

**SCHEME AND SYLLABUS**  
***B.tech Computer Science and***  
***Engineering***  
***(Artificial Intelligence and Machine***  
***Learning)***

*National Higher Education Qualification Framework (NHEQF)*  
*Level= 7*



**Department of Computer Science and**  
**Engineering University Institute of Engineering**  
**Technology**  
**Sant Baba Bhag Singh University**  
**Batch 2026**

Sr.No	Subject Code	Subject Name	Semester	Page No
<b>About the Department, Vision, Mission, Eligibility Criteria, Career Pathway, CBCS model, PO's, PEO's, PSO</b>			ALL	i-xii
1.		Course Scheme and Summary	ALL	1-9
2.	CSE111	Introduction to programming in C	1	10
3.	MAT171		1	11
4.	AEC0010	Communication Skills –I	1	12
5.	*PHY115	Engineering Physics (include semiconductor unit)	1	13-14
6.	EVS002	Environmental Sciences	1	15
7.	CSE113	Programming in C Practical	1	16
8.	ME105	Workshop /Manufacturing Practices Practical	1	17
9.	*PHY107	Engineering Physics Practical	1	18
10.	CHM105	Engineering Chemistry	2	20-21
11.	MAT172	Engineering Mathematics II	2	22
12.	ME101	Engineering Graphics and Design	2	23-24
13.	*EE102	Basic Electrical Engineering	2	25-26
14.	MDC023	INDIAN KNOWLEDGE SYSTEM	2	27
15.	CHM107	Engineering Chemistry Practical	2	28
16.	*EE104	Basic Electrical Engineering Laboratory	2	29
17.	CSE247	Digital Communication and Networks	3	31-32
18.	CSE253	Data structure and Algorithms	3	33-34
19.	CSE259	Object Oriented Programming using C++	3	35-36
20.	CSE255	Operating System	3	37-38
21.	CAI201	Artificial Intelligence and Expert System	3	39-40
22.	CSE263	Data structure and Algorithms Laboratory	3	41-42
23.	CSE267	Operating System Laboratory	3	43-44
24.	CSE265	Object Oriented Programming using C++ Laboratory	3	45-46
25.	CSE269	Digital Communication and Networks Laboratory	3	47-48
26.	CSE271	Four Week Sinstitutional/Industrial Training Evaluation (Undertaken after 2 <sup>nd</sup> Sem)	3	49
27.	MAT272	Discrete Mathematics	4	51-52
28.	CSE254	Database Management System	4	53-54
29.	CSE272	Computer Architecture	4	55
30.	MDC019	Universal Human Values: Understanding Harmony	4	56-57
31.	AEC0015	Effective Technical Communication Skills	4	58-59
32.	CSE260	Database Management System Laboratory	4	60-61
33.	CSE274	Computer Architecture Laboratory	4	62

34.	CAI202	Introduction to Data Analytics Laboratory	4	63
35.	CAI204	Artificial Intelligence Laboratory	4	64
36.	CSE351	Advanced Computer Networks	5	66-67
37.	CSE353	Design and Analysis of Algorithms	5	68-69
38.	CAI301	Fuzzy System & Evolutionary Computing	5	70
39.	CSE359	Computer Programming using Python	5	71
40.	CSE355	Computer Graphics	5	72-73
41.	CSE357	Advanced Computer Networks Laboratory	5	74
42.	CSE377	Design and Analysis of Algorithms Laboratory	5	75-76
43.	CAI305	Fuzzy and Neural Network Laboratory	5	77
44.	CSE365	Computer Programming using Python Laboratory	5	78
45.	CSE367	Four weeks industrial/institutional training evaluation (undertaken after 4th sem)	5	79
46.	CSE361	Computer Graphics Laboratory	5	80
47.	CSE369	Mobile Application Development	5	81-82
48.	CSE371	Introduction to Internet of Things	5	83
49.	CAI307	Pattern Recognition & Neural Networks	5	84
50.	CSE373	Cloud computing	5	85
51.	CAI309	Generative AI and Large Language Model	5	86
52.	CSE352	Internet web Programming	6	88-89
53.	CSE354	Software Engineering	6	90
54.	CSE356	Programming in Java	6	91
55.	MDC018	Gender, Culture & Development	6	92
56.	CAI302	Deep Learning	6	93
57.	CSE380	Software Engineering Laboratory	6	94
58.	CSE382	Internet web Programming Laboratory	6	95
59.	CSE384	Programming in Java Laboratory	6	96
60.	CSE366	Digital Image Processing	6	97
61.	CSE314	Computer Vision	6	98-99
62.	CSE362	Compiler Construction	6	100-101
63.	CAI306	MLOps (Machine Learning Operations)	6	102
64.	CSE348	Digital Marketing	6	103-104
65.	CSE378	Advanced Parallel Computing	6	105-106
66.	CAI304	Machine Learning	6	107
67.	CSE322	Distributed Systems	6	108-109
68.	CSE324	Wireless Communications	6	110-111
69.	CSE326	Block Chain	6	112
70.	CSE376	Advanced Database Management System	6	113
71.	CAI401	Data Visualization	7	115
72.	CSE403	Cyber Security	7	116
73.	CAI403	Big Data Analytics and its Applications	7	117
74.	MDC007	Managing Innovation and Entrepreneurship	7	118
75.	CSE405	Theory of Automata and Computation	7	119-120
76.	*CSE481	Major Project	7	121

77.	CSE485	Four weeks industrial training evaluation (undertaken after 6th sem)	7	122
78.	CSE451	Cryptography	7	123
79.	CAI405	Human Computer Interaction	7	124-125
80.	CAI407	AI Agents and Autonomous	7	126-127
81.	CSE455	Natural Language Processing	7	128
82.	CSE477	Data Mining in Business Intelligence	7	129
83.	CSE466	Six Months Industrial Training	8	131
<b>OPEN ELECTIVES</b>				
84.	CSE391	Basics of Artificial Intelligence		133-134
85.	CSE393	Introduction to Cloud Computing		135
86.	CSE491	Introduction to Operating System		136-137
87.	CSE493	Basics of Networking		138
88.	CSE495	Introduction to Digital Marketing		139
89.	CSE497	Basic Concepts of IOT		140
90.	CSE489	E-Commerce		141
91.	CSE499	Introduction to Cyber Security		142-143
92.	EMC	Entrepreneurship Mindset		144-153

# ProgrammeCode:UG056

## ABOUT THE DEPARTMENT

The Department of Computer Science and Engineering focuses not only on theoretical aspects but also emphasizes the overall development of students. There are Special Interest Groups among the faculty, each concentrating on specific research domains such as Data Mining and Big Data Analytics, Wireless and Mobile Computing, Security and Trust Computing, Wireless Sensor Networks and IoT, Soft Computing, Image Processing, Machine Learning and Data Analytics, Natural Language Processing, Cloud Computing and Social Networking, Network Security, Service-Oriented Architecture, and Theoretical Computer Science.

The department's many strengths include a high faculty-to-student ratio, state-of-the-art facilities, a strong focus on teaching and learning balanced with cutting-edge research, and an emphasis on leadership, service, and ethics. The efficacy of the teaching-learning process is reflected in the consistently excellent results achieved each year.

To enhance professional competence, the department encourages collaboration with external talent and regularly organizes hackathons, seminars, workshops, industrial visits, and expert lectures. These initiatives not only enrich the learning experience but also foster leadership qualities in budding engineers.

## SALIENT FEATURES OF THE DEPARTMENT

1. Provides a learning environment strongly focused on collaborative and interdisciplinary research under the guidance of experienced and qualified faculty. The majority of the faculty members are doctorates.
2. The teaching programme here is devised keeping in view the significance of Industry-Academia interaction, enabling the students to face global competitiveness with effective communication skills.
3. The CSE Department regularly organizes conferences, hackathons, seminars, student symposia, short-term training programs, and value-added courses. This provides a wide range of opportunities for faculty and students to bring out their potential and innovative skills in a variety of fields.
4. The department has well-equipped computing laboratories and a rich repository of software covering a wide spectrum of applications. The department, in collaboration with IIT, has set up a Virtual Lab for remote experiments. Besides this, the department takes in NPTEL and MOOC courses for both its students and faculty.
5. Digital Library with access to journals and video lectures of eminent professors.

## **B.Tech (Bachelor of Technology)**

Educational qualification matters a lot in gaining success. Along with academic qualifications, technical skills are also required. Job openings for Software professionals are much higher in the corporate sector than in the public sector. Professionals can join as junior programmer, database administrator, junior network manager, Data Analyst, Software Developer, Software Engineer, and Client-Server Systems Manager, etc., in the initial stage.

Students have job opportunities at organizations like IBM, Intel, HP, TCS, Infosys, Wipro, Tech Mahindra, CTS, and Dell in India and abroad.

# ProgrammeCode:UG056

## VISION

"Empower every student to be innovative, creative, and acquire skills in Computer Science & Engineering to enrich society and achieve a happy, successful, and meaningful life."

## MISSION

"Our mission is to provide high-quality undergraduate and postgraduate education in Computer Science & Engineering that promotes the all-round growth of individuals. We aim to create a futuristic environment that fosters critical thinking, dynamism, and innovation, transforming students into globally competitive professionals. Additionally, we are committed to empowering youth in rural communities with computer education."

## ELIGIBILITY CRITERIA

Passed the 10+2 examinations with Physics and Mathematics as compulsory subjects, along with one of the following: Chemistry, Computer Science, Biology, Biotechnology, or Technical Vocational subjects. Obtained at least 45% marks (40% in the case of candidates belonging to reserved categories) in the above subjects taken together.

B.Tech (Lateral Entry): A Diploma in Engineering & Technology from an AICTE-approved institution or a B.Sc (N.M.) from a UGC-approved university with at least 45% marks (40% in the case of reserved categories).

### **DURATION**

*B.Tech CSE AI & ML 4 years*

*B.Tech CSE AI & ML Leet-3 years*

### **CAREER PATHWAY**

Job openings for software professionals are much higher in the corporate sector than in the public sector. Professionals can join as junior programmers, database administrators, junior network managers, data analysts, software developers, software engineers, and client-server systems managers, etc., in the initial stage.

Students have job opportunities in organizations like IBM, Intel, HP, TCS, Infosys, Wipro, Tech Mahindra, CTS, and Dell, both in India and abroad. These are some of the big names that aspiring software engineers are aware of. On the other hand, companies like Infosys, Capgemini, Accenture, Cognizant, etc., pay anywhere between 3–3.5 LPA to freshers.

All the companies mentioned above are leading organizations that hire B.Tech CSE freshers. So, as a B.Tech CSE graduate, a candidate can earn anywhere from 2 LPA to 10 LPA, depending on their skills and experience. There's definitely a lot of potential to earn.

# ProgrammeCode:UG056

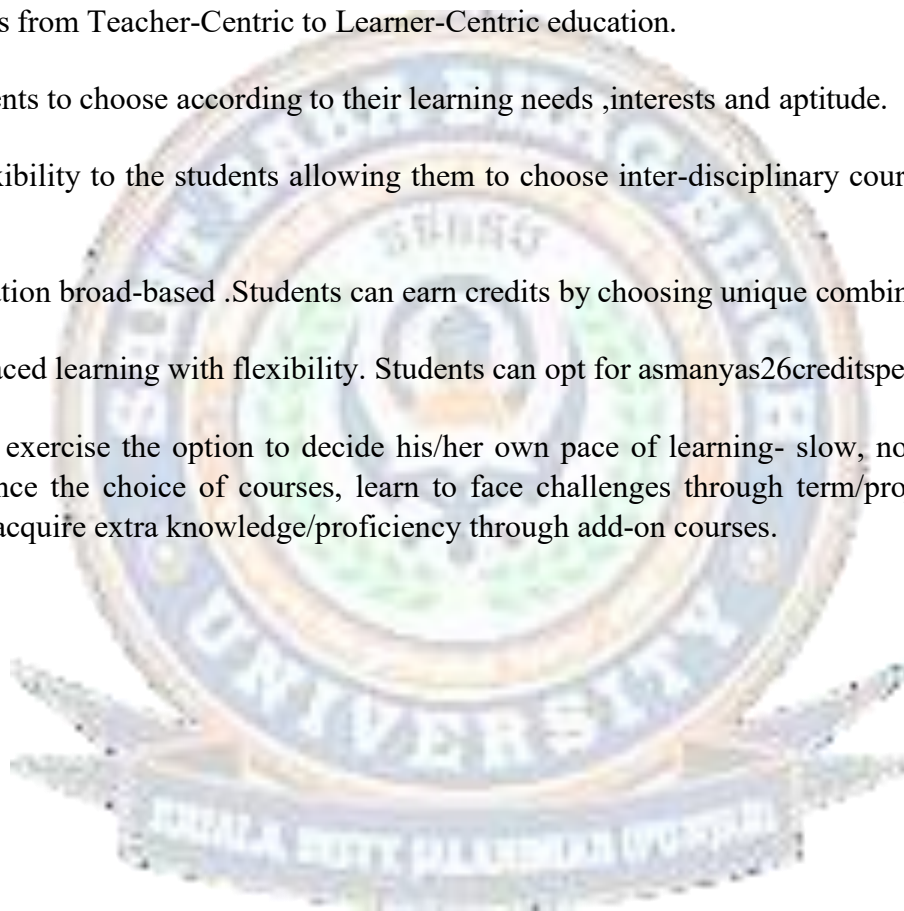
## CHOICE BASED CREDIT SYSTEM (CBCS)

### PREAMBLE:

The University Grants Commission, New Delhi, in its 12th Plan Guidelines, directed the Universities in the country to implement the Choice Based Credit System (CBCS) to set a benchmark in the University education and fulfil expectations of all the stakeholders.

### OBJECTIVES

1. Shift in focus from Teacher-Centric to Learner-Centric education.
2. Allow students to choose according to their learning needs ,interests and aptitude.
3. Provide flexibility to the students allowing them to choose inter-disciplinary courses, change majors, programs
4. Make education broad-based .Students can earn credits by choosing unique combinations.
5. Help self-paced learning with flexibility. Students can opt for asmanyas26creditspersemester.
6. Student can exercise the option to decide his/her own pace of learning- slow, normal or accelerated planned sequence the choice of courses, learn to face challenges through term/project work and may venture out to acquire extra knowledge/proficiency through add-on courses.



# ProgrammeCode:UG056

All India Council for Technical Education, New Delhi

## UPDATION/ADDENDUM

in

### Model Curriculum for Undergraduate Degree Courses in Engineering & Technology

**January 2018 (Volume-II)**

(As per Inputs of Experts)

1. The curriculum of **Humanities, Social Science including Management courses (HSMC)**
  - (i) Human Values courses is updated.
  - (i) Course Code HSMC (HU-102) may be read as (H-102) along with the following:-
    - a. The name of the course "Universal Human Values 2: Self, Society, and Nature" has been renamed as "Universal Human Values 2: Understanding Harmony".
    - b. The contents of "Universal Human Values 2: Understanding Harmony" will be included.

# ProgrammeCode:UG056

All India Council for Technical Education Model Curriculum for Undergraduate

Degree Courses in Engineering & Technology

COMPUTER SCIENCE AND ENGINEERING

Chapter-1

General, Course structure Theme&

## General, Course Structure & Theme, and Semester-wise Credit Distribution

### A. Definition of Credit:

1Hr.Lecture(L)per week	1credit
1 Hr. Tutorial(T)per week	1credit
1 Hr. Practical(P)per week	0.5credit
2 Hours Practical (Lab)/week	1credit

**B. Range of credits-**A range of credits from 150 to 160 for a student to be eligible to get Under Graduate degree in Engineering. A student will be eligible to get Under Graduate degree with Honours or additional Minor Engineering, if he/she completes an additional 20 credits. These could be acquired through MOOCs.

### C. Structure of Undergraduate Engineering program:

S. No.	Category	Credit Breakup for CSE students
1	<b>Humanities and Social Sciences, including Management courses</b>	15
2	<b>Basic Science courses</b>	23
3	<b>Engineering Science courses, including workshop, drawing, basics of electrical/mechanical/computer, etc.</b>	29
4	<b>Professional core courses</b>	49
5	<b>Professional Elective courses relevant to chosen specialization/branch</b>	18
6	<b>Open subjects – Electives from other technical and/or emerging subjects</b>	12
7	<b>Project work, seminar, and internship in industry or elsewhere</b>	15
8	Mandatory Courses [Environmental Sciences, Induction Program, Indian Constitution, Essence of Indian Knowledge Tradition]	(non-credit)
	<b>Total</b>	<b>162</b>

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

# ProgrammeCode:UG056

## D. Course code and definition:

Course code	Definitions
L	Lecture
T	Tutorial
P	Practical
BS	Basic Science Courses
ES	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management courses
PC	Professional core courses
PE	Professional Elective courses
OE	Open Elective courses
MC/ AU	Mandatory courses/ Audit Courses
EEC	Employment Enhancement Courses (Project/ Summer Internship/ Seminar)

### ➤ Course level coding scheme

Three-digit number (odd numbers are for the odd semester courses and even numbers are for even semester courses) used as suffix with the Course Code for identifying the level of the course. Digit at hundred's place signifies the year in which course is offered.

e.g. 101, 102 ... etc. for first year.

201, 202 .... Etc. for second year.

301, 302 ... for third year.

### ➤ Course Level/Duration/System:

Undergraduate / Three or Four years/6 or 8 Semesters with multiple entry and exit. The following option will be made available to the students joining Computer Science and Engineering (AI and ML) Program:

**A. One year:** Under Graduate Certificate in Computer Science and Engineering (AI and ML)

**B. Two years:** Under Graduate Diploma in Computer Science and Engineering (AI and ML)

**C. Three years:** Bachelor of Vocational in Computer Science and Engineering (AI and ML) (B.Voc.)

**D. Four years:** Bachelor of Engineering / Bachelor of Technology (B.E/B.Tech) in Computer Science and Engineering (AI and ML) Engineering

# ProgrammeCode:UG056

## Concept of Minor Degree

All branches of Engineering and Technology shall offer Elective Subjects in the Emerging/ Multidisciplinary/ Region Specific Areas as specified in the Approval Process Hand book (APH).

