection techniques.
5. Implement statistical machine learning algorithms using programming tools (such as Python or R) and apply them to real-world datasets.

Course Outcomes: On completion of this course, the students will be able to:
CO1: Understand the difference between supervised and unsupervised learning
CO2: Apply the common probability distributions in machine learning applications
CO3: Implement the fundamental learning algorithms such as logistic regression and k-means clustering
CO4: Apply the maximum likelihood estimation (MLE) for parameter estimation.
CO5: Understand technique of deep learning architectures.

Syllabus

UnitI
Introduction: Overview of machine learning, Styles of machine learning – supervised learning and unsupervised learning, Data representations- numerical and graphical representation, Applications of machine learning-Recognizing examples, Familiar applications, Emerging applications.

Unit II
Probability- Discrete Random Variables, Probability Mass Function (PMF), Common Distributions of

PMF - Uniform, Binomial, Joint Probability Mass Function, Conditional Probability, Relationship Between Marginal and Joint Probability, Bayes Theorem, Independent Random Variables, Continuous Random Variables Probability Density Function (PDF), Common Distributions of PDF - Normal ,Beta, Joint Probability Density Function, Moments of Random Variables Module Maximum Likelihood Estimation-> Likelihood function - For discrete probability distribution, For continuous probability distribution Maximum likelihood estimation_> For discrete probability distribution, For continuous probability distribution, For mean and standard deviation.

Unit III

Generative vs Discriminative models for supervised learning->Essential distinction, Generative Model-Naive Bayes, Discriminative Model-Logistic Regression. Naive Bayes->Naive Bayes Assumption, Decision Rule Parameters of Naive Bayes, Maximum Likelihood Estimation (MLE) for Naive Bayes Parameters, Text Classification using Naive Bayes, Bag of Words Model for Text. Logistic Regression->Logistic Function, Linear Classifier, Parameter Estimation, Maximizing Conditional Log Likelihood Gradient Ascent, Optimization Algorithm, Comparing the Models Empirical Comparison of Naive Bayes and Logistic Regression. Support Vector Machine (SVM)-> separable and non-separable, Cross Validation.

Unit IV

Introduction to Clustering, Clustering Methods: Partitioned based Clustering- K-means, K-medoids; Hierarchical Clustering: Agglomerative, Divisive, Distance measures; Density based Clustering - DB Scan; Spectral clustering, Dimensionality Reduction, Principle Component Analysis (PCA).

Unit V

Introduction to artificial neural network and deep learning, Early models of artificial neural network and their learning algorithms, Deep learning: what it is and what it is not, Key Techniques Enabling Deep Learning Back-propagation algorithm for learning, Choice of activation functions, A few regularization methods, Some Basic Deep Architecture Convolutional Neural Network, Recurrent Neural Networks and Autoencoders.

Suggested Books

Text Book

1.K. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press.

2. 2.E. Alpaydin, “Machine Learning”, MIT Press.

Reference Books

1. John Mueller and Luca Massaron, “Machine Learning for Dummies “, John Wiley & Sons.
2. Shai Shalev-Shwartz, Shai Ben-David, “Understanding Machine Learning: From

Theory to Algorithms", Cambridge University Press.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.1	2	2	0	3	3	2	2.2	2	2.2

Course Name: IOT Architectures & Protocols	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEI205	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	3	3	0	0	40	0	60	100
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Understand the **fundamental concepts of the Internet of Things (IoT)** and its real-world applications
2. Explain the **layers of IoT architecture** (device, gateway, data management, application) and their functions
3. Identify and compare **IoT communication models** (device-to-device, device-to-cloud, gateway-based)
4. Understand commonly used **IoT protocols** such as MQTT, CoAP, HTTP, AMQP, and DDS
5. Analyze the role of **networking technologies** used in IoT (Wi-Fi, Bluetooth, Zigbee, LoRaWAN, cellular)

Course Outcomes: On completion of this course, the students will be able to:
CO1: Understand the concepts of Internet of Things.
CO2: Analyze basic protocols in wireless sensor network.
CO3: Design IoT applications in different domain and be able to analyze their performance.
CO4: Implement basic IoT applications on embedded platform
CO5: Evaluate security, privacy, and interoperability issues in IoT architectures and protocols.

Syllabus

UnitI

Overview and Introduction: Internet of Things (IoT) and Web of Things (WoT): What's WoT?, The Internet of Things Today, Time for Convergence, Towards the IoT Universe, Internet, of Things Vision, IoT Strategic Research and Innovation Directions, IoT Applications, Future Internet Technologies, Infrastructure, Networks and Communication, Processes, Data Management, Security, Privacy & Trust, Device Level Energy Issues, IoT Related Standardization, Recommendations on Research Topics.

UnitII

M2M to IoT A Basic Perspective: Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven, global value chain and global information monopolies. M2M to IoT-An Architectural Overview: Building architecture, Main design, principles and needed capabilities, An IoT architecture outline, standards considerations.

Unit III

IoT Architecture -State of the Art: Introduction, State of the art, Architecture Reference Model-Introduction, Reference Model and architecture, IoT reference Model, IoT Reference Architecture-Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views.

Unit IV

IoT Applications for Value Creations: Introduction, IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications, Four Aspects in your Business to Master IoT, Value, Creation from Big Data and Serialization, IoT for Retailing Industry, IoT For Oil and Gas, Industry, Opinions on IoT Application and Value for Industry, Home Management, eHealth.

Unit V

Internet of Things Privacy, Security and Governance: Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, Privacy and Trust in IoTData-Platforms for Smart Cities, First Steps Towards a Secure Platform, Smartie Approach.

Suggested Books

Text Book

1. Jan Holler, Vlasios Tsiatsis, et.al., Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st Edition, Academic Press, 2014.

Reference Books

1. ijay Madisetti and Arshdeep Bahga, Internet of Things (A Hands-on-Approach), 1st Edition, VPT, 2014.
2. Francis daCosta, Rethinking the Internet of Things: A Scalable Approach to Connecting Everything, 1st Edition, Apress Publications, 2013.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.1	2	2	0	3	3	2	2.2	2	2.2

Course Name: Software Development in IOT	Credit Scheme				Evaluation Components			
Course Code: CSEI305	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	50	50	50	150
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. To understand the **fundamental concepts of IoT** and the role of software in IoT systems
2. To learn **programming techniques for IoT devices** using embedded and high-level languages
3. To understand **IoT communication models and protocols** used in software development
4. To develop skills for building **end-to-end IoT applications**, from device to cloud
5. To gain knowledge of **IoT middleware and cloud platforms** for data storage and application services

Course Outcomes: On completion of this course, the students will be able to:
CO1: Explain the fundamentals of IoT systems and the role of software in IoT applications.
CO2: Develop embedded software to interface sensors and actuators with IoT devices
CO3: Implement IoT communication protocols (such as MQTT, HTTP/REST, CoAP) for data exchange.
CO4: Design and develop end-to-end IoT applications integrating devices, gateways, and cloud platforms.
CO5: Create web or mobile applications for monitoring and controlling IoT systems.

Syllabus

Unit I
Overview of Internet of Things (IoT), IoT characteristics and applications, IoT system components, IoT software stack and architecture, Role of software in IoT systems, Embedded software vs application software in IoT

Unit II
IoT hardware platforms (Arduino, Raspberry Pi, ESP32 – overview), Sensors and actuators interfacing, Embedded programming concepts,Programming using C/C++ / Python for IoT devices, Device drivers and firmware basics, Power management in IoT software

Unit III
IoT communication models, Network layers in IoT, Application layer protocols: MQTT, CoAP, HTTP/REST, Data formats: JSON, XML, Message handling and data transmission

Unit IV
Role of middleware in IoT, IoT gateways and data aggregation, Cloud platforms for IoT (AWS IoT, Azure IoT, Google IoT – overview), Device management and monitoring, Data storage and visualization, APIs and web services for IoT applications

Unit V
End-to-end IoT application development, Mobile and web applications for IoT, Real-time data processing, Event handling and automation, IoT security challenges, Authentication and authorization, Secure communication (TLS/SSL – overview), Software testing for IoT applications

Suggested Books
Text Book

1. Arshdeep Bahga & Vijay Madisetti – *Internet of Things: A Hands-On Approach*

Reference Books

1. **Raj Kamal – *Internet of Things: Architecture and Design Principles***
2. **Adrian McEwen & Hakim Cassimally – *Designing the Internet of Things***

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3
Avg	2.8	2.4	2.1	2	2	0	3	3	2	2.2	2	2.2

Course Name: Dot Net Framework & C# Programming	Credit Scheme				Evaluation Components			
Course Code: CSE 318A	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	50	50	50	150
Prerequisite Course and code (if any):								

Course Objectives

1. To introduce the fundamentals of the .NET framework and its architecture
2. To develop applications using C# and .NET libraries
3. To understand object-oriented programming using C#
4. To build Windows and web-based applications
5. To gain exposure to data access using ADO.NET

Course Outcomes

On completion of this course, students will be able to:

- **CO1:** Explain the architecture and components of the .NET framework
- **CO2:** Develop programs using C# language fundamentals
- **CO3:** Apply object-oriented concepts in C# application development
- **CO4:** Build Windows and web-based applications using .NET
- **CO5:** Implement database-driven applications using ADO.NET

Detailed Syllabus

Unit I – Introduction to .NET Framework

Evolution of .NET ,.NET Framework architecture, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS),Managed vs unmanaged code,Just-In-Time (JIT) compilation, .NET languages and platform support
Introduction to Visual Studio IDE

Unit II – C# Language Fundamentals

C# program structure, Data types, variables, constants, Operators and expressions, Control statements: if, switch, loops, Arrays and strings, Type casting and conversions, Command-line arguments

Unit III – Object-Oriented Programming in C#

Classes and objects, Constructors and destructors, Properties and methods, Inheritance and polymorphism, Method overloading and overriding, Interfaces and abstract classes, Encapsulation and access modifiers, Namespace and assemblies