- a. Undergraduate Degree Subjects in Emerging / Multidisciplinary/ Region Specific Areas shall be allowed as specialization from the same department. The minimum additional Credits for such areas shall be in the range of 18-20 (including credit transferred from the SWAYAM platform) and the same shall be mentioned in the degree certificate, as specialization in that particular area. For example, doing extra credits for Cyber Security in Computer Science and Engineering shall earn B.E./B.Tech. (Hons.) Computer Science and Engineering with specialization in Cyber Security.
- b. Minor specialization may be allowed in any Undergraduate Degree Courses where a student of another Department shall take the minimum additional Credits in the range of 18-20 and get a degree with a minor from another Department.
- c. AICTE approval is not required for offering Minor Degree/Hons. in any such area, however the criteria is “Minor Degree or Hons. will cumulatively require additional 18 to 20 credits in the specified area in addition to the credits essential for obtaining the Undergraduate Degree in Major Discipline (i.e. 160 credits)”.

## Concept of Micro Credits / Micro Specialization

Micro Credits can be applied across various disciplines, including technical skills, soft skills, interdisciplinary topics, and emerging fields such as AI, data science, sustainability, and entrepreneurship.

Micro Credits are small, modular units of learning that allow students to gain specific skills or knowledge in a short time. These credits can be accumulated and used to meet the requirements for a diploma, undergraduate, or postgraduate degree.

Besides the core courses, programs normally have professional elective courses. Each HEI decides the electives it can or wishes to offer. In some areas may be desirable to organize a set of electives as micro specializations. A micro- specialization is to provide a limited specialization in some sub-area of various disciplines, by offering suitable electives. The goal of micro specialization is to provide deeper understanding and skill development in that area, and can provide multiple pathways to students, as different students can graduate with 15 Model curriculum for UG Degree in Computer Science and Engineering different specializations (or not). The areas in which micro specialization are offered should be aligned to industry careers or higher studies. A micro specialization for various disciplines may be defined as follows:

- It has a core course as the head (starting) course for the micro specialization
- It has a clearly defined goal, and learning outcomes for the goal
- It can have 2 +/- 0.5 additional courses (besides the head course) in the sub- area aligned to the goal.

These courses can be full course (4-credits) or half-course (2 credit), and can be taken as electives by students (or extra credits.)

## ProgrammeCode:UG056

Institutions can replace or add a course aligned to the micro specialization goal and also define a set of courses for a micro specialization and require that a subset be taken, with perhaps one being compulsory. It should be added that HEIs are completely free to decide whether to offer micro specializations or not, and if they decide to offer, which areas to provide the specialization in. How the micro specialization is to be reflected in a student's records/certificates is also to be decided entirely by HEIs based on their policies and practices.

Multiple pathways: For supporting multiple pathways within the academic program, we propose to provide for micro specializations through thematic course streams. These can be further enhanced by HEIs with programs like honors for advanced students with more credits or advanced learning outcomes, etc.

### **Benefits of Integrating Micro Credits:**

**Enhanced Learning Flexibility:** Students can choose from a wide array of micro-courses, allowing them to tailor their education to their career goals and interests.

**Skill Development:** Micro Credits focus on specific, practical skills that are immediately applicable in the workplace, enhancing employability.

**Lifelong Learning:** Micro Credits support continuous learning, making it easier for students and professionals to upskill or reskill in response to industry changes.

**Global Recognition:** Micro Credits can often be recognized across institutions and countries, allowing students to study globally and transfer credits easily.

# ProgrammeCode:UG056

Induction Program (Please refer Appendix-A for guidelines)

<b>Induction program(mandatory)</b>	<b>3weeksduration</b> (Please refer Appendix-A for guidelines & also details available in the curriculum of Mandatory courses)
Induction program for students to be offered right at the start of the First year.	<ul style="list-style-type: none"><li>• Physical activity</li><li>• Creative Arts</li><li>• Universal Human Values</li><li>• Literary</li><li>• Proficiency Modules</li><li>• Lectures by Eminent People</li><li>• Visits to local Areas</li><li>• Familiarization to Dept./Branch&amp;Innovations</li></ul>

## **E. Mandatory Visits/ Workshop/Expert Lectures:**

- a. It is mandatory to arrange one industrial visit every semester for the students of each branch.
- b. It is mandatory to conduct a One-week workshop during the winter break after fifth semester on professional/ industry/ entrepreneurial orientation.
- c. It is mandatory to organize at least one expert lecture per semester for each branch by inviting resource persons from domain specific industry.

# ProgrammeCode:UG056

<u>Undergraduate Programme Outcomes(PO)</u>	
<u>At the end of the Programme/Degree mentioned above, the graduates will be able to:</u>	
PO1.	<b>Engineering knowledge</b> Apply the knowledge of mathematics, science, engineering complex engineering specialization to the solution of complex engineering problems.
PO2.	<b>Problem analysis:</b> Identify, formulate, research literature, and analyze engineering problems to arrive at substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	<b>Design/development of solutions:</b> Design solutions for complex engineering problems and design system components and processes to meet the specifications with consideration for public health and safety, and the cultural, societal, and environmental considerations.
PO4	<b>Conduct investigations of complex problems:</b> Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.
PO5	<b>Modern tool usage:</b> Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities, with an understanding of the limitations.
PO6	<b>Engineer and society:</b> Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to professional engineering practice.
PO7	<b>Environment and sustainability:</b> Understand the impact of professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for, sustainable development.
PO8	<b>Ethics:</b> Apply ethical principles and commit to professional ethics, responsibilities, and norms of engineering practice.
PO9	<b>Individual and team work:</b> Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
P10	<b>Communication:</b> Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports and documentation. Make effective presentations and give and receive clear instructions.
P11	<b>Project management and finance:</b> Demonstrate knowledge and understanding of engineering and management principles, and apply these to one’s own work, as a member and leader in a team. Manage projects in multidisciplinary environments.
P12	<b>Life-long learning:</b> 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.

# ProgrammeCode:UG056

<b><u>Under Graduate Programme Specific Outcomes (PSO)</u></b>	
PS O1	Ability to acquire knowledge in Computer Science and Engineering and develop innovative solutions to complex problems.
PS O2	Design and build websites, android apps, automated projects using the knowledge of programming, testing, lifecycle models, artificial intelligence ,machine learning and CASE tools.
PSO3	Pursue life long learning in advanced technologies of Computer Science and Engineering and apply it for the benefit of the society.
<b><u>Under Graduate Programme Educational Objective (PEO)</u></b>	
<b>The Graduate/Undergraduate will be....</b>	
<b>Acquiring knowledge of Computer Science and other engineering disciplines for analyzing and developing innovative solutions to real-world problems.</b>	
<b>Developing interdisciplinary projects using the latest tools, techniques, and models for the benefit of society and the environment.</b>	
<b>Demonstrating team leadership and effective communication skills while pursuing a career in life-long learning, research and development, or generating employment through startups.</b>	
<b>Preparing for competitive examinations for higher studies abroad or for securing jobs in private, public, or multinational companies.</b>	

*Semester-wise structure of curriculum*  
[L=Lecture, T=Tutorials, P=Practicals& C=Credits]

## SEMESTER I

Scheme for B.Tech.1<sup>st</sup> Semester (common to all branches)

### I. Theory Subjects

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	ES	CSE111	Introduction to programming in C	3:0:0	3:0:0	3	3
2	BS	MAT171	Engineering Mathematics-I	4:0:0	4:0:0	4	4
3	AEC-1/ HS	AEC0010	Communication Skills-I	2:0:0	2:0:0	2	2
4	BS	*PHY115	Engineering Physics (include semiconductor unit)	4:0:0	4:0:0	4	4
5	MC	EVS002	Environmental Sciences	3:0:0	NC	3	NC

### II. Practical Subjects

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	ES	CSE113	Programming in C Practical	0:0:2	0:0:1	2	1
2	ES/SEC-I	ME105	Workshop/Manufacturing Practices Practical	0:0:6	0:0:3	6	3
3	BS	*PHY107	Engineering Physics Practical	0:0:2	0:0:1	2	1
4	EMC	EMC111	Entrepreneurship Mindset & Curriculum-I (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2
5	PT	*PT101/PT103 /PT105	Physical Training-I (Sports and Yoga/NCC/NSS)	0:0:2	NC	2	NC

Total Contact Hours=32

Total Credit Hours= 20

# SEMESTER II

Scheme for B.Tech. 2<sup>nd</sup> semester (common to all branches)

## I. Theory Subjects

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	BS	CHM105	Engineering Chemistry	3:1:0	3:1:0	4	4
2	BS	MAT172	Engineering Mathematics-II	4:0:0	4:0:0	4	4
3	ES	ME101	Engineering Graphics and Design	2:0:4	2:0:2	6	4
4	ES	EE102	Basic Electrical Engineering	3:0:0	3:0:0	3	3
5	MC	MDC023	Indian Knowledge System	3:0:0	3:0:0	3	3

## II. Practical Subjects

S. No	Type	Subject code	Subject Name	Contact Hours (L:T:P)	Credit Hours (L:T:P)	Total Contact Hours	Total Credit Hours
1	BS	CHM107	Engineering Chemistry Practical	0:0:2	0:0:1	2	1
2	ES	*EE104	Basic Electrical Engineering Laboratory	0:0:2	0:0:1	2	1
3	EMC	EMC112	Entrepreneurship Mindset & Curriculum-II (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2
4	PT	*PT102/PT104/ PT106	Physical Training-II(Sports and Yoga/NCC/NSS)	0:0:2	NC	2	NC

Note: Four weeks Institutional/ Industrial Internship in campus/ industry after 2<sup>nd</sup> semester and its evaluation in 3<sup>rd</sup> semester

Total Contact Hours= 30  
Total Credits Hours = 22

## SEMESTER III

### I. Theory subjects:

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE247	Digital Communication and Networks	4:0:0	4:0:0	4	4
2	PC	CSE253	Data structure and Algorithms	4:0:0	4:0:0	4	4
3	PC	CSE259	Object Oriented Programming using C++	4:0:0	4:0:0	4	4
4	PC	CSE255	Operating System	4:0:0	4:0:0	4	4
5	BS	CAI201	Artificial Intelligence and Expert System	3:0:0	3:0:0	4	4

### II. Practical subjects:

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE263	Data structure and Algorithms Laboratory	0:0:2	0:0:1	2	1
2	PC	CSE267	Operating System Laboratory	0:0:2	0:0:1	2	1
3	PC	CSE265	Object Oriented Programming using C++ Laboratory	0:0:2	0:0:1	2	1
4	PC	CSE269	Digital Communication and Networks Laboratory	0:0:2	0:0:1	2	1
5	SI	CSE271	FOUR WEEKS INSTITUTIONAL/ INDUSTRIAL TRAINING EVALUATION (undertaken after 2 <sup>nd</sup> sem)		0:0:3		3
6	EMC	EMC211	Entrepreneurship Mindset & Curriculum-III (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2
7	PT	PT201/PT203 /PT205	Physical Training-III(Sports and Yoga/NCC/NSS)	0:0:2	NC	2	NC

Total Contact Hours= 34

Total Credits Hours= 29

# SEMESTER IV

## I.Theory Subjects:

Sr.No	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	BS	MAT272	Discrete Mathematics	4:0:0	4:0:0	4	4
2	PC	CSE254	Database Management System	4:0:0	4:0:0	4	4
3	PC	CSE272	Computer Architecture	4:0:0	4:0:0	4	4
4	MDC	MDC019	Universal Human Values Understanding Harmony	3:0:0	3:0:0	3	3
5	AEC	AEC0015	Effective Technical Communication Skills	2:0:0	2:0:0	2	2

## Practical Subjects

Sr.No	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE260	Database Management System Laboratory	0:0:2	0:0:1	2	1
2	PC	CSE274	Computer Architecture Laboratory	0:0:2	0:0:1	2	1
3	PC	CAI202	Introduction To Data Analytics Laboratory	0:0:2	0:0:1	2	1
4	PC	CAI204	Artificial Intelligence Laboratory	0:0:2	0:0:1	2	1
5	EMC	EMC212	Entrepreneurship Mindset & Curriculum- IV (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2
6	MC	PT202/PT204/PT206	Physical Training- IV (Sports and Yoga/NCC/NSS)	0:0:2	NC	2	NC

**Note: 4 weeks industrial training after 2<sup>nd</sup> year/4<sup>th</sup> semester**

**Total Contact Hours = 31**  
**Total Credit Hours = 23**

**SEMESTER V**

**I. Theory Subjects**

S. No.	Type	Subject Code	Subject Name	Contact Hours(L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE351	Advanced Computer Networks	3:0:0	3:0:0	3	3
2	PC	CSE353	Design and Analysis of Algorithms	4:0:0	4:0:0	4	4
3	PC	CAI301	Fuzzy System and Evolutionary Computing	4:0:0	4:0:0	4	4
4	PC	CSE359	Computer Programming Using Python	4:0:0	4:0:0	4	4
5	PE		Professional Elective-I	3:0:0	3:0:0	3	3
6	PC	CSE355	Computer Graphics	4:0:0	4:0:0	4	4

**II. Practical Subjects**

S. No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE357	Advanced Computer Networks Laboratory	0:0:2	0:0:1	2	1
2	PC	CSE377	Design and Analysis of Algorithms Laboratory	0:0:2	0:0:1	2	1
3	PC	CAI305	Fuzzy and Neural Network Laboratory	0:0:2	0:0:1	2	1
4	PC	CSE365	Computer Programming Using Python Laboratory	0:0:2	0:0:1	2	1
5	EMC	EMC311	Entrepreneurship Mindset & Curriculum-V (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2
6	SEC-II	CSE367	Four weeks industrial training evaluation (undertaken after 4 <sup>th</sup> sem)	-	0:0:3	-	3
7	PC	CSE361	Computer Graphics Laboratory	0:0:2	0:0:1	2	1

**III. Professional Elective-I**

S. No.	Type	Subject Code	Subject Name	Contact Hours(L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PE	CSE369	Mobile Application Development	3:0:0	3:0:0	3	3
2	PE	CSE371	Introduction to Internet to Things	3:0:0	3:0:0	3	3
3	PE	CAI307	Pattern Recognition and Neural Networks	3:0:0	3:0:0	3	3
4	PE	CSE373	Cloud computing	3:0:0	3:0:0	3	3
5	PE	CAI309	Generative AI and Large Language Models	3:0:0	3:0:0	3	3

Total Contact Hours= 36  
Total Credits Hours= 32

## SEMESTER VI

### I. Theory Subjects

S.No.	Type	Subject Code	Subject Name	Contact hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE352	Internet web Programming	4:0:0	4:0:0	4	4
2	PC	CSE354	Software Engineering	4:0:0	4:0:0	4	4
3	PC	CSE356	Programming in Java	4:0:0	4:0:0	4	4
4	PE		Professional Elective-II	3:0:0	3:0:0	3	3
5	PE		Professional Elective-III	3:0:0	3:0:0	3	3
6	MDC/HS	MDC018	<b>Gender, Culture &amp; Development</b>	3:0:0	3:0:0	3	3
7	PC	CAI302	Deep Learning	4:0:0	4:0:0	4	4

### II. Practical Subjects

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PC	CSE380	Software Engineering Laboratory	0:0:2	0:0:1	2	1
2	PC	CSE382	Internet web Programming Laboratory	0:0:2	0:0:1	2	1
3	PC	CSE384	Programming in Java Laboratory	0:0:2	0:0:1	2	1
4	EMC	EMC312	Entrepreneurship Mindset & Curriculum-VI (Mandatory course by Punjab Government)	0:0:4	0:0:2	4	2

### III. Professional Elective-II

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PE	CSE366	Digital Image Processing	3:0:0	3:0:0	3	3
2	PE	CSE314	Computer Vision	3:0:0	3:0:0	3	3
3	PE	CSE362	Compiler Construction	3:0:0	3:0:0	3	3
4	PE	CAI306	<b>MLOps (Machine Learning Operations)</b>	3:0:0	3:0:0	3	3
5	PE	CSE348	Digital Marketing	3:0:0	3:0:0	3	3
6	PE	CSE378	Advanced Parallel Computing	3:0:0	3:0:0	3	3

### IV. Professional Elective-III

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PE	CAI304	Machine Learning	3:0:0	3:0:0	3	3
2	PE	CSE322	Distributed Systems	3:0:0	3:0:0	3	3
3	PE	CSE324	Wireless Communications	3:0:0	3:0:0	3	3
5	PE	CSE326	Blockchain	3:0:0	3:0:0	3	3
5	PE	CSE376	Advanced Database Management System	3:0:0	3:0:0	3	3

Note: 4 weeks industrial training after 3<sup>rd</sup> year/6<sup>th</sup> sem

Total Credits Hours = 35  
Total Contact Hours=30

# SEMESTER VII

## I. Theory Subjects

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PE		Professional Elective-IV	3:0:0	3:0:0	3	3
2	PC	CAI401	Data Visualization	3:0:0	3:0:0	3	3
3	PC	CSE403	Cybersecurity	4:0:0	4:0:0	4	4
4	PC	CAI403	Big Data Analytics and its applications	4:0:0	4:0:0	4	4
5	OE		Open Elective-III	3:0:0	3:0:0	3	3
6	MDC	MDC007	Managing Innovation and Entrepreneurship	3:0:0	3:0:0	3	3
7	PC	CSE405	Theory of Automata and Computation	4:0:0	4:0:0	4	4

## II. Practical Subjects

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	SEC	*CSE481	Major Project	0:0:4	0:0:4	4	2
2	SEC-III	CSE485	Four weeks industrial training evaluation (undertaken after 6 <sup>th</sup> sem)		Four Weeks		3

## III. Professional Elective-IV

S.No.	Type	Subject Code	Subject Name	Contact Hours (L:T:P)	Credits (L:T:P)	Total Contact Hours	Total Credit Hours
1	PE	CSE451	Cryptography	3:0:0	3:0:0	3	3
2	PE	CAI405	Human-Computer Interaction	3:0:0	3:0:0	3	3
3	PE	CAI407	AI Agents & Autonomous Systems	3:0:0	3:0:0	3	3
4	PE	CSE455	Natural Language Processing	3:0:0	3:0:0	3	3
5	PE	CSE477	Data Mining in Business Intelligence	3:0:0	3:0:0	3	3

Total Contact Hours= 28

Total Credits Hours=29

## *SEMESTER VIII*

### *I. Practical Subjects*

<b>S.No.</b>	<b>Type</b>	<b>Subject Code</b>	<b>Subject Name</b>	<b>Total Credit Hours</b>
1	SEC-IV	CSE466	Six Months Industrial Training	20

Total Credits Hours= 20

**List of Open Electives offered by the department**

**Open-Elective-I**

- 1. Basics of Artificial Intelligence**
- 2. Introduction to Cloud Computing**

**Open Elective-II**

- 1. Introduction to Operating System**
- 2. Basics of Networking**

**Open Elective-III**

- 1. Introduction to Digital Marketing**
- 2. Basic Concepts of IOT**

**Open Elective-IV**

- 1. E-commerce**
- 2. Introduction to Cyber security**

<b>Course Code</b>	<b>CSE111</b>
<b>Course Title</b>	Introduction to programming in C
<b>Type of Course</b>	ES
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic Knowledge about Computers
<b>Course Objective(s)</b>	To gain experience about structured programming. To help students to understand the implementation of Programming language. To understand various features in Programming Language.
<b>Course Outcome (CO)</b>	The students will be able to: 1. Illustrate the flowchart and to develop C programs. 2. Develop conditional and iterative statements to write C programs and exercise user defined functions to solve real time problems 3. Inscribe C programs that use Pointers to access arrays, strings and functions. 4. Exercise user defined data types including structures and unions to solve problems.