Unit IV – Advanced C# Concepts

Exception handling, Delegates and events, Generics, Collections (ArrayList, List, Dictionary, Stack, Queue), File handling and streams, Multithreading and task-based programming, LINQ basics

Unit V – Windows and Web Application Development

Windows Forms: controls, events, layout, Introduction to WPF, ASP.NET fundamentals, ASP.NET Web Forms / MVC architecture (overview), State management, Server controls and validation, Introduction to Web Services, ADO.NET architecture, Connected and disconnected architecture, DataReader, DataSet, DataAdapter., CRUD operations, Transaction management, Introduction to Entity Framework

Suggested Books

Text Books

1. Andrew Troelsen, *Pro C# and .NET*, Apress

2. Herbert Schildt, *C# 4.0: The Complete Reference*, McGraw-Hill

Reference Books

1. Jesse Liberty, *Programming C#*, O'Reilly

2. Microsoft Docs – C# and .NET Framework

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	1	–	2
CO2	3	3	2	2	–	–	–	–	–	1	–	2
CO3	3	3	3	2	2	–	–	–	–	1	–	2
CO4	3	2	3	3	2	–	–	–	1	1	–	2
CO5	3	3	3	3	3	–	–	–	1	1	1	3
Avg	3	2.6	2.4	2.2	1.8	–	–	–	0.4	1	0.2	2.2

List of practicals:

1. First Console Application (Hello World) in C#
2. Program to demonstrate Convert function
3. Program to demonstrate Boolean Operators
4. Write a program to take two numbers from user and determine the largest one?
5. Write a program to calculate simple interest
6. Draw Mandelbrot sets using for loop and switch case
7. Write a program using enum
8. Write a program using struts
9. Write a program using string arrays
10. Write a program using properties
11. Write a program for single level inheritance
12. Write a program for polymorphism
13. Write a program for method overriding

14. Write a program of single cast delegate
15. Write a program of multicast delegate
16. Write a program of using event
17. Write a program of exception handling
18. Write a program of multithreading
19. Write a program of indexers
20. Write a program of creating a assembly
21. Write a program of unsafe code
22. Write a program of boxing
23. Write a program of unboxing
24. Write a program of Arraylist Collection
25. Write a program of HashTable collection
26. Write a program of reading and writing to a data file

List of Practicals

1. Basic C# programs using control statements
2. Programs demonstrating OOP concepts
3. Exception handling and file operations
4. Collection and LINQ programs
5. Windows Forms application development
6. ASP.NET web application
7. Database connectivity using ADO.NET
8. Mini project using C# and .NET

Course Name: Dot Net Framework & C# Programming Lab

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

CO1: Explain the architecture, components, and execution model of the .NET Framework including CLR, CTS, and CLS.

CO2: Develop C# programs using language fundamentals such as data types, control structures, arrays, strings, and exception handling.

CO3: Apply object-oriented programming concepts including inheritance, polymorphism, interfaces, and

abstraction in C# applications.

CO4: Design and implement Windows-based and web-based applications using .NET technologies and Visual Studio tools.

CO5: Create database-driven applications using ADO.NET for data access, manipulation, and transaction management.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	1	–	2
CO2	3	3	2	2	–	–	–	–	–	1	–	2
CO3	3	3	3	2	2	–	–	–	–	1	–	2
CO4	3	2	3	3	2	–	–	–	1	1	–	2
CO5	3	3	3	3	3	–	–	–	1	1	1	3
Avg	3	2.6	2.4	2.2	1.8	–	–	–	0.4	1	0.2	2.2

Course Name: Digital Marketing	**Credit Scheme**				**Evaluation Components**			
Course Code: CSEA306	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objective:

1. To introduce the fundamentals of digital marketing, its evolution, and its significance in comparison with traditional marketing.
2. To familiarize students with digital consumers, their behavior, and the strategic frameworks (such as POEM) used to engage them effectively.
3. To equip students with knowledge of various digital marketing techniques, including SEO, PPC, social media, affiliate, email, and mobile marketing.
4. To develop an understanding of advanced advertising concepts, such as display advertising, programmatic buying, and digital marketing analytics.
5. To enable students to design, implement, and evaluate digital marketing strategies and campaigns using tools like Google Ads, Google Analytics, CRM platforms, and social media channels.

Course Outcomes:

Towards the end of the course, the students will be able to:
CO1: Understand the importance of digital marketing for marketing success.
CO2: Learn about the customer relationships across all digital channels and build better customer relationships.
CO3: Create a digital marketing plan starting from the SWOT (Strength, Weaknesses, Opportunities and Threats) Analysis and defining the target group.
CO4: Identify digital channels, their advantages and limitations.
CO5: Perceive ways of their integration taking into consideration the available budget.

Detailed Syllabus:

Unit I **7 Hours**

Fundamentals of Digital Marketing, Significance of Digital Marketing, Traditional Marketing vs Digital Marketing, Evolution of Digital Marketing, Digital Marketing Landscape, Key Drivers of Digital Marketing, Digital Consumer & Communities, Gen Y & Netizen Expectations, Influence of Digital Media on Consumers, Case Study.

Unit II **7 Hours**

Digital Users in India, Digital Marketing Strategy, Consumer Decision Journey, POEM Framework – Overview, Paid Media, Owned & Earned Media, Segmenting & Customizing Messages, Digital Advertising Market in India, Skills in Digital Marketing, Digital Marketing Plan (Case Study).

Unit III **6 Hours**

Techniques of Digital Marketing, PPC and Online Marketing, Social Media Marketing (Overview), SEO Techniques – Basics, Keyword Advertising, Google Webmaster Tools Overview, Google Analytics Overview, Affiliate Marketing, Email Marketing, Mobile Marketing.

Unit IV **8 Hours**

Display Advertising: Introduction, Display Advertising Buying Models, Advertising Tools & Platforms, Display Advertising Terminology, Types of Display Ads, Different Ad Formats, Ad Placement Techniques, Important Ad Terminology, Programmatic Advertising: Overview, Case Study on Programmatic Ads.

Unit V **8 Hours**

SEO Optimization – Writing SEO Content, Google AdWords – Creating Accounts, Google AdWords – Types, Introduction to CRM, CRM Platforms & Models, Creating a Facebook Page, Business Tools on LinkedIn, Creating Campaigns on LinkedIn, YouTube Business Accounts & Ads, YouTube Analytics & Course Revision.

Suggestive Readings:

Text Books:

1. Quick win Digital Marketing, H. Annmarie, A. Joanna, Paperback edition
2. Digital Marketing, Swaminathan T. N./Karthik Kumar, Cengage Learning India Pvt. Ltd.

Reference Books:

1. Marketing Strategies for Engaging the Digital Generation, D. Ryan
2. Digital Marketing, Seema Gupta, McGraw-Hill

CO-PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	Po12
CO1	3	2	2	1	2	1	-	-	2	2	-	2
CO2	2	2	2	1	2	2	1	1	3	3	-	3
CO3	2	3	3	2	2	2	-	-	3	3	2	3
CO4	2	2	2	2	3	-	-	-	2	3	2	3
CO5	2	2	3	2	3	2	-	-	3	3	2	3

List of Practicals:

S. No.	Practical Title	Aim of the Practical
1	Website Analysis	To analyze an existing website for SEO, usability, and performance using online tools.
2	Keyword Research	To perform keyword research using tools like Google Keyword Planner and identify high-ranking keywords.
3	On-Page SEO Optimization	To optimize meta tags, headings, content, and images for search engine ranking.
4	Off-Page SEO Techniques	To practice backlink creation, directory submission, and social bookmarking strategies.
5	Google Analytics Setup	To create and configure a Google Analytics account and track website performance metrics.
6	Google Search Console	To monitor website indexing, search queries, and technical SEO performance.
7	Social Media Marketing Campaign	To design and schedule posts for platforms like LinkedIn, Instagram, or Facebook.
8	Content Marketing Plan	To develop a content calendar and blog strategy for brand promotion.
9	Email Marketing Campaign	To design and execute an email marketing campaign using tools like Mailchimp.
10	Pay-Per-Click (PPC) Campaign	To create and analyze a Google Ads campaign.
11	Social Media Advertisement	To design and run a paid advertisement campaign on social media platforms.
12	Affiliate Marketing Model	To study and design a basic affiliate marketing strategy.
13	Online Branding Strategy	To develop a digital branding strategy for a product/service.
14	Influencer Marketing Case Study	To analyze influencer marketing strategies used by brands.
15	Digital Marketing Audit Report	To prepare a complete digital marketing audit report of a selected company.
16	Mini Project	To develop and execute a complete digital marketing strategy for a real or hypothetical product.

Course Outcomes (COs)

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

CO1: Understand fundamental concepts and tools of digital marketing and online branding.
CO2: Apply SEO, SEM, and social media strategies to improve online visibility.
CO3: Design and execute digital marketing campaigns using analytics tools.
CO4: Analyze website and campaign performance using digital analytics.

CO5: Develop comprehensive digital marketing strategies for business growth and entrepreneurship.

CO-PO Mapping:

PO → CO ↓	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	Po12
CO1	2	1	1	–	3	–	–	–	1	2	–	2
CO2	2	3	2	1	3	–	–	–	1	2	1	2
CO3	2	3	3	2	3	1	–	–	2	3	2	3
CO4	2	3	2	3	3	–	–	–	1	2	1	3
CO5	1	2	3	1	2	2	1	1	2	2	3	3

Course Name: Web Technology	**Credit Scheme**				**Evaluation Components**			
Course Code: CSE020	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2	40	50	60	150
Prerequisite Course and code (if any):								

Course Objective

1. To introduce the fundamentals of the Internet, and the principles of web design.
2. To construct basic websites using HTML and Cascading Style Sheets.
3. To build dynamic web pages with validation using Java Script objects and by applying different event handling mechanisms.
4. To develop modern interactive web applications using Client Side and Server Side.
5. To design the static website using HTML, Javascript and CSS.

Course Outcome:

CO1: Apply the knowledge of the internet and client side and server side concepts in understanding and developing various applications.
CO2: Analyze and develop static interactive web pages using HTML and CSS.
CO3: Design forms and check for data accuracy.
CO4: Understand, analyze and develop event-driven programs using Client side programming like JavaScript.
CO5: Analyze and develop database connectivity programs. Understand, analyze and develop web applications using Server side programming.