## SYLLABUS

### UNIT-I

#### Program Development & Basics

Introduction to Programming, Steps in Program Development (Problem Analysis, Algorithm Design, Coding, Testing & Debugging, Documentation), Algorithms and Flowcharts, Symbol Conventions, Constructing Flowcharts for Problem Solving, Program Debugging Techniques, Common Programming Errors and Debugging Strategies.

### UNIT-II

#### Program Structure and Operators

Structure of a C Program, Input/Output Statements (printf, scanf), Variables and Constants, Data Types (int, float, char), Type Conversion (Implicit and Explicit), Operators (Arithmetic, Relational, Logical, Bitwise, Conditional), Operator Precedence and Associativity, Standard Libraries (stdio.h, stdlib.h), Basic User-defined Data Types (struct, union).

### UNIT-III

#### Control Structures, Pointers and Functions

Decision Making (if, if-else, nested if, switch), Looping (while, do-while, for), Jump Statements (break, continue, goto), Functions (Declaration, Definition, Prototypes), Introduction to Pointers, Pointer Arithmetic, Parameter Passing (Call by Value, Call by Reference), Recursion, Scope of Variables (Local and Global).

### UNIT-IV

#### Arrays and Structures

Arrays (Single and Multidimensional), Array Initialization and Manipulation, Passing Arrays to Functions, Searching and Sorting Basics, Dynamic Memory Allocation, Pointers with Arrays and Functions, Structures and Unions (Declaration, Initialization, Arrays of Structures, Pointer to Structures), Difference between Structure and Union.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Programming in C	Byron Gottfried, Jitender chhabra	Schuam out line series
2.	Let us C	Yaswant Kanetkar	BPB Publication

3.	A structured Programming approach using C	Behrouz Forouzan	Thomas learning
<b>Course Code</b>		<b>MAT171</b>	
<b>Course Title</b>		<b>Engineering Mathematics-1</b>	
<b>Type of course</b>		BS	
<b>L T P</b>		4:0:0	
<b>Credits</b>		4	
<b>Course prerequisite</b>		Students must have knowledge of Basic Mathematics	
<b>Course Objective (CO)</b>	The objective of this course is to develop a deep understanding of fundamental concepts in calculus, including single-variable and multivariable analysis, sequences and series, and vector calculus.		
<b>Course Outcomes</b>	<p>At the end of the course, the students will be able to:</p> <p>CO1: Understand the concepts of curvature, evolutes, and involutes, and apply definite and improper integrals, including Beta and Gamma functions, to compute areas and volumes of revolutions.</p> <p>CO2: Analyze and apply theorems like Rolle's and Mean Value Theorem to functions, evaluate extreme values, and resolve indeterminate forms using L'Hôpital's rule.</p> <p>CO3: Understand and apply vector calculus concepts such as scalar and vector fields, Del operator, gradient, divergence, and curl, along with their physical interpretations in mechanics.</p> <p>CO4: Evaluate limits of sequences, determine convergence of infinite series using standard tests, and construct Taylor and Maclaurin series with error estimates.</p> <p>CO5: Perform multivariable function analysis involving partial derivatives, gradients, directional derivatives, and optimization using Lagrange multipliers.</p>		

#### UNIT-I:

Basic Calculus Curvature, evolutes and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions.

#### UNIT-II:

Single-variable Calculus Rolle's Theorem, Mean value theorems and applications; Extreme values of functions; Linear approximation; Indeterminate forms and Hospital rule. Scalar and vector fields, Differentiation of vectors, Velocity and acceleration, Del, Gradient, Divergence, Curl and their physical interpretation.

#### UNIT -III:

Sequences and series Limits of sequence of numbers, Calculation of limits, Infinite series; Tests for convergence; Power series, Taylor and Maclaurin series; Taylor theorem, convergence of Taylor series, error estimates.

#### UNIT-IV:

Multivariable Calculus (Differentiation) Limit, continuity and partial derivatives, directional derivatives, gradient, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers.

#### Recommended books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9 th Edition, John Wiley & Sons, 2006.
2. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11 th Reprint, 2010.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
4. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36 th Edition, 2010.

<b>Course Code</b>	<b>AEC0010</b>
<b>Course Title</b>	<b>Communication Skills-I</b>
<b>Type of course</b>	AEC-1
<b>L T P</b>	2:0:0
<b>Credits</b>	2
<b>Course prerequisite</b>	+2 in any stream
<b>Course Objective (CO)</b>	Objectives of the course is to: 1. Equip the learner with proficiency in reading comprehension.. 2. Enable the learner with improved writing skills and command over official/ corporate communication. 3. Enhance the learners' range of vocabulary and knowledge of the essentials of grammar.
<b>Course Outcomes</b>	At the conclusion of the course the learner will be able to: 1. Have fairly good proficiency in reading comprehension. 2. Have enhanced writing skills and command in official/ corporate communication. 3. Develop confidence in making presentation: oral or documentary. 4. Develop speaking skills.

## SYLLABUS

### UNIT-I

**Basics of Communication Skills:** Communication, Process of Communication, Types of Communication- Verbal and Nonverbal communication, Channels of Communication- Upward, Downward, Horizontal, Barriers to Communication, Role of Communication in society.

### UNIT-II

**Listening Skills:** Listening Process, Hearing and Listening, Types of Listening, Effective Listening, Barriers of Effective Listening, Note Taking

**Reading Skills:** Purpose of reading, Process of reading, reading skills Models and strategies, scanning, skimming, SQ3R, Approaches of Reading, Comprehension passages for practice.

### UNIT III

**Writing Skills:** Purpose of writing, Effective writing, Types of writing, Business Correspondence, Precise writing, Memo writing, minutes of meeting.

### UNIT-IV

**Speaking Skills:** Speech process, Skills of effective speaking, Role of audience, Feedback Skill, Oral Presentation.

### Recommended Books

Sr No	Author(s)	Title	Publisher
1.	Bhupender Kour	Effectual Communication Skills	S.K. Kataria and Sons
2.	R. Datta Roy and K.K. Dheer	Communications Skills	Vishal Publishing Company
3	The Essence of Effective Communication	Ludlow and Panton	Prentice Hall of India

Course Code	PHY115
Course Title	Engineering Physics (include semiconductor unit)
Type of course	BS
LTP	4 0 0
Credits	4
Course prerequisite	10+2 with physics as core subject.
Course Objective(CO)	The aim of the subject is to enhance the knowledge of engineering students about Semiconductor Physics and apply the knowledge to engineered semi conductor materials.
Course Outcome (CO)	<b>Students will able to:</b> CO1: Gain the knowledge to explain the concept of electronics materials. CO2: Understand the physics of semiconductors and light semi conductor interaction. CO3: illustrate the measurements of carrier density, resistivity and hall mobility using different techniques. CO4: Analyze engineered semiconductor materials and its applications.

## SYLLABUS

### UNIT I

**Electronic materials:** Free electron theory, Density of states and energy band diagrams, Kronig- Penny model (to introduce origin of band gap), Energy bands in solids, E-k diagram, Direct and indirect band gaps. Types of electronic materials: metals, semiconductors, and insulators, Density of states, Occupation probability, Fermi level, Effective mass, Phonons.

### UNIT II

**Semiconductors and Light-semiconductor interaction:** Intrinsic and extrinsic semiconductors, Dependence of Fermi level on carrier-concentration and temperature (equilibrium carrier statistics), Carrier generation and recombination, Carrier transport: diffusion and drift, p-n junction, Metal- semiconductor junction (Ohmic and Schottky), Semiconductor materials of interest for opto- electronic devices.

**Optical transitions in bulk semiconductors:** absorption, spontaneous emission, and stimulated emission; Joint density of states, Density of states for photons, Transition rates (Fermi's golden rule), Optical loss and gain; Photovoltaic effect, Exciton, Drude model.

### UNIT III

**Measurements:** Four-point probe and vander Pauw measurements for carrier density, resistivity, and hall mobility; Hot-point probe measurement, capacitance-voltage measurements, parameter extraction from diode I-V characteristics, DLTS, bandgap by UV-spectroscopy, absorption/transmission.

### UNIT IV

**Engineered semi conductor materials:** Density of states in 2D, 1d and 0D (qualitatively). Practical examples of low-dimensional systems such as quantum wells, wires, and dots: design, fabrication, and characterization techniques. Hetero junctions and associated band-diagrams.

## Text and Reference Books

<b>Sr No</b>	<b>Author(s)</b>	<b>Title</b>	<b>Publisher</b>
1.	Bhupender Kour	Effectual Communication Skills	S.K. Kataria and Sons
2.	R. Datta Roy and K.K. Dheer	Communications Skills	Vishal Publishing Company
3	The Essence of Effective Communication	Ludlow and Panton	Prentice Hall of India

<b>Course Code</b>	<b>EVS002</b>
Course Title	Environmental Education
Type of course	MC
L T P	3 0 0
Credits	NC
Course prerequisite	Nil
Course Objective (CO)	To make students aware about environment and need of maintaining it with best possible knowledge.
Course Outcomes	CO1: Measure environmental variables and interpret results. CO2: Evaluate local, regional, and global environmental topics related to resource use and management. CO3: Propose solutions to environmental problems related to resource use and management.

### SYLLABUS

#### UNIT-I

Introduction to Environment and Ecosystem: Definition and scope and importance of multidisciplinary nature of environment. Need for public awareness, Concept of Ecosystem, Structure, interrelationship, producers, Consumers and decomposers, ecological pyramids-biodiversity and importance. Hot spots of biodiversity.

#### UNIT-II

Environmental Pollution & Natural Resources: Definition, Causes, effects and control measures of air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards. Solid waste Management: Causes, effects and control measure of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies. Disaster Management: Floods, earthquake, cyclone and landslides, Natural Resources and associated problems, use and over exploitation, case studies of forest resources and water resources.

#### UNIT-III

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

#### UNIT-IV

Human Population and the Environment & Field Work: Population growth, variation among nations. Population explosion –Family Welfare Programme. Environment and human health, Human Rights, Value Education, HIV/AIDS. Women and child Welfare. Role of Information Technology in Environment and human health. Case studies

Visit to a local area to document environmental assets river/forest/grassland/hill/mountain; Visit to a local polluted site-Urban/Rural/Industrial/Agricultural; Study of common plants, insects, birds; Study of simple ecosystems-pond, river, hill slopes, etc.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	A Textbook for Environmental Studies	Erach Bharucha	Orient BlackSwan
2	Environmental Biology	Agarwal, K.C. 2001	Nidi Publ. Ltd. Bikaner.
3	Environmental Science	Miller T.G. Jr.	Wadsworth

<b>Course Code</b>	CSE113
<b>Course Title</b>	Programming in C Practical
<b>Type of Course</b>	ES
<b>L T P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	NA
<b>Course Objective(s)</b>	To connect and sensitize the students towards the environment and prevailing environmental issues (natural, physical, social and cultural).
<b>Course Outcome (CO)</b>	The students will be able to: 1. Illustrate the flowchart and to develop C programs. 2. Develop conditional and iterative statements to write C programs and exercise user defined functions to solve real time problems 3. Inscribe C programs that use Pointers to access arrays, strings and functions. Exercise user defined data types including structures and unions to solve problems.

- **Input and Output:** Write a program that takes user input for their name and age, then outputs a greeting message with their age.
- **Assignment Operators:** Create a program to calculate the area of a rectangle using assignment operators (+, -, \*, /=).
- **Variable Declaration and Initialization:** Write a program to calculate the simple interest, prompting the user for principal amount, rate, and time.
- **Conditional Statements:** Implement a program that determines whether a given number is even or odd using if-else statements.
- **Looping Constructs:** Create a program to find the factorial of a number using for loop.
- **Termination and Iteration Control:** Write a program that reads integers from the user until they enter a negative number using a do-while loop.
- **Switch Statement:** Implement a program that calculates the result of a basic arithmetic operation (+, -, \*, /) based on user input using a switch statement.
- **Functions and Global/Local Variables:** Develop a program to find the sum of elements in an array using a function with global and local variables.
- **Function Prototypes and Definitions:** Create a program with function prototypes and definitions to find the maximum of two numbers.
- **Parameter Passing:** Write a program that swaps two numbers using call-by-reference parameter passing.
- **Array Manipulation:** Implement a program to sort an array of integers in ascending order using the bubble sort algorithm.
- **Pointers and Dynamic Memory Allocation:** Develop a program to dynamically allocate memory for an array of integers, input values from the user, and then display the array.
- **Structures:** Create a program to store and display information about a student using a structure.
- **Unions:** Write a program to create a union of different data types and display its values.

<b>Course Code</b>	ME105
<b>Course Title</b>	<b>Workshop/Manufacturing Practices</b>
<b>Programme</b>	ES/SEC-1
<b>L T P</b>	0:0:6
<b>Credits</b>	3
<b>Course Prerequisites</b>	+2 Physics and Mathematics
<b>Course Objectives</b>	Upon completion of this course, the students will gain knowledge of the different manufacturing processes which are commonly employed in the industry, to fabricate components using Different materials.
<b>Course Outcome(CO)</b>	CO1: Understanding different manufacturing techniques and their relative advantages/ disadvantages with respect to different applications with selection of a suitable technique for meeting a specific fabrication need. CO2: Acquire a minimum practical skill with respect to the different manufacturing methods and develop the confidence to design and fabricate small components for their project work. CO3: Introduction to different manufacturing methods in different fields of engineering. CO4: Practical exposure to different fabrication techniques and Creation of simple components using different materials.

## LECTURES

1. Manufacturing Methods-casting, forming, machining, joining, advanced manufacturing methods
2. Fitting operations and power tools
3. Electrical and Electronics
4. Carpentry
5. Metal casting
6. Welding (arc welding and gas welding), brazing

## WORKSHOP PRACTICE

1. Machine shop
2. Fitting shop
3. Carpentry
4. Electrical and Electronics
5. Welding shop
6. Casting
7. Smithy

Examinations could involve the actual fabrication of simple components, utilizing one or more of the techniques covered above

<b>Sr.no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Workshop Technology	HSBawa	McGraw-Hill Publishing Company Limited
2.	Workshop Technology I,II,III	S K Hajra, Choudhary and AKChoudhary	Media Promoters and Publishers Pvt. Ltd., Bombay
3.	Manual on Workshop Practice	K Venkata Reddy	New Delhi
4.	Basic Workshop Practice Manual	T Jeyapoovan	Vikas Publishing House (P)Ltd., New Delhi

<b>Course Code</b>	<b>PHY107</b>
<b>Course Title</b>	<b>Engineering Physics Practical</b>
<b>Type of course</b>	Practical
<b>L TP</b>	0:0:2
<b>Credits</b>	1
<b>Course prerequisite</b>	10+2 with physics as core subject.
<b>Course Objectives</b>	The aim of the subject is to enhance the Practical knowledge of students about various aspects of fundamental of physics including mechanics, optics, wave optics, quantum mechanics; solid-state physics and its applications.
<b>Course Outcome(CO)</b>	<b>Students will be able to:</b> CO1: Measure the Magnetic effects along axis of circular coil, magnetic dipole moment of a bar magnet. CO2: Infer the characteristics, wavelength & diffraction of laser beam using Michels on interferometer, grating elements. CO3: determine numerical aperture, attenuation and propagation losses in optical fiber, various crystal structures, polarizability of a dielectric substance. CO4: Determine the resistivity, band gap of semiconductor materials. .