Detailed Syllabus:

Unit I **10 Hours**

Information Files, Creation, Web Server, Web Client, Web Browser, Understanding how a Browser communicates with a Web Server, Hyper Text Markup Language (HTML), HTML Tags (Paired Tags and Singular Tags), Common Commands, Structure of an HTML Document, Document Body, Titles and Footers, Text Formatting (Paragraph Breaks, Line Breaks), Emphasizing Material, and Drawing Lines.

Unit II **13 Hours**

Text Styles, Underline, Other Text Effects, Centering (Text and Images), Lists and List Tags, Types of Lists, Unordered Lists (Bullets), Ordered Lists (Numbering), Definition Lists, Adding Graphics to HTML Documents using Border, Width, Height, Align, and ALT attributes, and Tables—Introduction and use of Width and Border attributes.

Unit III **7 Hours**

Table attributes: Cell Padding, Cell Spacing, BGCOLOR, COLSPAN, and ROWSPAN. Introduction to JavaScript, Java Scripting for Web Pages, Database Connectivity, Client-side JavaScript, Capturing User Input, Advantages of JavaScript, and Basic Programming Techniques (Data Types and Literals).

Unit IV **7 Hours**

Typecasting, Creating Variables, Incorporating Variables in a Script, JavaScript Arrays and their Elements, Array Properties, Operators and Expressions, JavaScript Programming Constructs, Conditional Checking (If-then-else, Immediate If), Loops (For Loop), Functions in JavaScript (Built-in and User Defined), Declaring Functions, and Place of Declaration.

Unit V **9 Hours**

Passing Parameters, Variable Scope, Return Values, Recursive Functions, Dialog Boxes (Alert, Prompt, Confirm), JavaScript Document Object Model (DOM), JavaScript Assisted Style Sheets DOM (JSSS DOM), Understanding HTML Objects (Properties and Methods), Browser Objects (HTML Object Hierarchy and Access), Manipulating Web Page Elements, Handling Events using JavaScript Event Handlers, and Forms—Form Object, Form Methods, Text Element, Password Element, Button, Submit, Reset, Checkbox, Radio, Text Area, Select and Option Elements, and Multi-Choice Select Lists.

List of Practicals:

1. Create a basic HTML webpage containing headings, paragraphs, line breaks, and horizontal lines.
2. Design a webpage using text formatting tags (bold, italic, underline, subscript, superscript, font color, size).
3. Create an HTML page using lists — ordered, unordered, and definition lists.
4. Develop an HTML page that demonstrates image insertion using different attributes (border, height, width, align, alt).
5. Design a web page using tables with attributes like cell padding, cell spacing, bgcolor, colspan, and rowspan.
6. Create a webpage with hyperlinks (internal, external, and image links).
7. Design a webpage using frames or iframes to divide the browser window into multiple sections.
8. Create an HTML form using different input controls — text, password, radio buttons, checkboxes, select lists, and buttons.
9. Implement client-side validation in forms using JavaScript (e.g., check empty fields, password match, valid email).
10. Write a JavaScript program to perform arithmetic operations (addition, subtraction, multiplication, division) using user input.
11. Develop a JavaScript program using conditional statements to check the greatest of three numbers.

12. Implement loops in JavaScript to display prime numbers or factorial of a number.
13. Write a JavaScript program to demonstrate array operations (insertion, deletion, searching, sorting).
14. Create a webpage that changes background color and text color dynamically using JavaScript functions and events.
15. Develop a mini web project combining HTML and JavaScript — for example, a simple student registration form with validation and formatted output.

Suggested Books:

Text Books:

1. "Web Enable Commercial Application Using HTML, DHTML", Ivan Bay Ross- BPB Publication

References:

1. "Web Technology: A developer perspective", Gopalan & Akilandeswari, PHI
2. Programming the World Wide Web, Robert W Sebesta, 7ed, Pearson.

CO – PO

PO → **CO ↓**	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	2	1	1	1	1	1	1	1	1	1	2
CO2	3	3	3	2	2	1	1	1	2	1	1	2
CO3	3	3	2	2	1	1	1	1	3	2	1	2
CO4	3	3	2	2	2	1	1	1	2	1	1	2
CO5	3	3	3	2	3	1	1	1	2	1	2	3

1 = Low, **2 = Moderate**, **3 = High** contribution.

OPEN ELECTIVE

Course Name: Research and Publication Ethics	Credit Scheme				Evaluation Components			
Course Code: ASC322	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2							
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Understand the fundamental concepts of philosophy and ethics and their relevance to research and professional conduct.
2. Explain ethical principles governing scientific research and the importance of honesty, integrity, and accountability.
3. Familiarize with publication ethics, authorship norms, and procedures for addressing misconduct.
4. Introduce open access publishing models, tools, and resources for identifying credible journals and publishers.
5. Develop competence in using plagiarism detection tools, understanding research metrics, and evaluating publication quality.