## SYLLABUS

**\*Note:** Perform at least 12-14 experiments from list of experiment given below.

1. To study the variation of magnetic field with distance along the axis of a circular coil carrying current.
2. To determine the magnetic dipole moment of a bar magnet and horizontal intensity of earth's magnetic field using a deflection galvanometer.
3. To study B-H curve using CRO.
4. To study the laser beam characteristics like divergence using diffraction grating aperture.
5. To determine the wavelength of a laser using Michelson interferometer.
6. To study diffraction using laser beam and thus to determine the grating element.
7. To find the refractive index of a material using spectrometer.
8. To find the refractive index of a liquid using a hollow prism and spectrometer.
9. To determine numerical aperture of an optical fiber.
10. To determine attenuation and propagation losses in optical fibers.
11. To study various crystal structures.
12. To find out polarizability of a dielectric substance.
13. To set up and observe Newton's rings.
14. To Determine Energy Band Gap of Semiconductor.
15. To determine the number of lines per millimeter of the grating using the green line of the mercury spectrum.
16. To calculate the wavelength of the other prominent lines of mercury by normal incidence method.
17. To find the acceleration of the cart in the simulator(Newton 2<sup>nd</sup> law)
18. To determine the resistivity of semiconductors by four probe Method.

<b>S. No</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Advanced Practical Physics for students	B.L. Flint & H.T. Worsnop	Asia Publishing House.
2	Advanced level Physics Practical's	Michael Nelson and Jon M. Ogborn	Heinemann Educational Publishers
3	A Text Book of Practical Physics	Indu Prakash and Ramakrishna	Kitab Mahal, New Delhi

# 2<sup>nd</sup> Semester

<b>Course Code</b>	<b>CHM105</b>
<b>Course Title</b>	Engineering Chemistry
<b>Type of course</b>	BS
<b>L T P</b>	3:1:0
<b>Credits</b>	4
<b>Course prerequisite</b>	NA
<b>Course Objective (CO)</b>	The objectives of the engineering chemistry are to relate the students with basic concepts of chemistry. Some new topics have been introduced to the syllabus for the development of the right attitudes by the engineering students to cope with new technology
<b>Course Outcomes</b>	<p><b>The course will enable the student to:</b></p> <p>CO1: Analyze microscopic chemistry in terms of atomic and molecular orbital's and intermolecular forces. Rationalize bulk properties and processes using thermodynamic considerations.</p> <p>CO2: Distinguish the ranges of the electromagnetic spectrum used for exciting different molecular energy levels in various spectroscopic techniques</p> <p>CO3: Rationalize periodic properties such as ionization potential, electronegativity, oxidation states and electronegativity.</p> <p>CO4: List major chemical reactions that are used in the synthesis of molecules.</p>

### UNIT-I

Atomic and molecular structure Schrodinger equation. Particle in a box solutions and their applications for conjugated molecules and nanoparticles. Forms of the hydrogen atom wave functions and the plots of these functions to explore their spatial variations. Molecular orbitals of diatomic molecules and plots of the multi center orbitals. Equations for atomic and molecular orbitals. Energy level diagrams of diatomics. Pi-molecular orbitals of butadiene and benzene and aromaticity. Crystal field theory and the energy level diagrams for transition metal ions and their magnetic properties. Band structure of solids and the role of doping on band structures.

### UNIT-II

Spectroscopic techniques and applications, Principles of spectroscopy and selection rules. Electronic spectroscopy. Fluorescence and its applications in medicine. Vibrational and rotational spectroscopy of diatomic molecules. Applications. Nuclear magnetic resonance and magnetic resonance imaging, surface characterization techniques. Diffraction and scattering.

*Intermolecular forces and potential energy surfaces*

Ionic, dipolar and van DerWaals interactions. Equations of state of real gases and critical phenomena Potential energy surfaces of H<sub>3</sub>, H<sub>2</sub>F and HCN and trajectories on these surfaces

### UNIT-III

Use of free energy in chemical equilibria: Thermodynamic functions: energy, entropy and free energy Estimations of entropy and free energies. Free energy and emf. Cell potentials, the Nernst equation and applications. Acid base, oxidation reduction and solubility equilibria. Water chemistry. Corrosion. Use of free energy considerations in metallurgy through Ellingham diagrams.

Periodic properties: Effective nuclear charge, penetration of orbitals, variations of s, p, d and f orbital energies of atoms in the periodic table, electronic configurations, atomic and ionic sizes, ionization energies, electron affinity and electronegativity, polarizability, oxidation states, coordination numbers and geometries, hard soft acids and bases, molecular geometries

### UNIT-IV

Stereochemistry Representations of 3 dimensional structures, structural isomers and stereo isomers, configurations and symmetry and chirality, enantiomers, diastereomers, optical activity, absolute configurations and conformational analysis. Isomerism in transitional metal compounds Organic reactions and synthesis of a drug molecule: Introduction to reactions involving substitution, addition, elimination, oxidation, reduction, cyclization and ring openings. Synthesis of a commonly used drug molecule.

<b>S.No</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1.	Engineering chemistry	J.C. Curiacose and J.Raja Ram	Tata Mcgraw-Hill Co. New Delhi.
2	Inorganic Chemistry	Gary L. Miessler, Paul J. Fischer and Donald A. Tarr, (2013).	Pearson
3	Introduction to spectroscopy (2008).	Pavia, D. L., Lampman, G. M., Kriz, G. S., and Vyvyan, J. A.	Cengage Learning.
4	Principles of Organic Synthesis	Norman and Coxon	<i>CRC Press</i>
5	Inorganic Chemistry 4 <sup>th</sup> edition	D. F. Shriver and P. W. Atkins,	Oxford University, Oxford(2006)
6	Stereochemistry conformation and Mechanism	P. S. Kalsi	New Age International
7	Thermodynamics for Chemists	S. Glasstone	East West Press, New Delhi (1950).

<b>Course Code</b>	MAT172
<b>Course Title</b>	Engineering Mathematics-II
<b>Type of course</b>	BS
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course prerequisite</b>	Students must have the knowledge of Basic Mathematics
<b>Course Objective (CO)</b>	The objective of this course is to provide students with a strong foundation in linear algebra, differential equations, and complex analysis, which are essential tools in mathematical modeling of engineering systems.
<b>Course Outcomes (CO)</b>	At the end of the course, the students will be able to: CO1: Solve systems of linear equations using matrix operations, analyze linear independence, compute eigenvalues and eigenvectors, and apply concepts like orthogonal transformations and the Cayley-Hamilton theorem. CO2: Solve first-order differential equations, including exact, linear, and Bernoulli's equations, and recognize and solve equations not of the first degree using standard techniques. CO3: Solve higher-order linear differential equations with variable coefficients using Euler-Cauchy and variation of parameters methods; apply power series methods, including Frobenius method, and solve Legendre's equations. CO4: Understand the theory of complex variables, including analytic and harmonic functions, and compute contour integrals using Cauchy's theorems and residue theory.

#### **UNIT-I:**

Matrices Linear Systems of Equations; Linear Independence; Rank of a Matrix; Determinant, Inverse of a matrix, System of linear equations; Symmetric, skew-symmetric and orthogonal matrices. Determinants; Eigenvalues and eigenvectors; Cayley-Hamilton Theorem (without proof).

#### **UNIT-II:**

First order ordinary differential equations Exact, linear and Bernoulli's equations. Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.

#### **UNIT-III:**

Ordinary differential equations of higher orders Second order linear differential equations with variable coefficients: Euler-Cauchy equations, solution by variation of parameters; Power series solutions: Legendre's equations and Legendre polynomials, Frobenius method.

#### **UNIT-IV:**

Complex Variables Differentiation, Cauchy-Riemann equations, analytic functions, harmonic functions, finding harmonic conjugate; elementary analytic functions (exponential, trigonometric, logarithm) and their properties. Contour integrals, Cauchy-Goursat theorem (without proof), Cauchy Integral formula (without proof); Taylor's series, zeros of analytic functions, Laurent's series; Cauchy Residue theorem (without proof).

#### **Recommended books:**

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10 th Edition, John Wiley & Sons, 2006.
2. S. L. Ross, Differential Equations, 3 rd Edition, Wiley India, 1984. E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India, 1995

<b>Course Code</b>	<b>ME101</b>
<b>Course Title</b>	<b>Engineering Graphics and Design</b>
<b>Type Of Course</b>	ES
<b>L T P</b>	2:0:4
<b>Credits</b>	4
<b>Course Pre-requisites</b>	Basics of Electrical Engineering
<b>Course objectives</b>	Basic Mathematics
<b>Course outcome (CO)</b>	Main objective of the Engineering Drawing is to introduce the students to visual science in the form of technical graphics. General instructions related to Theory of Orthographic Projection of points, lines, planes and solids as per the BIS codes prevalent to drawing practice will be introduced initially. Section of solids, intersection and development of surfaces, isometric projection and orthographic projection of simple solids/blocks will further upgrade the basic understanding and visualization of geometrical objects and to certain extent the machine parts.

## SYLLABUS

### UNIT-I

#### **Introduction to Engineering Drawing**

Principles of Engineering Graphics and their significance, usage of Drawing instruments, lettering, Conic sections including the Rectangular Hyperbola (General method only); Cycloid, Epicycloid, Hypocycloid and Involute; Scales – Plain, Diagonal and Vernier Scales.

#### **Orthographic Projections**

Principles of Orthographic Projections-Conventions – Projections of Points and lines inclined to both planes; Projections of planes inclined Planes – Auxiliary Planes;

### UNIT-II

#### **Projections of Regular Solids**

Inclined to both the Planes- Auxiliary Views; Draw simple annotation, dimensioning and scale. Floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc.

#### **Sections and Sectional Views of Right Angular Solids Covering**

Prism, Cylinder, Pyramid, Cone – Auxiliary Views; Development of surfaces of Right Regular Solids - Prism, Pyramid, Cylinder and Cone; Draw the sectional orthographic views of geometrical solids, objects from industry and dwellings (foundation to slab only)

### UNIT-III

#### **Isometric Projections**

Principles of Isometric projection – Isometric Scale, Isometric Views, Conventions; Isometric Views of lines, Planes, Simple and compound Solids; Conversion of Isometric Views to Orthographic Views and Vice-versa, Conventions;

#### **Overview of Computer Graphics**

listing the computer technologies that impact on graphical communication, Demonstrating knowledge of the theory of CAD software [such as: The Menu System, Toolbars (Standard, Object Properties, Draw, Modify and Dimension), Drawing Area (Background, Crosshairs, Coordinate System), Dialog boxes and windows, Shortcut menus (Button Bars), The Command Line (where applicable), The StatusBar, Different methods of zoom as used in CAD, Select and erase objects.; Isometric Views of lines, Planes, Simple and compound Solids];

### UNIT-IV

### **Customization & CAD Drawing**

consisting of set up of the drawing page and the printer, including scale settings, Setting up of units and drawing limits; ISO and ANSI standards for coordinate dimensioning and tolerancing; Orthographic constraints, Snap to objects manually and automatically; Producing drawings by using various coordinate input entry methods to draw straight lines, Applying various ways of drawing circles;

### **Annotations, layering & other Functions**

applying dimensions to objects, applying annotations to drawings; Setting up and use of Layers, layers to create drawings, Create, edit and use customized layers; Changing line lengths through modifying existing lines (extend/lengthen); Printing documents to paper using the print command; orthographic projection techniques; Drawing sectional views of composite right regular geometric solids and project the true shape of the sectioned surface; Drawing annotation, Computer-aided design (CAD) software modeling of parts and assemblies. Parametric and non-parametric solid, surface, and wireframe models. Part editing and two-dimensional documentation of models. Planar projection theory, including sketching of perspective, isometric, multiview, auxiliary, and section views. Spatial visualization exercises. Dimensioning guidelines, tolerancing techniques; dimensioning and scale multi views of dwelling;

### **Demonstration of a Simple Team Design Project that Illustrates**

Geometry and topology of engineered components: creation of engineering models and their presentation in standard 2D blueprint form and as 3D wire-frame and shaded solids; meshed topologies for engineering analysis and tool-path generation for component manufacture; geometric dimensioning and tolerancing; Use of solid-modeling software for creating associative models at the component and assembly levels; floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc. Applying colour coding according to building drawing practice; Drawing sectional elevation showing foundation to ceiling; Introduction to Building Information Modelling (BIM).

<b>S. No</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1.	Engineering Drawing	Bhatt N.D., Panchal V.M. & Ingle P.R., (2014)	Charotar Publishing House
2.	Engineering Drawing and Computer Graphics	Shah, M.B. & Rana B.C. (2008)	Pearson Education
3.	Engineering Graphics	Agrawal B. & Agrawal C. M. (2012)	TMH Publication
4.	Text book on Engineering Drawing	Narayana, K.L. & P Kannaiah (2008)	Scitech Publishers

<b>Course Code</b>	<b>EE102</b>
<b>Course Title</b>	<b>Basic Electrical Engineering</b>
<b>Type Of Course</b>	ES
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Physics & Mathematics
<b>Course objectives</b>	To familiarize with AC, DC circuits & their fundamentals, Magnetic circuits & Transformer, Electrical Machines and Measuring Instruments
<b>Course Outcome (CO)</b>	<b>By the end of the course, students will be able to:</b> <ol style="list-style-type: none"> <li>1. Understand and analyze basic electric and magnetic circuits</li> <li>2. Study the working principles of electrical machines and power converters.</li> <li>3. Introduce the components of low voltage electrical installations.</li> </ol>

## SYLLABUS

### UNIT-I

#### DC Circuits

Electrical circuit elements (R, L and C), voltage and current sources, Kirchoff current and voltage laws, analysis of simple circuits with dc excitation. Superposition, Thevenin and Norton Theorems. Time- domain analysis of first-order RL and RC circuits.

### UNIT-II

#### AC Circuits

Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power, power factor. Analysis of single-phase ac circuits consisting of R, L,C, RL, RC, RLC combinations (series and parallel), resonance. Three-phase balanced circuits, voltage and current relations in star and delta connections.

### UNIT-III

#### Transformers

Magnetic materials, BH characteristics, ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Auto-transformer and three-phase transformer connections.

### UNIT-IV

#### Electrical Machines

Generation of rotating magnetic fields, Construction and working of a three-phase induction motor, Significance of torque-slip characteristic. Loss components and efficiency, starting and speed control of induction motor. Single-phase induction motor. Construction, working, torque-speed characteristic and speed control of separately excited dc motor. Construction and working of synchronous generators.

#### Power Converters

DC-DC buck and boost converters, duty ratio control. Single-phase and three-phase voltage source inverters; sinusoidal modulation.

#### Electrical Installations

Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, power factor improvement and battery backup.

<b>S. No</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1.	Basic Electrical Engineering	D.P. Kothari and I. J. Nagrath	Tata McGraw Hill, 2010
2.	Basic Electrical Engineering	D.C. Kulshreshtha	McGraw Hill, 2009
3.	Fundamentals of Electrical Engineering	L.S. Bobrow	Oxford University Press, 2011
4.	Electrical and Electronics Technology	E. Hughes	Pearson, 2010
5.	Prentice Hall India, 1989	Electrical Engineering Fundamentals	V.D. Toro

<b>Course Code</b>	<b>MDC023</b>
<b>Course Title</b>	<b>Indian Knowledge System</b>
<b>Type of Course</b>	MDC-1
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	NA
<b>Course Objective(s)</b>	<ol style="list-style-type: none"> <li>1. Comprehend the core principles of the Indian knowledge system, including health, spirituality, and cultural preservation.</li> <li>2. Explore the contributions of ancient Indian mathematicians to number systems, geometry, and astronomy.</li> </ol>
<b>Course Outcome (CO)</b>	<p>The students will be able to:</p> <p><b>CO1:</b> Understand &amp; appreciate the uniqueness of <b>Indian Knowledge System (IKS)</b> through its Vedic literature, traditional sciences, art, &amp; architecture.</p> <p><b>CO2:</b> Explain the <b>scientific and mathematical contributions</b> of ancient Indian scholars and their relevance to modern scientific thought.</p> <p><b>CO3:</b> Demonstrate an understanding of <b>simple philosophical concepts</b> such as mindfulness, ethics, and the importance of justice and truth in personal and societal development.</p> <p><b>CO4:</b> Identify &amp; describe major aspects of <b>Indian art, culture, &amp; social systems</b>, including festivals, music, dance, cuisine &amp; community life.</p>

## SYLLABUS

### UNIT 1:

Overview of Indian Knowledge System: Introduction to Indian knowledge system, its uniqueness; Vedic literature: Rigveda, Samaveda, Yajurveda, Atharvaveda; Ayurveda: holistic health principles, natural remedies; Yoga, meditation: physical, mental well-being; Art, architecture: monuments, preservation of cultural heritage.

### UNIT 2:

Scientific and Mathematical Contributions: Introduction to science and math; Ancient Indian mathematicians: Aryabhata, Brahmagupta, Bhaskara; Number systems, invention of zero, basic arithmetic; Geometry, trigonometry: shapes, angles, trigonometric ratios; Astronomy, calendar systems, ancient Indian discoveries.

### UNIT 3:

Simple Philosophical Ideas: Importance of Kindness, Power of Imagination, Respect for Nature, Golden Rule, Mindfulness and Self-awareness, Curiosity and Asking Questions, Journey of Learning, Gratitude and Appreciation, Value of Friendship, Power of Positive Thinking, Existence of Truth, Ethics and Morality, Importance of Justice, and Nature of Reality

### UNIT 4:

Art, Culture, and Society Made Simple: Traditional Indian Festivals, Indian Folk Dances, Indian Classical Music, Indian Clothing Styles, Indian Cuisine, Indian Mythology, Traditional Indian Art Forms, and Historical Monuments, Languages of India, and Indian Wildlife and Nature, Family Structure in India, Caste System, Education System, Indian Weddings, Role of Women in Indian Society, Religious Diversity, Rural vs. Urban Life, Social Issues, Traditional Occupations, and Community Celebrations.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	The Story of Numbers	David M. Burton	<b>McGraw Hill Education</b>
2.	Introduction to Indian Philosophy	Sati Chandra Chatterjee	<b>Rupa &amp; Co</b>

Course Code	<b>CHM107</b>
Course Title	<b>Engineering Chemistry Practical</b>
Type of course	BS
L T P	0 0 2
Credits	1
Course Objectives	The chemistry laboratory course will consist of experiments illustrating the principle soft chemistry relevant to the study of science and engineering.
Course Outcome(CO)	The students will learn to: 1. Estimate rate constants of reactions from concentration of reactants/products as a function of time 2. Measure molecular/system properties such as surface tension, viscosity, conductance of solutions, redox potentials, chloride content of water, etc 3. Synthesize a small drug molecule and analyze a salt sample.

## SYLLABUS

### List of Experiment (Choice of 10-12 experiments from the following)

1. Determination of surface tension and viscosity of liquids.
2. Determination of viscosity of oil by means of Red Wood Viscometer.
3. Thin layer chromatography: Determination of R<sub>f</sub> value of a mixture by TLC.
4. Separation of methyl orange and phenolphthalein from given mixture by paper chromatography
5. Ion exchange column for removal of hardness of water/Determination of total hardness of water by EDTA method.
6. Determination of Total Residual Chlorine in water sample.
7. Colligative properties using freezing point depression
8. Determination of the rate constant of a reaction
9. Determination of cell constant and conductance of solutions
10. Potentiometry-determination of redox potentials and emfs:
11. Determine the strength of a solution pH metrically.
12. Synthesis of a polymer/drug: Preparation of Urea Formaldehyde Resin;
13. Synthesis of paracetamol
14. To bring catalysed condensation or to prepare a pure sample of dibenzal propane
15. Saponification/acid value of an oil
16. Chemical analysis of a salt
17. Lattice structures and packing of spheres
18. Models of potential energy surfaces
19. Chemical oscillations-Iodine clock reaction
20. Determination of the partition coefficient to of a substance between two immiscible liquids
21. Adsorption of acetic acid by charcoal
22. Use of the capillary viscosity meters to the demonstrate of the isoelectric point as the pH of minimum viscosity for gelatin sols and/or coagulation of the white part of egg.

<b>Course Code</b>	<b>EE104</b>
<b>Course Title</b>	<b>Basic Electrical Engineering Laboratory</b>
<b>Type Of Course</b>	ES
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Pre-requisites</b>	Basics of Electrical Engineering
<b>Course objectives</b>	To familiarize with various AC, DC circuits, Transformer, Electrical Machine and Measuring Instruments
<b>Course outcome (CO)</b>	<p><b>By the end of the course, students will be able to:</b></p> <p>CO1: Get an exposure to common electrical components and their ratings.</p> <p>CO2: Make electrical connections by wires of appropriate ratings.</p> <p>CO3: Understand the usage of common electrical measuring instruments.</p> <p>CO4: Understand the basic characteristics of transformers and electrical machines.</p> <p>CO5: Get exposure to the working of power electronic converters.</p>

**List of experiments/demonstrations:**

1. Basic safety precautions. Introduction and use of measuring instruments—voltmeter, ammeter, multi-meter, oscilloscope. Real-life resistors, capacitors and inductors.
2. Measuring the steady-state and transient time-response of R-L, R-C, and R-L-C circuits to a step change in voltage (transient may be observed on a storage oscilloscope). Sinusoidal steady-state response of R-L, and R-C circuits – impedance calculation and verification. Observation of phase differences between current and voltage. Resonance in R-L-C circuits.
3. Transformers: Observation of the no-load current waveform on an oscilloscope (non- sinusoidal wave-shape due to B-H curve nonlinearity should be shown along with a discussion about harmonics). Loading of a transformer: measurement of primary and secondary voltages and currents, and power.
4. Three-phase transformers: Star and Delta connections. Voltage and Current relationships (line-line voltage, phase-to-neutral voltage, line and phase currents). Phase-shifts between the primary and secondary side. Cumulative three-phase power in balanced three-phase circuits.
5. Demonstration of cut-out sections of machines: dc machine (commutator-brush arrangement), induction machine (squirrel cage rotor), synchronous machine (field winding – slip ring arrangement) and single-phase induction machine.
6. Torque Speed Characteristic of separately excited dc motor.
7. Synchronous speed of two and four-pole, three-phase induction motors. Direction reversal by change of phase-sequence of connections. Torque-Slip Characteristic of an induction motor. Generator operation of an induction machine driven at super- synchronous speed.
8. Synchronous Machine operating as a generator: stand-alone operation with a load. Control of voltage through field excitation.
9. Demonstration of (a) dc-dc converters (b) dc-ac converters – PWM waveform (c) the use of dc-ac converter for speed control of an induction motor and (d) Components of LT switchgear.

# **3<sup>RD</sup> SEMESTER**

<b>Course Code</b>	<b>CSE247</b>
<b>Course Title</b>	<b>Digital Communication and Networks</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of Computer, Digital Circuits and Network Arrangement.
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. Develop a foundational understanding of number systems, logic gates, and flip-flops.</li> <li>2. Explore the functionalities of each layer in the OSI model</li> <li>3. Analyze the principles of data transmission, encoding, and addressing in computer networks.</li> <li>4. Understand error detection/correction mechanisms and network control protocols.</li> </ol>
<b>Course outcome</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Convert numbers across binary, octal, decimal, and hexadecimal systems and perform binary arithmetic.</li> <li>2. Apply logic gates and flip-flops to design simple digital circuits.</li> <li>3. Understand and explain data communication functions at each layer of the OSI model</li> <li>4. Evaluate various transmission modes, multiplexing techniques, and protocol performance</li> <li>5. Analyze network design issues including error handling addressing, and congestion control</li> </ol>

### ***SYLLABUS***

#### **UNIT-I Number Systems and Digital Basics**

- Number Systems: Binary, Octal, Decimal, Hexadecimal
- Number System Conversions and Applications
- Complements: 1's and 2's
- Binary Arithmetic (Addition, Subtraction, Multiplication, Division)
- Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR
- Boolean Algebra: Laws, Simplification using K-Maps

#### **UNIT-II Introduction to Data Communication**

- Data Communication: Components, Types
- Guided vs Unguided Transmission Media
- Transmission Modes: Simplex, Half Duplex, Full Duplex
- Multiplexing: Frequency Division (FDM), Time Division (TDM), Wavelength Division (WDM)

#### **UNIT-III Design Issues and Data Link Layer**

- Framing Techniques (Character Count and Bit Stuffing)
- Error Detection Methods: Parity Check, CRC, Checksum
- Error Correction: Hamming Code
- Flow Control: Stop-and-Wait, Sliding Window

#### **UNIT-IV Network Layer, Transport and Application Layer**

##### Network Layer

- Routing Concepts: Routing vs Forwarding
  - Routing Algorithms: Distance Vector, Link State
  - Switching Techniques: Circuit Switching, Packet Switching, Message Switching
- ##### Transport & Application Layers
- Transport Layer Functions: Ports, Sockets,
  - Application layer Protocols

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Data Communication and Networking (5 <sup>th</sup> Edition)	Behrouz A. Forouzan	McGraw-Hill.
2	Data and Computer Communication (10 <sup>h</sup> Edition)	William Stallings	PearsonPrentice Hall India.
3	Computer Networks (8th Edition)	Andrew S. Tanenbaum	Pearson New InternationalEdition
4	Internetworking withTCP/IP, Volume 1, 6 <sup>th</sup> Edition	DouglasComer	Prentice Hall of India
5	TCP/IP Illustrated, Volume1	W.Richard Stevens,	Addison-Wesley, United States of America.

<b>Course Code</b>	<b>CSE253</b>
<b>Course Title</b>	<b>Data Structure and Algorithms</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of C language and C++ language
<b>Course Objectives</b>	This course work provides the thorough understanding of the Linear and Non- Linear Data Structures in solving problems and to give the idea of the efficiency of various algorithms.
<b>Course Outcome (CO)</b>	The learner will be able to – 1. For a given algorithm student will be able to analyze the algorithms to determine the time and computation complexity and justify the correctness. 2. For a given Search problem (Linear Search and Binary Search) student will be able to implement it. 3. For a given problem of Stacks, Queues and linked list student will be able to implement it and analyze the same to determine the time and computation complexity. 4. Student will be able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in terms of Space and Time complexity. 5. Student will be able to implement Graph search and traversal algorithms and determine the time and computation complexity.

### *Syllabus*

#### UNIT-I

**Introduction:** Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.

#### UNIT-II

**Stacks and Queues:** ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each type of Queues: Algorithms and their analysis.

#### UNIT-III

**Linked Lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.

**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.

#### UNIT-IV

**Sorting and Hashing:** Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.

**Graph:** Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

<b>RECOMMENDED BOOKS</b>			
<b>Sr No</b>	<b>Author(s)</b>	<b>Title</b>	<b>Publisher</b>
1.	Fundamentals of Data Structures	Illustrated Edition by Ellis Horowitz, Sartaj Sahni	Computer Science Press
2.	Algorithms, Data Structures, and Problem Solving with C++	Illustrated Edition by Mark Allen Weiss	Addison-Wesley Publishing Company
3.	How to Solve it by Computer	2 <sup>nd</sup> Impression by R.G. Dromey	Pearson Education

<b>Course Code</b>	<b>CSE259</b>
<b>Course Title</b>	<b>Object Oriented Programming with C++</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4:0:0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic Knowledge about Computers
<b>Course Objective(s)</b>	To gain experience in structured programming. To help students to understand the implementation of Programming language. To understand various features in Programming Language.
<b>Course Outcome(CO)</b>	The students will be able to: <ol style="list-style-type: none"> <li>1. Understand how C++ improves C with object-oriented features.</li> <li>2. Learn how to write inline functions for efficiency and performance.</li> <li>3. Learn the syntax and semantics of the C++ programming language.</li> <li>4. Learn how to design C++ classes for code reuse.</li> </ol>

## SYLLABUS

### UNIT-I:

**Basics:** Introduction to C++, Tokens, Identifiers, data types, control statements, functions, array, structure, union, pointers.

**Classes and Objects:** Classes, Structures and Classes, Unions and Classes are Related, Friend Functions, Friend Classes, Inline Functions, Constructors and its types, Static Class Members, When Constructors and Destructors are Executed, Scope Resolution Operator, Nested Classes, Local Classes, Passing and Returning Objects, Object Assignment

Arrays, Pointers, References and Dynamic Allocation: Arrays of objects, pointers, references, dynamic allocation operators, and the placement forms of new and delete.

### UNIT-II:

**Function Overloading and Default Arguments:** Function overloading, overloading constructor functions, finding the address of an overloaded function, overload anachronism, default arguments, function overloading and ambiguity.

**Operator Overloading:** Creating member operator function, overloading using a friend function, overloading new and delete, overloading special operators & comma operator.

### UNIT-III:

**Inheritance:** Base-class access control, inheritance and protected members, inheriting multiple base classes, constructors, destructors and inheritance, granting access, virtual base classes.

**Virtual Functions & Polymorphism:** Virtual functions, the virtual attribute is inherited, virtual functions are hierarchical, pure virtual functions, using virtual functions, early vs late binding.

**Templates:** Generic functions, applying generic functions, generic classes, type name and export keywords, power of templates.

### UNIT-IV:

Exception Handling: Fundamentals, derived-class exceptions, options, terminate() and unexpected(), uncaught exception(), exception and bad exception classes, applying exception handling.

The C++ I/O System Basics: Old vs modern C++ I/O, streams, stream classes, formatted I/O, overloading << and >>, creating manipulators.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Object Oriented Programming with C++	E.Balaguruswamy	Tata Mc.Graw Hill
2.	Object Oriented Programming using C++	R.Lafore	Galgotia Publications
3.	Mastering C++	A.R.Venugopal,Rajkumar, T.Ravishanker	TMH

<b>Course Code</b>	<b>CSE255</b>
<b>Course Title</b>	<b>Operating Systems</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4:0:0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Overview of Computer Architecture
<b>Course Objectives</b>	To learn the fundamentals of Operating Systems. <ol style="list-style-type: none"> <li>1. To learn the mechanisms of OS to handle processes and threads and their communication</li> <li>2. To learn the mechanisms involved in memory management in contemporary OS</li> <li>3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols</li> <li>4. To know the components and management aspects of concurrency management</li> <li>5. To learn to implement simple OS mechanisms</li> </ol>
<b>Course Outcome (CO)</b>	<b>The learner will be able to-</b> <ol style="list-style-type: none"> <li>1. Create processes and threads.</li> <li>2. Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.</li> <li>3. For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.</li> <li>4. Design and implement file management system.</li> <li>5. For a given I/O devices and OS (specify) develop the I/Omanagement functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/O controllers.</li> </ol>

## SYLLABUS

### UNIT-I

**Introduction:** Operating Systems functions, Types of operating systems, Multiprogramming systems, Batch systems, Time-sharing systems, Operating system operations, Special purpose operating systems, distributed systems, Different computing environments.

### UNIT-II

**Operating System Organization:** Processor and user modes, user operating system interface, Kernels, System calls and its types, System programs, Operating system structures, Virtual machines.

**Process Management:** Process states, Process Scheduling, Process hierarchy, Threads, Threading issues, Multi-threading models, Non-pre-emptive and pre-emptive scheduling algorithms, Concurrent processes, Critical section, Semaphores, methods for inter-process communication, Deadlocks.

### UNIT-III

**Memory Management:** Physical and virtual address space, Memory allocation strategies, Paging, Segmentation, Virtual memory and Demand paging, Page replacement algorithms.

File and I/O Management: Directory structure, File operations, Files system mounting, File allocation methods, Device management, Disk scheduling algorithms.

#### UNIT-IV

OS and Security: Security breaches, types of attacks, attack prevention methods, security policy and access control, OS design considerations for security, access control lists and OS support, internet and general network security, Policy mechanism, Program, network and system threats, Authentication.

Case Study: UNIX and LINUX operating systems

<b>RECOMMENDED BOOKS</b>		
<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
Operating System Concepts Essentials	9 <sup>th</sup> Edition by Avi Silberschatz, Peter Galvin, Greg Gagne	Wiley Asia Student Edition.
Operating Systems: Internals and Design Principles	5 <sup>th</sup> Edition, William Stallings	Prentice Hall of India
Operating System: A Design-oriented Approach	1st Edition by Charles Crowley	Irwin Publishing
Operating Systems: A Modern Perspective	2 <sup>nd</sup> Edition by Gary J. Nutt	Addison-Wesley
Design of the Unix Operating Systems	8 <sup>th</sup> Edition by Maurice Bach	Prentice-Hall of India
Understanding the Linux Kernel	3rd Edition, Daniel P. Bovet, Marco Cesati	O'Reilly and Associates

<b>Course Code</b>	<b>CAI201</b>
<b>Course Title</b>	<b>Artificial Intelligence and Expert System</b>
<b>Type of Course</b>	ES
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Overview of <b>AI and ML</b>
<b>Course Objectives</b>	<ol style="list-style-type: none"> <li>1. To review and strengthen important mathematical concepts required for AI &amp; ML.</li> <li>2. Introduce the concept of learning patterns from data and develop a strong theoretical foundation for understanding state of the art Machine Learning algorithms.</li> </ol>
<b>Course Outcome (CO)</b>	<b>The learner will be able to-</b>
	<ol style="list-style-type: none"> <li>1. Understand fundamental AI concepts, logic representations, and knowledge engineering.</li> <li>2. Apply mathematical and statistical tools to machine learning models.</li> <li>3. Design and implement regression and classification algorithms for real- world datasets.</li> <li>4. Analyze and evaluate clustering techniques and tackle overfitting challenges in AI systems..</li> </ol>

## SYLLABUS

### UNIT-I

Introduction: Introduction to artificial intelligence, background and applications, turing test and rational agent approaches, introduction to intelligent agents, their structure, behaviour and environment. Problem Solving and Searching Techniques: Problem characteristics, production systems, breadth first search, depth first search, heuristics search techniques, best first search, A\*algorithm, hill climbing, AND/OR graph AO\*, constraint satisfaction problem, means-end analysis, introduction to game playing, min max and alpha beta pruning.

### UNIT-II

Knowledge Representation: introduction to first order predicate logic, well-formed formulas, quantifiers, rule-based system, resolution principle, unification, forward reasoning: conflict resolution, backward reasoning, structured knowledge representation.

AI programming language: PROLOG: Syntax, procedural and declarative meaning, PROLOG unification mechanism, converting english to PROLOG facts and rules, goals, anonymous variable, lists, use of fail, CUT, NOT

### UNIT-III

Introduction to Neural Network: Hop field network, single and multi layer networks, perceptions, back propagations learning, Boltzman machine. Introduction to genetic algorithm: The genetic algorithm, genetic operators, working of genetic algorithm, problem with genetic algorithm.

### UNIT-IV

Expert System: introduction, skills/knowledge, characteristics of expert system, knowledge engineering, inferencing, forward chaining and backward chaining expert system tools, applications

and future scope Natural language processing: Introduction, language parsing, syntactic and semantic analysis, top down and bottom-up parsing, chart parsing, knowledge representation languages, ELIZA, speech Recognition

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Artificial Intelligence: A Modern Approach	Stuart Russell, Peter Norvig	Pearson Education
2	Artificial Intelligence and Expert System	DW Patterson	Prentice Hall of India
3	Artificial Intelligence	Elaine Rich, Kevin Knight, Shivashankar B. Nair	Tata Mc Graw Hill

<b>Course Code</b>	<b>CSE 263</b>
<b>Course Title</b>	<b>Data Structure and Algorithms Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of C++ Programming Language
<b>Course Objectives</b>	Allows the students to understand the implementation of data structures.
<b>Course Outcome (CO)</b>	The learner will be able to- <ul style="list-style-type: none"> <li>1. Design and analyze the time and space efficiency of the data structure</li> <li>2. Identity the appropriate data structure for given problem</li> <li>3. Gain practical knowledge on the applications of data structures</li> </ul>

## SYLLABUS

### Laboratory Experiments-

#### 1: Sequential Arrays

- 1.1: Insert a new element at end as well as at a given position
- 1.2: Delete an element from a given whose value is given or whose position is given
- 1.3: To find the location of a given element
- 1.4: To display the elements of the linear array

#### 2: Linear Linked Lists

- 2.1: Insert a new element
- 2.2: Delete an existing element
- 2.3: Search an element
- 2.4: Display all the elements

#### 3: Stacks and Queues

- 3.1: Program to demonstrate the use of stack.
- 3.2: Program to demonstration the implementation of various operations on a linear queue represented using a linear array.
- 3.3: Program to demonstration the implementation of various operations on a circular queue represented using a linear array.
- 3.4: Program to demonstration the implementation of various operations on a queue represented using a linear linked list (linked queue).