Course Outcomes:
On completion of this course, students will be able to:

CO1: Explain the concepts of philosophy, ethics, and moral reasoning in relation to research and professional life.
CO2: Identify types of scientific misconduct and apply principles of research integrity and intellectual honesty.
CO3: Evaluate ethical practices and publication standards to ensure compliance with COPE, WAME, and related guidelines.
CO4: Utilize open access resources, identify predatory journals, and select credible publishing platforms.
CO5: Apply plagiarism detection tools and interpret research metrics and databases for assessing research quality.

Detailed Syllabus:

Unit I 8 Hours
Philosophy and Ethics: Introduction to philosophy: definition, nature and scope, concept, branches - Ethics: definition, moral philosophy, nature of moral judgements and reactions; Scientific Conduct:

Ethics with respect to science and research - Intellectual honesty and research integrity - Scientific misconducts: Falsification, Fabrication and Plagiarism (FFP) - Redundant Publications: duplicate and overlapping publications, salami slicing - Selective reporting and misrepresentation of data.

Unit II 7 Hours

Publication ethics: definition, introduction and importance - Best practices / standards setting initiatives and guidelines: COPE, WAME, etc. - Conflicts of interest - Publication misconduct: definition, concept, problems that lead to unethical behaviour and vice versa, types - Violation of publication ethics, authorship and contributor ship - Identification of publication misconduct, complaints and appeals - Predatory publisher and journals.

Unit III 4 Hours

Open access publications and initiatives - SHERPA/RoMEO online resource to check publisher copyright & self-archiving policies - Software tool to identify predatory publications developed by SPPU - Journal finger / journal suggestion tools viz. JANE, Elsevier Journal Finder, Springer, Journal Suggester, etc.

Unit IV 8 Hours

PUBLICATION MISCONDUCT: a) Subject specific ethical issues, FFP, authorship b) Conflicts of interest c) Complaints and appeals: examples and fraud from India and abroad Software tools: Use of plagiarism software like Turnitin, Urkund and other open source software tools.

Unit VI: 8 Hours

Databases: Indexing databases, Citation databases: Web of Science, Scopus, etc. Research Metrics: Impact Factor of journal as per Journal Citations Report, SNIP, SJR, IPP, Cite Score - Metrics: h-index, g index, i10 Index, altmetrics.

Suggested Books:

Text Books:

1. Nicholas H. Steneck. Introduction to the Responsible Conduct of Research. Office of Research Integrity. 2007. Available at: https://ori.hhs.gov/sites/default/files/rcrintro.pdf
2. The Student's Guide to Research Ethics By Paul Oliver Open University Press, 2003
3. Responsible Conduct of Research By Adil E. Shamoo; David B. Resnik Oxford University Press, 2003

Reference Books:

1. Ethics in Science Education, Research and Governance Edited by Kambadur Muralidhar, Amit Ghosh Ashok Kumar Singhvi. Indian National Science Academy, 2019. ISBN : 978-81-939482-1-7. 2. Anderson B.H., Dursaton, and Poole M.: Thesis and assignment writing, Wiley Eastern 1997.
3. Bijorn Gustavii: How to write and illustrate scientific papers? Cambridge University Press.
4. Bordens K.S. and Abbott, B.b.: Research Design and Methods, Mc Graw Hill, 2008.
5. Graziano, A., M., and Raulin, M.,L.: Research Methods – A Process of Inquiry, Sixth Edition, Pearson, 2007.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	2	3	2	–	2	–	2
CO2	3	3	2	2	2	2	3	3	–	2	–	2
CO3	3	3	3	2	2	2	3	3	2	3	2	2
CO4	2	3	2	2	3	2	3	2	2	3	2	2
CO5	2	3	2	3	3	2	3	2	2	3	3	3

Course Name: Employability Skills II	Credit Scheme				Evaluation Components			
Course Code: ES 002	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2							
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. To enhance self-management abilities such as goal setting, time management, and stress control.

2. To improve proficiency in Information and Communication Technology (ICT) tools for professional use.

3. To introduce entrepreneurial concepts and encourage innovative and business-oriented thinking.

4. To create awareness about sustainable practices and the importance of green skills in employment.

5. To develop effective communication skills required for workplace interactions.

Course Outcomes:

On completion of this course, students will be able to:

CO1. Communicate confidently and effectively using verbal, non-verbal, and written communication skills.

CO2. Use ICT tools such as word processors, spreadsheets, presentations, and the internet for workplace tasks.

CO3. Identify entrepreneurial opportunities and demonstrate basic skills required to plan and manage a small venture.

CO4. Adopt environmentally responsible behavior and apply green skills in personal and professional life.

CO5. Apply self-management strategies to improve personal efficiency, confidence.