#### 4: Sorting and Searching

- 4.1: Program to sort an array of integers in ascending order using bubble sort.
- 4.2: Program to sort an array of integers in ascending order using selection sort.
- 4.3: Program to sort an array of integers in ascending order using insertion sort.
- 4.4.: Program to demonstrate the use of linear search to search a given element in an array.
- 4.5: Program to demonstrate the use of binary search to search a given element in a sorted array in ascending order.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Practical Data Structures Using C :: Beginner's Easy	Harry H. Chaudhary	Createspace LLC USA
2	Object Oriented Programming with C++	Balaguruswamy	Tata McGraw-Hill Education
3	Data Structures through C++	Yashavant P. Kanetkar	BPB Publications

<b>CourseCode</b>	<b>CSE267</b>
<b>Course Title</b>	<b>Operating System Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of Operating System, DOS Commands
<b>Course Objectives</b>	To provide the understanding of the operating system operation and inter-process communication.
<b>Course Outcome- (CO)</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Understand and execute basic commands of shell script.</li> <li>2. Apply basic operations in shell scripts which are required for different applications.</li> <li>3. Identify and understand concept of file systems in shell script</li> </ol> <p>Apply concept of creating new process from parent process.</p>

## SYLLABUS

### LIST OF PRACTICALS

1. Simulation of the CPU scheduling algorithms:
  - a) Round Robin
  - b) SJF
  - c) FCFS
  - d) Priority
2. Simulation of continuous memory management allocation techniques:
  - a) First Fit
  - b) Best Fit
  - c) Worst Fit
3. Simulation of page Replacement Algorithms:
  - a) FIFO
  - b) LRU
  - c) OPT
4. Simulation of file allocation Strategies:
  - a) Sequential
  - b) Indexed
  - c) Linked
5. Simulation of file organization techniques:
  - a) Single Level Directory
  - b) Two Level
6. Unix Commands
7. Reading from a file, Writing into a file, File Creation

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Practical Linux Programming: Device Drivers, Embedded Systems	Ashfaq A. Khan	Firewall Media
2	A Practical Guide to Linux Commands, Editors, and Shell Programming	Mark G. Sobell	Pearson Education
3	A Practical Guide to UNIX System V Release 4	M. G. Sobell	Benjamin/Cummings Publishing Company
4	100 Shell Programs in Unix	Sarika Jain	Pinnacle Technology

<b>Course Code</b>	<b>CSE265</b>
<b>Course Title</b>	<b>Object Oriented Programming using C++ Laboratory</b>
<b>Type of Course</b>	PC
<b>LTP</b>	002
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of C++ Programming Language Concepts
<b>Course Objectives</b>	This course is to help the students to give the practical implementation of the C++ programs.
<b>Course outcome</b>	The learner will be able to: <ol style="list-style-type: none"> <li>1. Design a program using member functions in and out of the class.</li> <li>2. Write a program to demonstrate the use of Constructors and Destructors.</li> <li>3. Implement operator overloading through C++ programming.</li> <li>4. Demonstrate inheritance and polymorphism in real-world problems using C++.</li> </ol>

## **LIST OF PRACTICALS**

### **1: Classes and Objects**

- 1.1 : Write a program that uses a class where the member functions are defined inside a class.
- 1.2 : Write a program that uses a class where the member functions are defined outside a class.
- 1.3 : Write a program to demonstrate the use of static data members.
- 1.4 : Write a program to demonstrate the use of const data members.

### **2: Constructors and Destructors**

- 2.1 : Write a program to demonstrate the use of zero argument and parameterized constructors.
- 2.2 : Write a program to demonstrate the use of dynamic constructor.
- 2.3 : Write a program to demonstrate the use of explicit constructor.

### **3: Operator Overloading**

- 3.1 : Write a program to demonstrate the overloading of increment and decrement operators.
- 3.2 : Write a program to demonstrate the overloading of binary arithmetic operators.
- 3.3 : Write a program to demonstrate the overloading of memory management operators.

### **4: Typecasting**

- 4.1 : Write a program to demonstrate the typecasting of basic type to class type.
- 4.2 : Write a program to demonstrate the typecasting of class type to basic type.
- 4.3 : Write a program to demonstrate the typecasting of class type to class type.

### **5: Inheritance**

- 5.1 : Write a program to demonstrate the multilevel inheritance.
- 5.2 : Write a program to demonstrate the multiple inheritances.
- 5.3 : Write a program to demonstrate the virtual derivation of a class.

### **6: Polymorphism**

- 6.1 : Write a program to demonstrate the runtime polymorphism.

### **7: Exception Handling**

- 7.1 : Write a program to demonstrate the exception handling.

## 8: File Handling

8.1 : Write a program to demonstrate the reading and writing of mixed type of data.

8.2 : Write a program to demonstrate the reading and writing of objects.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Object Oriented Programming in C++	LaforeR.	Waite Group
2	Object Oriented Programming with C++	E.Balaguruswamy	Tata McGraw Hill
3	Mastering Object-Oriented Programming with C++	R.S.Salaria	Salaria Publishing House

<b>Course Code</b>	<b>CSE269</b>
<b>Course Title</b>	<b>Digital Communication and Networks Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Basic knowledge of Computer, Digital Circuits and Network Arrangement.
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. To develop an understanding of modern network architecture from a design and performance perspective</li> <li>2. To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).</li> <li>3. To provide an opportunity to do network programming</li> <li>4. To provide a WLAN measurement idea.</li> </ol>
<b>Course outcome</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Understand functionality of various network components.</li> <li>2. Prepare straight cable and cross cable</li> <li>3. Configure TCP/IP protocol in windows &amp; LINUX</li> <li>4. Implement file and printer sharing</li> <li>5. Design class A, B and C network</li> </ol>

## SYLLABUS

### List of Practical's: -

1: Number System Conversion

1.1 : Convert binary, octal, decimal and hexadecimal numbers

2. Logic gate simulation

2.1: Verify Truth tables of AND, OR, NOT, NAND, NOR, XOR, XNOR gates

3. Specification, Familiarization of Networking Components & devices.

3.1 : Specification of laptop & computers.

3.2 : Familiarization of Networking Components & devices: LAN adapter, Hub, Switches, Routers.

4: Familiarization with transmission media & tools, Preparing cables.

4.1 : Coaxial cable, UTP Cable, Coaxial cable, UTP Cable.

4.2: Preparing straight cable & cross cable.

5: Study of topology, Study of TCP/IP Protocol.

5.1 : Study of LAN topology & their creation using N/W devices, cables & computers.

5.2: Configuration of TCP/IP protocol in windows & LINUX.

6: Addressing, File & Printer sharing.

6.1: Implementation of file & printer sharing.

6.2: Designing & implementing class A, B, C network

7: Subnet planning, FTP Server, TCP/UDP

7.1: Subnet planning & implementation.

7.2: Installation of FTP server & client.

7.3: Study of TCP/UDP performance.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1.	A+ Guide to PC Hardware Maintenance and Repair, Volume 1	Michael W. Graves	Cengage Learning
2.	Practical TCP/IP and Ethernet Networking	Deon Reynders, Edwin Wright	Newnes
3.	Data Communication and Networking: A Practical Approach	Massoud Moussavi	Cengage Learning
4.	A Practical Guide to Advanced Networking	Jeffrey S. Beasley, Piyasat Nilkaew	Pearson

<b>Course Code</b>	<b>CSE271</b>
<b>Course Title</b>	<b>Four Weeks Industrial Training Evaluation (Undertaken after 4<sup>th</sup> semester)</b>
<b>Type of Course</b>	SEC
<b>L T P</b>	
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basics of programming and software development
<b>Course Objectives</b>	To enhance programming skills of a learner, so that the learner finds solutions to problems. He also gets industrial experience of software development
<b>Course Outcomes- (CO)</b>	The learner will be able to- 1. implement software using proper software life cycle models 2. works with latest IT tools 3. Develop team leadership

Four weeks industrial training will give exposure to the practical aspects of the discipline, in real time working scenario. In addition, the student may also work on a specified task or project which may be assigned to him/her, by the industry person. The student will maintain the daily diary which will have signature of industry expert, assigned to him/ her. This daily diary will be produced by the student during practical examinations, as and when scheduled by the institute. The department will evaluate student performance based on his/her project report, running software code, CD containing code and daily diary.

4<sup>TH</sup>  
SEMESTER

<b>Course Code</b>	<b>MAT272</b>
<b>Course Title</b>	<b>Discrete Mathematics</b>
<b>Type of Course</b>	BS
<b>L T P</b>	4:0:0
<b>Credits</b>	4
<b>Course Prerequisites</b>	+2 in any stream
<b>Course objective(s)</b>	The objective of this course is to introduce students to the foundational concepts of discrete mathematics including set theory, logic, combinatorics, graph theory, algebraic structures, and probability theory. It aims to develop students' ability to apply mathematical reasoning and proof techniques to solve problems relevant to computer science and engineering.
<b>Course Outcome (CO)</b>	At the end of the course, the students will be able to: <b>CO1:</b> Understand and apply fundamental set theory concepts, relations, and functions, and use standard proof techniques such as induction, contradiction, and contraposition. <b>CO2:</b> Solve problems in number theory and combinatorics using modular arithmetic, Euclidean algorithms, Chinese Remainder Theorem, and counting techniques such as permutations, combinations, inclusion-exclusion, and recurrence relations. <b>CO3:</b> Analyze and construct different types of graphs and trees, evaluate their properties (like connectivity, planarity, coloring), and understand logical reasoning using propositional and first-order logic. <b>CO4:</b> Understand algebraic structures such as groups, rings, and fields, especially finite fields and their applications in computer science. <b>CO5:</b> Apply basic concepts of discrete probability, compute expectations and variances, and use conditional probability and Bayes' theorem in problem-solving.

## SYLLABUS

### UNIT-I: Set, Relations, Functions

Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem.

Proof Methods and Strategies: Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency, Case analysis, Induction.

### UNIT-II: Modular Arithmetic and Combinatorics.

Extended Euclid's Greatest Common Divisor algorithm, The Fundamental Theorem of Arithmetic, Modular arithmetic, Coprimality (or Euler's totient function), Chinese Remainder Theorem. Permutation & Combination, Inclusion- Exclusion, Pigeon-hole principle, Generating functions, Recurrence.

### UNIT-III: Graphs and Logics

Connected components, Paths, Cycles, Trees, Hamiltonian/ Eulerian Walks, Coloring, Planarity, Matching. Languages of Propositional logic and First-order logic, expressing natural language sentences in languages of propositional and first-order logic, expressing natural language predicates in the language of first-order logic. Semantics of First- order logic: interpretation and its use in evaluating a formula.

#### **UNIT-IV: Algebra & Discrete Probability**

Group, Permutation Groups, Cosets, Normal Subgroups, Ring, Field, Finite fields, Fermat's little theorem. Discrete Sample Space, Probability Distribution, Random variables, Expectation, Variance, Bernoulli trials, Conditional probability & independence (Bayes' Theorem).

#### **Text Books and References:**

1. Singh, S.B., Discrete Mathematics, Khanna Book Publishing Company, New Delhi.
2. Liu, C. L., & Mohapatra, D. P. (2008). Elements of Discrete Mathematics. Tata McGraw-Hill.
3. Niven, I., Zuckerman, H. S., & Montgomery, H. L. (1991). An introduction to the theory of numbers. John Wiley & Sons.

<b>Course Code</b>	<b>CSE254</b>
<b>Course Title</b>	<b>Database Management Systems</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Elementary knowledge about computers including some experience using Windows. Basic knowledge about programming in some common programming language.
<b>Course Objectives</b>	<ul style="list-style-type: none"> <li>• To understand the different issues involved in the design and implementation of a database system.</li> <li>• To study the physical and logical database designs, database modeling, relational, hierarchical, and network models</li> <li>• To understand and use data manipulation language to query, update, and manage a database</li> <li>• To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.</li> <li>• To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.</li> </ul>
<b>Course Outcome (CO)</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. For a given query write relational algebra expressions for that query and optimize the developed expressions</li> <li>2. For a given specification of the requirement design the databases using ER method and normalization.</li> <li>3. For a given specification construct the SQL queries for Open source and Commercial DBMS -MYSQL, ORACLE, and DB2.</li> <li>4. For a given query optimize its execution using Query optimization algorithms</li> <li>5. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.</li> <li>6. Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.</li> </ol>

## Syllabus

### Unit-I

#### **Introduction**

Database Systems; Database and its purpose, Characteristics of the database approach, Advantages and disadvantages of database systems. Classification of DBMS Users; Actors on the scene, Database Administrators, Database Designers, End Users, System Analysts and Application Programmers, Workers behind the scene (DBMS system designers and implementers, tool developers, operator and maintenance personnel)

### Unit-II

#### **Database System Concepts and Architecture**

Data models, schemas, instances, data base state. DBMS Architecture; The External level, The conceptual level, The internal level, Mappings. Data Independence, and its types, DBMS Language, DBMS Interfaces, Classification of Database Management Systems

#### **Data Modeling using E.R. Model (Entity Relationship Model)**

Data Models Classification; Entities and Attributes, Entity types and Entity sets, Key attribute and domain of

attributes, Relationship among entities

### **Unit-III**

#### **Relational Algebra and Relational Model**

Introduction to Relational algebra, Selection and projection, set operations, renaming, Joins, Division  
Relational Model Concepts: Domain, Attributes, Tuples and Relations. Relational constraints and relational database schemes; Domain constraints, Key constraints and constraints on Null. Entity integrity, referential integrity and foreign key

#### **Normalization**

Non-loss decomposition and functional dependencies, First, Second and Third normal forms, Boyce/Codd normal form, denormalization

### **Unit-IV**

#### **Database Access and Security**

Database security, process controls, database protection, grant and revoke

#### **SQL (Structured Query Language)**

Creating Tables, Creating a table with data from another table, Inserting values into a table, updating columns of a Table, Deleting Rows, Dropping a Table. Database Security and Privileges, Grant and Revoke Command, Commit and Rollback, various types of select commands, various types of joins, sub query, aggregate functions.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Database System Concepts	6th Edition by Abraham Silberschatz, Henry F.Korth, S. Sudarshan,	McGraw-Hill.
2	Principles of Database and Knowledge – Base Systems	Vol 1, J. D. Ullman	Computer Science Press
3	Foundations of Databases	Reprint by Serge Abiteboul, Richard Hull, Victor Vianu	Addison-Wesley

<b>Course Code</b>	<b>CSE272</b>
<b>Course Title</b>	<b>Computer Architecture</b>
<b>Type of Course</b>	<b>PC</b>
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of computers and their components.
<b>Course Objectives</b>	<ol style="list-style-type: none"> <li>1. Understand the basic structure and functioning of digital computers.</li> <li>2. Learn instruction formats, addressing modes, and control unit operations.</li> <li>3. Explore memory hierarchy and I/O systems.</li> <li>4. Analyze performance metrics and parallel processing concepts relevant to AI/ML.</li> </ol>
<b>Course Outcome (CO)</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Design and analyze the architecture of modern computer systems.</li> <li>2. Evaluate instruction sets and control mechanisms.</li> <li>3. Apply memory and I/O concepts to optimize system performance.</li> <li>4. Relate architectural principles to AI/ML workloads and parallel computing.</li> </ol>

## SYLLABUS

### Unit I: Fundamentals of Computer Architecture

- a. Functional units of a computer,
- b. Von Neumann vs Harvard architecture,
- c. Instruction cycle and types of instructions (RISC vs CISC)
- d. Addressing modes and instruction formats

### Unit II: Arithmetic and Control Units

- a. ALU design and operations (integer and floating-point),
- b. Overflow and underflow
- c. Control unit: hardwired vs microprogrammed,
- d. Pipelining basics and hazards

### Unit III: Memory Organization

- a. Memory hierarchy: cache, RAM, ROM.
- b. Virtual memory and paging
- c. Associative and cache memory
- d. Memory management techniques

### Unit IV: I/O and Performance Analysis

- a. I/O interfaces and buses
- b. Interrupts and DMA
- c. Flynn's taxonomy and parallel architectures
- d. Performance metrics: throughput, latency
- e. Amdahl's Law

<b>S.No.</b>	<b>Author(S)</b>	<b>Author</b>	<b>Publisher</b>
1	Computer Architecture: A Quantitative Approach	John L. Hennessy & David A. Patterson	TMH
2.	Computer Organization and Design	David A. Patterson & John L. Hennessy	PHI

<b>Course Type</b>	<b>MDC</b>
Course Code	MDC019
Course Title	Universal Human Values : Understanding Harmony
Type of course	Theory
L T P	3 0 0
Credits	3
Course Objective(s)	<p><b>The main objective of this course is to:</b></p> <ol style="list-style-type: none"> <li>1. Develop a universal perspective based on self- exploration about themselves (human being), family, society and nature.</li> <li>2. Understand the harmony in human relations.</li> <li>3. Strengthen the Self-Reflection.</li> <li>4. Develop commitment and courage.</li> </ol>
Course Outcomes (CO)	<p><b>After the completion of the course, students will be able:</b></p> <p>CO1: To become more aware of themselves, and their surroundings (family, society, nature).</p> <p>CO2: To become more responsible in life and in handling problems with sustainable solutions.</p> <p>CO3: To develop a sense of commitment and courage to act.</p>

## UNIT-I

### Need, Basic Guidelines, Content and Process for Value Education

1. Concept of Value Education including its needs, basic guidelines, content & process.
2. Concept of Self-Exploration, Natural Acceptance & Experiential Validation as the process for Self-Exploration.
3. Happiness & Prosperity.

## UNIT-II

### Understanding Harmony in Human Beings

1. Understanding Human Being as a Co-existence of Self (I) & Body.
2. Understanding the needs of Self (I) & Body.
3. Understanding harmony of Self (I) with Body: Sanyam & Swasthya.

## UNIT-III

### Understanding Harmony in Human Relationships

1. Need for and Importance of Values in Human Relationships.
2. Human Values to be practiced in a family.
3. Visualizing a Universal Human Order in Society.

## UNIT-IV

### Understanding Harmony in Nature & Existence

1. Need & Importance of Harmony in Nature.
2. Understanding the Interconnectedness among Four Orders of Nature.
3. Holistic Perception of Harmony at all levels of Existence.

### Sessional work:

- Organize a group discussion regarding the ways and measures to inculcate values among individuals.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1.	Human Values and Professional Ethics	Varinder Kumar	Kalyani Publishers
2.	A Foundation Course in Value Education	R.R. Gaur & R. Sangal	Excel Books Publishers
3.	Human Values and Professional Ethics	RishabhAnand	Satya Prakashan, New Delhi

<b>Course Code</b>	<b>AEC0015</b>
<b>Course Title</b>	<b>Effective Technical Communication Skills</b>
<b>Type of Course</b>	AEC
<b>L T P</b>	2:0:0
<b>Credits</b>	2
<b>Course Prerequisites</b>	General English
<b>Course Objectives</b>	Aims to teach oral and written skills in English with illustrations and examples drawn from project reports, paper presentations and published papers in scientific journals. The grammar exercises are not taught in a rule-based manner but through observation and use in specific contexts. Newspaper and popular scientific reports are also included as course material. Presentation skills will be taught through practice sessions. During the course, all participants make presentations and sympathize with the presentations. Emphasis is placed on teaching how to present the same findings orally and in writing.
<b>Course Outcome (CO)</b>	<p><b>The learner will be able to:</b></p> <ol style="list-style-type: none"> <li>1. Understand language skills.</li> <li>2. Use their technical writing and presentation skills effectively to draft business letters, email messages, faxes, acceptance and rejection letters.</li> <li>3. Analyse the importance of LSRW (Listening, Speaking, Reading, Writing) skills in communication.</li> <li>4. Enhance self-esteem and support personality development.</li> </ol>

## **SYLLABUS**

### **UNIT-I**

**Information Design and Development** – Different kinds of technical documents, information development life cycle, organizational structures, factors affecting information and document design, strategies for organization, information design and writing for print and for online media.

### **UNIT-II**

**Technical Writing, Grammar, and Editing** – Technical writing process, forms of discourse, writing drafts and revising, collaborative writing, creating indexes, technical writing style and language. Basics of grammar, study of advanced grammar, editing strategies to achieve appropriate technical style. Introduction to advanced technical communication, usability, human factors, managing technical communication projects, time estimation, single sourcing, and localization.

### **UNIT-III**

**Self-development and assessment** – self-assessment, awareness, perception and attitudes, values and beliefs, personal goal setting, career planning, self-esteem. Managing time; personal memory, rapid reading, taking notes; complex problem solving; creativity

### **UNIT-IV**

**Communication and Technical Writing-** Public speaking, Group discussion, Oral; presentation, Interviews, Graphic presentation, Presentation aids, Personality Development. Writing reports, project proposals, brochures, newsletters, technical articles, manuals, official notes, business letters, memos, progress reports,

minutes of meetings, event report.

#### UNIT-V

**Ethics-** Business ethics, Etiquettes in social and office settings, Email etiquettes, Telephone Etiquettes, engineering ethics, Managing time, Role and responsibility of engineering, Work culturing, Personal memory, Rapidreading ,Taking notes, Complex problem solving, Creativity

<b>RECOMMENDED BOOKS</b>			
<b>Sr No</b>	<b>Author(s)</b>	<b>Title</b>	<b>Publisher</b>
1.	DavidF.BeerandDavidMcMurrey	Guide to writing as anEngineer	JohnWilley.New York
2.	DianeHacker	PocketStyleManual	BedfordPublication,NewYork
3	ShivKhera	YouCanWin	MacmillanBooks
4	RamanSharma	TechnicalCommunications	OxfordPublication,London
5	DaleJungk	AppliedWritingforTechnicians	McGrawHill,NewYork

<b>Course Code</b>	<b>CSE260</b>
<b>Course Title</b>	<b>Database Management System Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of Program Development Constructs
<b>Course Objectives</b>	This practical course work allows the students to efficiently design a working software model.
<b>Course Outcome (CO)</b>	The learner will be able to- <ol style="list-style-type: none"> <li>1. Understand, appreciate and effectively explain the underlying concepts of database technologies</li> <li>2. Design and implement a database schema for a given problem-domain</li> <li>3. Normalize a database</li> <li>4. Populate and query a database using SQL DML/DDL commands.</li> <li>5. Declare and enforce integrity constraints on a database using a state-of-the-art RDBMS</li> </ol>

## SYLLABUS

### List of Practical's

#### 1: Introduction To DBMS And Its Applications

1.1 : Introduction to DBMS and its applications.

#### 2: Study of SQL Statements

2.1 : Data types, creating tables, retrieval of rows using select statement, conditional retrieval of rows, alter and drop statements.

2.2 : working with null values, matching a pattern from a table, ordering the result of a query, aggregate functions, grouping the result of a query, update and delete statements.

#### 3: Operators

3.1 : arithmetic operators- add, subtract, multiply, divide

3.2 : rename field

3.3 : logical operations-and, or, not

#### 4: Other Operations

4.1 : aggregate function- average, minimum, maximum, sum, count, count(\*)

4.2 : numeric functions- absolute, power, sqrt, round

4.3 : string functions: lower, upper, initcap, length, ltrim, rtrim, substring, lpad, rpad

#### 5: T-SQL: Transact Structured Query Language

5.1 : Implement grants and revoke commands, commit and rollback commands.

#### 6: Joins And Views

6.1 : program to illustrate use of join.

#### 6.2 : create a view. 7: Introduction to PL/SQL

7.1 : introduction to PL/SQL, basic code structure, difference b/w SQL and PL/SQL

7.2 : study PL/SQL control structure

7.2.1 Conditional control-if and case statements

7.2.2 Iterative control-loop and exit statements

7.2.3 Sequential control-goto and null statements programs

7.3 : Program to find greatest of two numbers

7.4 Program to find greatest of three numbers

7.5 : Program to perform addition, subtraction, multiplication, division according to user's choice

7.6 : Program to print first n natural numbers.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Database System Concepts	6th Edition by Abraham Silberschatz, Henry F.Korth, S. Sudarshan,	McGraw-Hill.
2	Principles of Database and Knowledge – Base Systems	Vol 1, J. D. Ullman	Computer Science Press
3	Foundations of Databases	Reprint by Serge Abiteboul, Richard Hull, Victor Vianu	Addison-Wesley

<b>Course Code</b>	<b>CSE274</b>
<b>Course Title</b>	<b>Computer Architecture Laboratory</b>
<b>SEC</b>	PC
<b>L: T:P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Basic Knowledge of infrastructure components, design infrastructure including devices, topologies and protocols.
<b>Course Objective(s)</b>	1. Develop a working understanding of digital logic and processor design. 2. Implement basic CPU operations using simulation tools. 3. Explore memory organization and I/O mechanisms through experiments. 4. Apply architectural principles to optimize performance for AI/ML workloads.
<b>Course Outcome (CO)</b>	1. Design and simulate basic components of a computer system. 2. Write and debug low-level programs using assembly language. 3. Evaluate architectural trade-offs in processor and memory design. 4. Relate practical implementations to theoretical concepts in AI/ML systems.

### List of Experiments

- 1 Multiplication and division of 8-bit numbers
- 2 Finding largest/smallest number in an array
- 3 Sorting array in ascending/descending order
- 4 Interrupt service routine implementation
- 5 Interfacing 8251 USART and initializing serial communication
- 6 Factorial computation using loops
- 7 Counting number of 1's in a byte
- 8 Reversing an array stored in memory
- 9 Logical operations using AND, OR, XOR
- 10 16-bit operations using LXI and DAD instructions

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	A Practical Introduction to Computer Architecture	Daniel Page	Cengage Learning
2	Computer System Architecture	M. Morris Mano	Newnes

<b>Course Code</b>	<b>CAI202</b>
<b>Course Title</b>	<b>Data Analytics Laboratory</b>
<b>SEC</b>	PC
<b>L:T:P</b>	0 0 2
<b>Credits</b>	1
<b>Course Objective(s)</b>	<ul style="list-style-type: none"> <li>• To understand and set up the Python programming environment for data analysis.</li> <li>• To explore Python libraries such as NumPy, SciPy, Pandas, Matplotlib, and Scikit-Learn.</li> <li>• To perform mathematical and scientific computations, data manipulation, visualization, and basic predictive modeling.</li> <li>• To enable students to handle real-world data science problems using Python tools effectively.</li> </ul>
<b>Course Outcome(CO)</b>	<ol style="list-style-type: none"> <li>1. Install and configure the Python environment with essential libraries for data science.</li> <li>2. Perform mathematical and scientific computations using NumPy and SciPy.</li> <li>3. Manipulate and clean data using the Pandas library.</li> <li>4. Create data visualizations and perform predictions using Matplotlib and Scikit- Learn.</li> </ol>

## LIST OF PRACTICALS

### 1. Python Environment Setup and Basic Essentials

Install Anaconda, Jupyter Notebook, and write a basic Python script to demonstrate variables, data types, loops, and conditionals.

### 2. Mathematical Computation using NumPy

Perform array operations, matrix multiplication, indexing, slicing, statistical calculations (mean, median, std) using NumPy.

### 3. Scientific Computing with SciPy

Solve a system of linear equations, integrate a function, and perform interpolation using SciPy.

### 4. Data Manipulation using Pandas

Load a CSV file, perform data cleaning (handling NaNs, duplicates), filtering, grouping, and aggregation.

### 5. Descriptive Statistics and Data Summary

Generate statistical summaries (describe, value\_counts), compute correlations and analyze basic trends using Pandas.

### 6. Data Visualization using Matplotlib

Create line charts, bar charts, histograms, scatter plots, and pie charts to visualize a given dataset.

### 7. Predictive Modeling using Scikit-Learn

Implement a simple Linear Regression and Decision Tree Classifier using Scikit-Learn on sample datasets.

### 8. Mini Project: Combine Tools for Data Science Workflow

Load a dataset, perform preprocessing, exploratory data analysis (EDA), visualization, and build a predictive model.

RECOMMENDED BOOKS		
Sr. No.	Name	Author
1	Python for Data Science Handbook	JakevanderPIAS
2	Python Data Science Essentials	Alberto Boschetti, Luca Massaron

<b>Course Code</b>	<b>CAI204</b>
<b>Course Title</b>	<b>Artificial Intelligence Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of AI and ML
<b>Course Objectives</b>	<ol style="list-style-type: none"> <li>1. To understand the fundamental concepts of Artificial Intelligence and Machine Learning.</li> <li>2. To learn how to represent knowledge using logic and rules.</li> <li>3. To implement regression and classification algorithms using Python.</li> <li>4. To understand and experiment with clustering and classification tasks on real-world datasets...</li> </ol>
<b>Course Outcome-(CO)</b>	<ol style="list-style-type: none"> <li>1. Explain basic AI techniques and knowledge representation in logic.</li> <li>2. Differentiate between supervised and unsupervised learning approaches.</li> <li>3. Apply machine learning algorithms like linear regression, logistic regression, and clustering.</li> <li>4. Analyze and solve real-world problems using AI and ML models.</li> </ol>

### LIST OF PRACTICALS

1. Program to Implement Knowledge Representation using Predicate Logic
2. Program to Represent Rules and Facts using Prolog or Python (Rule-Based System)
3. Program to Implement Linear Regression for Single Variable using Python
4. Program to Calculate Cost Function and Perform Gradient Descent in Linear Regression
5. Program to Implement Logistic Regression for Binary Classification using Python
6. Program to Perform Multi-class Classification using One-vs-All Strategy
7. Program to Implement K-Means Clustering Algorithm using Python
8. Program to Demonstrate Overfitting and Techniques to Prevent It (e.g., Regularization)

# 5<sup>TH</sup> SEMESTER

<b>Course Code</b>	<b>CSE351</b>
<b>Course Title</b>	<b>Advanced Computer Networks</b>
<b>Type of Course</b>	PC
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of computer and its components
<b>Course Objectives</b>	This subject dives into next-generation networking concepts, focusing on architecture, protocols, and technologies like SDN and NFV. It builds on foundational networking knowledge and prepares students for cutting-edge developments in the field.
<b>Course Outcome (CO)</b>	The learner will be able to- <ol style="list-style-type: none"> <li>1. Understand functionality of various network components, its architecture and services.</li> <li>2. Implement Congestion and Wireless Networks</li> <li>3. Compare Software defined networks and traditional networks</li> <li>4. Discuss Network Function Virtualization, its architectures, applications and use cases</li> </ol>

## SYLLABUS

### Unit I: Network Architecture & Services

- Overview of data communication models, Internet multicasting, NAT (Network Address Translation), VPN (Virtual Private Network)
- Routing algorithms: BGP, RIP, OSPF
- Differentiated and Integrated Services, SONET, ATM, MPLS, Next-generation Internet architectures
- Green communication networks and data center networking

### Unit II: Congestion & Wireless Networks

- Network congestion mechanisms
- ARQ protocols & TCP/IP variants
- Multimedia networking
- Sliding window protocol implementation
- Cellular & ad hoc wireless networks
- Medium access schemes, routing, transport layer protocols
- Security & energy management in wireless networks

### Unit III: Software Defined Networking (SDN)

- SDN vs traditional networks
- SDN controller & switch design
- OpenFlow protocol

### Unit IV: Network Function Virtualization (NFV)

- NFV architecture & use cases
- NFV orchestration
- NFV applications in 5G network

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Communication Networks: Fundamentals and Concepts and Key Architectures	Leon Garcia and IndraWidjaja	TMH
2	Computer Networks: A System approach (6th edition)	Larry L Peterson,	Bruce S. Davie
3	Advanced Computer Networks	Sanjay Sharma	S.K. Kataria and Sons

4	Advanced Computer Networks	Rahul Sharma, Manmohan Singh Kassahun Gashu Melesse	Lambert Academic Publishers
5	Advanced Computer Networks: A Practical approach	Dayanand Ambawade, Deven Shah	Technical Publications

<b>Course Code</b>	<b>CSE 353</b>
<b>Course Title</b>	<b>Design and Analysis of Algorithms</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Data Structures, C, C++ Programming language
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. Analyze the asymptotic performance of algorithms.</li> <li>2. Write rigorous correctness proofs for algorithms.</li> <li>3. Demonstrate a familiarity with major algorithms and data structures.</li> <li>4. Apply important algorithmic design paradigms and methods of analysis.</li> <li>5. Synthesize efficient algorithms in common engineering design situations.</li> </ol>
<b>Course Outcome (CO)</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms .</li> <li>2. Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.</li> <li>3. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and- conquer algorithms. Derive and solve recurrence relation.</li> <li>4. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For a given problems of dynamic-programming and develop the dynamic programming algorithms, and analyze it to determine its computational complexity.</li> </ol>

## SYLLABUS

### UNIT-I

Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade- offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters’ theorem.

### UNIT-II

Fundamental Algorithmic Strategies: Brute-Force, Greedy, Dynamic Programming, Branch- and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics – characteristics and their application domains.

### UNIT-III

Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

### UNIT-IV

Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook’s theorem, Standard NP-complete problems and Reduction techniques. Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP– P SPACE

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(S)</b>	<b>Publisher</b>
1	Introduction to Algorithms	4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein	MIT Press/McGraw-Hill

2	Fundamentals of Algorithms	E. Horowitz et al.	Pearson Education
3	Algorithm Design, 1ST Edition	Jon Kleinberg and Éva Tardos	Pearson
4	Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition	Michael T Goodrich and Roberto Tamassia	Wiley
5	Algorithms—A Creative Approach, 3RD Edition	Udi Manber	Addison-Wesley, Reading, MA

<b>Course Code</b>	<b>CAI301</b>
<b>Course Title</b>	<b>Fuzzy System and evolutionary computing</b>
<b>Type of Course</b>	OE
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge about programming in some common programming language.
<b>Course Outcomes (CO)</b>	The learner will be able to <ol style="list-style-type: none"> <li>1. Understand concepts of Fuzzy logic and fuzzy set operations.</li> <li>2. Understand and describe operations on fuzzy relations.</li> <li>3. Explain features of the membership function</li> <li>4. Implement conversion of fuzzy to crisp using fuzzy arithmetic</li> </ol>

## SYLLABUS

### UNIT-I

*Introduction, Classical Sets and Fuzzy Sets*

Background, Uncertainty and Imprecision, Statistics and Random Processes, Uncertainty in Information, Fuzzy Sets and Membership, Chance versus Ambiguity. Classical Sets - Operations on Classical Sets, Properties of Classical (Crisp) Sets, Mapping of Classical Sets to Functions Fuzzy Sets - Fuzzy Set operations, Properties of Fuzzy Sets. Sets as Points in Hypercubes

### UNIT-II

#### **Classical Relations and Fuzzy Relations**

Cartesian Product, Crisp Relations- Cardinality of Crisp Relations, Operations on Crisp Relations, Properties of Crisp Relations, Composition. Fuzzy Relations - Cardinality of Fuzzy Relations, Operations on Fuzzy Relations, Properties of Fuzzy Relations, Fuzzy Cartesian Product and Composition.