Syllabus

Unit I Communication Skills – II

Verbal & Non-verbal Communication , Effective Listening , Barriers to Communication, Writing Skills (Emails, Notices, Reports) , Workplace Communication Etiquette

Unit II Self-Management Skills – II

Self-Awareness (Strengths & Weaknesses) , Self-Confidence & Motivation Goal Setting (SMART Goals), Time Management, Stress Management Techniques

Unit III Information & Communication Technology (ICT) Skills – II

Advanced Word Processing, Spreadsheet Applications, Presentation Tools. Internet Applications & Online Safety, Digital Payment Systems

Unit IV Entrepreneurial Skills – II

Meaning & Types of Entrepreneurship, Characteristics of an Entrepreneur, Entrepreneurial Opportunities, Business Planning, Risk & Resource Management

Unit V Green Skills – II

Sustainable Development , Environmental Protection, Waste Management , Renewable & Non-renewable Resources, Green Jobs & Careers

Suggested Books:

Text Books:

1. Employability Skills (Common for All Trades) – I & II Year – Neelkanth Publishers *(English)*
2. NSQF Employability Skills for All Trades Theory (I & II Year) – Arihant Publications (English)

Reference Books:

1. Skills for Success: Personal Development and Employability – Stella Cottrell *(Bloomsbury Study Skills series)*
2. Employability Skills Vocational Subject (Level II) – Dr. Yasir Rashid & Mrs. Najmu Nissa

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	0	1	0	0	1	2	0	2	2	3	0	2
CO2	0	2	0	0	0	2	1	3	2	2	1	3
CO3	1	2	2	1	3	0	0	3	0	2	1	2
CO4	1	3	3	0	2	2	1	2	2	2	3	2
CO5	0	1	0	0	0	3	3	2	2	1	0	

Course Name: Big Data Analysis	Credit Scheme				Evaluation Components			
Course Code: CSE 408	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2							
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Make students comfortable with tools and techniques required in handling large amounts of datasets
2. Uncover various terminologies and techniques used in Big Data.
3. Use several tools publicly available to illustrate the application of these techniques.
4. Know about the research that requires the integration of large amounts of data.
5. This course will introduce students to this rapidly growing field and equip them with some of its basic principles and tools as well as its general mindset

COURSE OUTCOMES: On completion of this course, the students will be able to:

CO1: Identify and distinguish big data analytics applications

CO2: Design efficient algorithms for mining the data from large volumes.

CO3: Analyze the HADOOP and Map Reduce technologies associated with big data analytics.

CO4: Understand the fundamentals of various big data analytics techniques.

CO5: Present cases involving big data analytics in solving practical problems

Syllabus:

Unit I

Introduction to big data: Introduction to Big Data Platform – Challenges of Conventional Systems - Intelligent data analysis – Nature of Data - Analytic Processes and Tools - Analysis vs Reporting

Unit II

Mining data streams: Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams –Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness ina Window – Decaying Window - Real time Analytics Platform(RTAP) Applications – Case Studies - Real Time Sentiment Analysis- Stock Market Predictions

Unit III

Hadoop: History of Hadoop- the Hadoop Distributed File System – Components of Hadoop Analyzing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS Basics- Developing a

Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job Run-Failures-Job Scheduling-Shuffle and Sort – Task execution - Map Reduce Types and Formats- Map Reduce Features-Hadoop environment.

Unit IV

Frameworks: Applications on Big Data Using Pig and Hive – Data processing operatorsin Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and Zookeeper - IBM Infosphere Big Insights and Streams.

Unit V

Predictive Analytics- Simple linear regression- Multiple linear regression- Interpretation of regression coefficients. Visualizations - Visual data analysis techniques- interaction techniques - Systems and applications.

Suggested Books:

Text Books:

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
2. Tom White "Hadoop: The Definitive Guide" Third Edition, O'reilly Media, 2012.
3. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGrawHill Publishing, 2012

Reference Books:

1. Michael Minelli, Michele Chambers, and Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley,2013
2. Frank J. Ohlhorst, Big Data Analytics: Turning Big Data into Big Money, Wiley, 2012

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	3	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	3	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3

Course Name: Information Security Fundamentals	Credit Scheme				Evaluation Components			
Course Code: CSE C216	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	4	3	0	2				150
Prerequisite Course and code (if any):								

Course Objectives:

The course aims to:

1. Understand fundamental concepts of information and cyber security
2. Learn common threats, vulnerabilities, and attacks
3. Gain knowledge of security mechanisms, policies, and standards
4. Understand cryptography basics and system security
5. Introduce ethical, legal, and professional issues in information security

COURSE OUTCOMES: On completion of this course, the students will be able to:
CO1: Explain security principles and threat models
CO2: Identify and analyze security risks and attacks
CO3: Apply basic cryptographic techniques
CO4: Understand network, system, and application security
CO5: Recognize legal, ethical, and compliance requirements

Syllabus

Unit I

Information Security: Definition, Need, and Objectives, CIA Triad: Confidentiality, Integrity, Availability, Information Security vs Cyber Security, Threats, Vulnerabilities, Attacks, and Risks
Security Models and Principles, Overview of Security Architecture

Unit II

Passive and Active Attacks, Insider vs Outsider Attacks, Malware:(Viruses, Worms, Trojans, Ransomware, Spyware), Social Engineering Attacks, Phishing, Spoofing, and DoS/DDoS Attacks

Unit III

Introduction to Cryptography, Encryption and Decryption, Symmetric Key Cryptography (DES, AES – overview), Asymmetric Key Cryptography (RSA – overview), Hash Functions (MD5, SHA – basics), Digital Signatures, Key Management Concepts

Unit IV

Network Security Concepts, Firewalls: Types and Functions, Intrusion Detection and Prevention Systems (IDS/IPS), Secure Network Protocols (HTTPS, SSL/TLS – overview), Operating System Security, Authentication and Authorization, Access Control Models (DAC, MAC, RBAC)

Unit V

Web Application Vulnerabilities, SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Secure Software Development Lifecycle (SSDLC), Security Testing Basics

suggested books

Textbooks

1. William Stallings, *Cryptography and Network Security*
2. Michael E. Whitman & Herbert J. Mattord, *Principles of Information Security*

Reference Books

1. Charles P. Pfleeger, *Security in Computing*
2. Nina Godbole, *Information Systems Security*

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2

CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3

List of Practicals

1. Understand CIA Triad (Confidentiality, Integrity, Availability)
2. Password hashing using SHA algorithms
3. Implement encryption and decryption using: AES or DES (basic level)
4. Implement public-key cryptography using RSA
5. Implement hashing using:MD5 / SHA-256
6. Create and verify digital signatures
7. Capture and analyze network packets using Wireshark
8. Configure basic firewall rules (Linux / Windows)
9. Demonstration of Snort or similar IDS tools
10. Implement secure communication using SSL/TLS concepts

Course Name: Security and Privacy incloud	Credit Scheme				Evaluation Components			
Course Code: CSE C305	**Total Credits**	**L**	**T**	**P**	**I**	**P**	**E**	**Total**
	2							
Prerequisite Course and code (if any):								

COURSE OBJECTIVES:

The objectives of the course are to

1. Understand the fundamentals of cloud security and cryptographic techniques.
2. Learn the architectural and design principles for securing cloud environments.
3. Analyze access control mechanisms and identity management solutions in cloud platforms.
4. Apply cloud security design patterns and best practices.
5. Develop strategies for monitoring, auditing, and managing security in cloud environments

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

CO1: Explain core security concepts such as confidentiality, integrity, authentication, and cryptography in the cloud**.**
CO2: Design secure cloud architectures addressing common threats and applying best practices in isolation and data protection
CO3: Implement identity and access control mechanisms including RBAC, MFA, and identity federation
CO4: Apply security design patterns to real-world cloud scenarios for robust security architecture
CO5: Perform proactive monitoring and auditing for incident response and compliance in cloud systems

Syllabus

Unit I

Overview of cloud security- Security Services - Confidentiality, Integrity, Authentication, Nonrepudiation, Access Control - Basic of cryptography - Conventional and public-key cryptography, hash functions, authentication, and digital signatures

Unit II

Security design principles for Cloud Computing - Comprehensive data protection - End-to-end access control - Common attack vectors and threats - Network and Storage - Secure Isolation Strategies - Virtualization strategies - Inter-tenant network segmentation strategies - Data Protection strategies: Data

retention, deletion and archiving procedures for tenant data, Encryption, Data Redaction, Tokenization, Obfuscation, PKI and Key.

Unit III

Access control requirements for Cloud infrastructure - User Identification - Authentication and Authorization - Roles-based Access Control - Multi- factor authentication - Single Sign-on, Identity Federation - Identity providers and service consumers - Storage and network access control options - OS Hardening and minimization - Verified and measured boot - Intruder Detection and prevention.

Unit IV

Introduction to Design Patterns, Cloud bursting, Geo- tagging, Secure Cloud Interfaces, Cloud Resource Access Control, Secure On-Premise Internet Access, Secure External Cloud.

Unit V

Proactive activity monitoring - Incident Response, Monitoring for unauthorized access, malicious traffic, abuse of system privileges - Events and alerts - Auditing – Record generation, Reporting and Management, Tamper-proofing audit logs, Quality of Services, Secure Management, User management, Identity management, Security Information and Event Management

Suggested Book

Textbooks:

1. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance,: Tim Mather, Subra Kumaraswamy, and Shahed Latif, Publisher: O'Reilly Media, ISBN: 978-0596802769

Reference Book

1. Cloud Computing Security: Foundations and Challenges, Author: John R. Vacca, Publisher: CRC Press, ISBN: 978-1498782733

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	–	–	–	–	–	2	–	2
CO2	2	3	2	2	–	–	–	–	–	2	–	2
CO3	3	2	2	2	2	–	–	–	–	2	–	2
CO4	3	3	3	3	2	–	–	–	–	2	–	2
CO5	2	2	2	2	–	–	3	3	2	3	2	3

List of Practicals

1. Create and explore a basic cloud account (AWS / Azure / GCP – free tier)
2. Compare security and privacy issues in each model
3. Upload data to cloud storage (S3 / Blob / Cloud Storage)
4. Configure security groups / network rules
5. Configure virtual networks (VPC / VNet)
6. Encrypt and decrypt data using KMS
7. Enable cloud logging and monitoring services
8. Enable HTTPS using SSL/TLS certificates
9. Perform basic security assessment of cloud resources
10. Identify security and privacy failures