### UNIT-III

#### **Membership Functions**

Features of the Membership Function, Standard Forms and Boundaries, Fuzzification, Membership Value Assignments – Intuition, Inference, Rank Ordering, Angular Fuzzy Sets, Neural Networks, Genetic Algorithms, Inductive Reasoning

### UNIT-IV

Fuzzy-to-Crisp Conversions, Fuzzy Arithmetic, Defuzzification Methods Extension Principle - Crisp Functions, Mapping and Relations, Functions of fuzzy Sets.

Fuzzy Rule- Based Systems, Rule-Based Systems - Canonical Rule Forms, Decomposition of Compound Rules, Likelihood and Truth Qualification, Aggregation of Fuzzy Rules, Graphical Techniques of Inference Fuzzy Classification: - Classification by Equivalence Relations - Crisp Relations, Fuzzy Relations. Cluster Analysis, Cluster Validity, c-Means Clustering - Hard c-Means (HCM), Fuzzy c-Means (FCM). Classification Metric, Hardening the Fuzzy c-Partition.

<b>RECOMMENDED BOOKS</b>			
<b>S.No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
<b>1</b>	Fuzzy Sets And Fuzzy Logic	Klir.G, Yuan B.B	Prentice Hall Of India Private Limited, 1997
<b>2</b>	Fundamentals Of Neural Networks	Laurance Fausett	Prentice Hall

Course Code	<b>CSE359</b>
Course Title	<b>Computer Programming using python</b>
Type Course	PC
L T P	3:0: 0
Credits	3
Course Pre-requisite	NA
Course Objectives (CO)	<ol style="list-style-type: none"> <li>1. To understand syntax and data types used in python.</li> <li>2. To write and perform programs using control structures in python</li> <li>3. To implement programs using functions and to handle exceptions in python.</li> <li>4. To create and use classes in python programming</li> </ol>
Course Outcomes	<p>The learner will be able to:</p> <ol style="list-style-type: none"> <li>1. Understand basic syntax and data types used in python.</li> <li>2. Write and perform programs using control structures</li> <li>3. Implement programs with functions and handle Exceptions.</li> <li>4. Create and use classes in python</li> </ol>

## ***SYLLABUS***

### **UNIT-I**

Introduction : Introduction, History, Versions, Installation, Environment Variables, Command Line Execution, IDLE, Script mode and Batch mode, Editing Files, Documentation, Help, Dynamic Types, Reserved Words, Naming Conventions, Typing, id(), typeof(), Indentation, Basic Syntax, Comments, Datatype, String Values, String Methods, String formatting Method (f string,% method, and format method), String Operators, Data typecasting, Simple Output, Simple Input, print Function.

### **UNIT-II**

Control Structures: Indenting Requirements, Code Formatting Standards, Whitespace and Code Readability, Coding Style Guidelines, Nested if Statements, Ternary Operator, Truthy and Falsy Values, Operator Precedence and Associativity, Bitwise Shift Operators, Bitwise Operations, Masking and Bitwise Flags, Infinite Loops, Loop Termination Conditions, Do-While Loop, Loop Control Statements, break and continue, Exiting Nested Loops, Use Cases, Best Practices, Iterating Over Collections, Range- Based for Loops, Loop Indexing and Iteration Patterns

### **UNIT-III**

Functions and Modules: Introduction, Defining Your Own Functions, Pass, Parameters, Function Documentation, Keyword and Optional Parameters, Passing Collections to a Function, Variable Number of Arguments, Scope, Functions - "First Class Citizens", Passing Functions to a Function, map, filter, Mapping Functions in a Dictionary, Lambda, Modules and Importing, Creating Modules, Using Standard Library Modules, Module Search Path, Importing from Packages, Namespace and Module Attributes.

Exception: Errors, Runtime Errors, The Exception Model Exception Hierarchy, Handling Multiple Exceptions, Raise, assert

### **UNIT-IV**

Classes in Python: Classes in Python, Principles of Object Orientation, Creating Classes, Constructor, Constructor Overloading, Instance Methods, Static Method, Self-Keyword, File Organization, Special Methods, Class Variables, Inheritance, Polymorphism.

<b>Course Code</b>	<b>CSE 355</b>
<b>Course Title</b>	<b>Computer Graphics</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Computer graphics (basics), linear algebra, programming
<b>Course Objectives</b>	The main objective of this course is to give the student a comprehensive understanding of computer graphics and visualization and their applications. In particular participants will have the ability to understand the process of generating virtual images from virtual scenes, typically identified as a pipeline of generate, computer and store/display.
<b>Course Outcome (CO)</b>	The learner will be able to- <ol style="list-style-type: none"> <li>1. Understand the fundamental graphical operations and the implementation on computer.</li> <li>2. Get a glimpse of recent advances in computer graphics.</li> <li>3. Describe user interface issues that make the computer easy for thenovice to use.</li> <li>4. Discuss interface issues that make the computer easy for the novice to use.</li> </ol>

## SYLLABUS

### UNIT-I

**Introduction** - History of computer graphics, applications, graphics pipeline, physical and synthetic images, synthetic camera, modeling, animation, rendering, relation to computer vision and image processing, review of basic mathematical objects (points, vectors, matrix methods)

**Introduction to OpenGL** - OpenGL architecture, primitives and attributes, simple modeling and rendering of two- and three-dimensional geometric objects, indexed and RGB color models, frame buffer, double buffering, GLUT, interaction, and callbacks, picking.

### UNIT-II

**Geometric transformations**- Homogeneous coordinates, affine transformations (translation, rotation, scaling, shear), concatenation, matrix stacks and use of model view matrix in OpenGL for these operations. **Viewing** - Classical three-dimensional viewing, computer viewing, specifying views, parallel and perspective projective transformations; Visibility- Z-buffer, BSP trees, Open-GL culling, hidden- surface algorithms.

### UNIT-III

**Shading** - Light sources, illumination model, Gouraud and Phong shading for polygons. Rasterization- Line segment and polygon clipping, 3D clipping, scan conversion, polygonal fill, Bresenham's algorithm.

### UNIT-IV

**Discrete Techniques**- Texture mapping, compositing, textures in OpenGL; Ray Tracing- Recursiveray tracer, ray-sphere intersection.

**Representation and Visualization**- Bezier curves and surfaces, B-splines, visualization, interpolation, marching squares algorithm.

<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
<b>1</b>	Interactive Computer Graphics. A Top-Down Approach Using OpenGL	Edward Angel	Pearson Education
<b>2</b>	Computer Graphics with OpenGL	Donald Hearn and Pauline Baker	Prentice Hall
<b>3</b>	Computer Graphics using OpenGL	F. S. Hill Jr. and S. M. Kelley	Prentice Hall
<b>4</b>	Computer Graphics (first edition)	Peter Shirley and Steve Marschner	A. K. Peters

<b>Course Code</b>	<b>CSE357</b>
<b>Course Title</b>	<b>Advanced Computer Networks Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Basic Knowledge about Computers
<b>Course Objective(s)</b>	<i>Understand and implement advanced networking protocols and architectures. Gain hands-on experience with routing, DNS, DHCP, and network simulation tools. Analyze network performance and troubleshoot using diagnostic tools. Explore protocols relevant to modern networks including MANET, SDN, &amp; multimedia systems.</i>
<b>Course Outcome (CO)</b>	By the end of the lab course, students will be able to: a. Configure and troubleshoot advanced network services. b. Simulate and analyze routing protocols and mobility models. c. Apply theoretical knowledge to real-world networking scenarios. d. Demonstrate proficiency in using diagnostic and simulation tools.

*List of Experiments:*

These may vary slightly by university, but commonly include:

1. Router Configuration  
Basic setup and interface commands on Cisco routers.
2. IP Addressing & DHCP  
Configure IP addressing for various topologies. Set up DHCP server with BOOTP integration.
3. Routing Protocols  
Implement and debug ARP, RARP, RIP, OSPF, BGP. Static routing using netstat.
4. DNS Configuration  
Setup caching DNS client and proxy. Analyze traffic using Wireshark.
5. FTP & TFTP Servers  
Configure and test file transfers for small and large files.
6. Mail Server Setup  
Configure IMAP/POP protocols. Write SMTP client in C/C++/Java.
7. MANET Protocols  
Implement AODV, DSDV, and DSR using simulators like NS2/NS3.
8. Performance Analysis  
Study routing protocol impact on network performance. Simulate mobility models in MANET.

<b>Course Code</b>	<b>CSE377</b>
<b>Course Title</b>	<b>Design and Analysis of Algorithms Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of C++ Programming Language Concepts
<b>Course Objectives</b>	Makes the students proficient in implementing algorithms using the Algorithm design techniques.
<b>Course outcome</b>	The learner will be able to- 1. Analyze the complexities of various problems in different domains. 2. Understand methods for analyzing the efficiency and correctness of Algorithm (such as exchange arguments, recurrence, induction, and average case analysis). 3. Compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem. 4. Develop efficient algorithms for the new problem with suitable designing techniques.

## SYLLABUS

- 1. Array:** WAP. Two code and analyze to compute greatest common divisor of two numbers.
  - 1.1 : WAP two code and analyze to find the mid element in an array.
  - 1.2 : WAP. To code to analyze to find maximum and minimum element (without MAXMIN algorithm) in array.
  - 1.3 : WAP. To code and analyze to find the largest element in an array.
  - 1.4 : WAP. To code to analyze to enter elements in an array.
- 2. Searching**
  - 2.1 : WAP. To find maximum and minimum element choosing MAXMIN algorithm
  - 2.2 :WAP to code and analyze to find an element using binary search and find its time complexity
- 3. Sorting**
  - 3.1: WAP. To code and analyze to short an array of integer using HEAP Sort.
  - 3..2: WAP. To code and analyze to short an array of integer using Merge Sort.
- 4. Pattern Matching:** WAP. To code and array analyze to find all occurrence of pattern in each string.
- 5. Shortest Path Algorithm:** WAP. To code and analyze to find minimum path using Kruskal's Algorithm.
- 6. Dynamic Programming:** WAP. To code and analyze to find the distance between two characters

strings using Dynamic programming.

7. Divide and Conquer: WAP to code and analyze to find an element using linear search by applying divide and conquer technique and find its time complexity.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	The Algorithm Design Manual	Steven S Skiena	Springer Science & Business Media
2	Object Oriented Programming with C++	Balagurusamy	Tata McGraw-Hill Education
3	Object Oriented Programming Using C++	Jaspreet Singh, Mrs. Pinki Parampreet Kaur	Technical Publications

<b>Course Code</b>	<b>CAI305</b>
<b>Course Title</b>	<b>Fuzzy and Neural Network Laboratory</b>
<b>Type of Course</b>	PC
<b>L:T:P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	NA
<b>Course Objective(s)</b>	<ol style="list-style-type: none"> <li>1. To understand and perform python installation.</li> <li>2. To create python scripts using variable, data types and operators.</li> <li>3. To write programs on string manipulation, control structures and data structures.</li> <li>4. To implement programs in python using functions, modules and object oriented programming concepts</li> <li>5. To handle programs using file and exceptions</li> </ol>
<b>Course Outcome (CO)</b>	<p>The learner will be able to:</p> <ol style="list-style-type: none"> <li>1. Understand and perform python installation.</li> <li>2. Create python scripts using variable, data types and operators.</li> <li>3. Write programs on string manipulation, control structures and data structures.</li> <li>4. Implement programs in python using functions, modules and object oriented programming concepts</li> <li>5. Handle programs using file and exceptions</li> </ol>

### **LIST OF EXPERIMENTS**

1. Create triangular and trapezoidal fuzzy sets for temperature and plot their membership functions.
2. Perform union, intersection, and complement operations on two fuzzy sets and plot the results.
3. Construct fuzzy and crisp relation matrices for similarity between fruits based on sweetness and sourness.
4. Combine two fuzzy relations using max-min composition and analyze the resulting relation.
5. Calculate similarity between two products using fuzzy similarity measures based on feature fuzzy sets.
6. Design membership functions for speed categories using expert intuition and plot them.
7. Optimize membership functions using neural networks or genetic algorithms on temperature data.
8. Build a fuzzy inference system with IF–THEN rules to control fan speed based on temperature and humidity.
9. Apply defuzzification methods like centroid to convert fuzzy output values into crisp results.

<b>Course Code</b>	<b>CSE265</b>
<b>Course Title</b>	<b>Computer Programming using python laboratory</b>
<b>Type of Course</b>	PC
<b>L:T:P</b>	0:0:2
<b>Credits</b>	1
<b>Course Prerequisites</b>	NA
<b>Course Objective(s)</b>	<ol style="list-style-type: none"> <li>1. To understand and perform python installation.</li> <li>2. To create python scripts using variable, data types and operators.</li> <li>3. To write programs on string manipulation, control structures and data structures.</li> <li>4. To implement programs in python using functions, modules and object oriented programming concepts</li> <li>5. To handle programs using file and exceptions</li> </ol>
<b>Course Outcome (CO)</b>	<p>The learner will be able to:</p> <ol style="list-style-type: none"> <li>1. Understand and perform python installation.</li> <li>2. Create python scripts using variable, data types and operators.</li> <li>3. Write programs on string manipulation, control structures and data structures. Handle programs using file and exceptions.</li> <li>4. Implement programs in python using functions, modules an object oriented programming concepts</li> </ol>

### LIST OF EXPERIMENTS

1. **Installation and Environment Setup:** Install Python on your system and set up environment variables. Write a simple Python script and execute it using command line.
2. **Basic Syntax and Datatypes:** Create Python scripts to demonstrate basic syntax, including variables, data types, and operators. Experiment with different data types such as integers, floats, strings, lists, tuples, sets, and dictionaries.
3. **String Manipulation:** Write Python programs to demonstrate various string manipulation techniques, including string methods, formatting methods, and string operators.
4. **Control Structures:** Implement Python programs to practice control structures like if-else statements, nested if statements, loops (for, while), loop control statements (break, continue), and loop termination conditions.
5. **Data Structures:** Write Python programs to understand and utilize different data structures such as lists, tuples, sets, dictionaries, and demonstrate their access methods and built-in functions.
6. **Functions and Parameters:** Create Python functions with different parameter types (positional, keyword, default, variable-length) and demonstrate their usage.
7. **Lambda Functions and Functional Programming:** Practice using lambda functions, map, and filter functions for functional programming tasks like applying transformations and filtering elements
8. **Modules and Importing:** Develop Python modules with functions and import them into other scripts. Experiment with importing standard library modules and modules from packages.
9. **Exception Handling:** Write Python programs to handle exceptions gracefully using try-except blocks, raise statements, and assert statements.
10. **Object-Oriented Programming (OOP):** Implement Python classes with attributes, methods, constructors, and demonstrate concepts like inheritance, polymorphism, and class variables.
11. **Static Methods and Class Methods:** Create Python classes with static methods and class methods and demonstrate their usage.
12. **File Handling:** Write Python programs to read from and write to files, handle file objects, and practice different file handling techniques.
13. **Exception Handling in Classes:** Develop Python classes that raise and handle exceptions internally, demonstrating error handling within class methods.

Sr. No.	Name	Author(s)	Publisher
1	<i>Python Crash Course</i>	Eric Matthes	No Starch Press
2	<i>Automate the Boring Stuff with Python</i>	Al Sweigart	No Starch Press
3	<i>Learning Python</i>	Mark Lutz	O'Reilly Media

<b>Course Code</b>	<b>CSE367</b>
<b>Course Title</b>	<b>Four Weeks Industrial Training Evaluation (Undertaken after 4<sup>th</sup> semester)</b>
<b>Type of Course</b>	SEC
<b>L T P</b>	
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basics of programming and software development
<b>Course Objectives</b>	To enhance programming skills of a learner, so that the learner finds solutions to problems. He also gets industrial experience of software development
<b>Course Outcomes- (CO)</b>	The learner will be able to- 1. implement software using proper software life cycle models 2. works with the latest IT tools 3. Develop team leadership

The four weeks industrial training will give exposure to the practical aspects of the discipline, in the real-time working scenario. In addition, the student may also work on a specified task or project which may be assigned to him/her, by the industry person. The student will maintain the daily diary which will have the signature of an industry expert, assigned to him/ her. This daily diary will be produced by the student during practical examinations, as and when scheduled by the institute. The department will evaluate student performance based on his/her project report, running software code, CD containing code and daily diary.

<b>Course Code</b>	<b>CSE361</b>
<b>Course Title</b>	<b>Computer Graphics Laboratory</b>
<b>SEC</b>	PC
<b>L:T:P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of Program Development Constructs
<b>Course Objective(s)</b>	This practical course work allows the students to efficiently design a working software model.
<b>Course Outcome (CO)</b>	The learner will be able to- <ul style="list-style-type: none"> <li>1. Implement algorithms for drawing 2D primitives</li> <li>2. Implement transformations and clippings</li> <li>3. Implement 3D projections</li> </ul>

### SYLLABUS

1. Implementation of Algorithms for drawing 2D Primitives – Line (DDA, Bresenham) – all slopes  
Circle (Midpoint)
2. 2D Geometric transformations –
  - Translation
  - Rotation
  - Scaling
  - Reflection
  - Shear
  - Window-Viewport
3. Composite 2D Transformations
4. Line Clipping
5. 3D Transformations – Translation, Rotation, Scaling.
6. 3D Projections – Parallel, Perspective.
7. Creating 3D Scenes.
8. Image Editing and Manipulation – Basic Operations on image using any image editing software, Creating gif animated images, Image optimization.
9. 2D Animation – To create Interactive animation using any authoring tool.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Interactive Computer Graphics. A Top-Down Approach Using OpenGL	Edward Angel	Pearson Education
2	Computer Graphics with OpenGL	Donald Hearn and Pauline Baker	Prentice Hall

<b>Course Code</b>	<b>CSE369</b>
<b>Course Title</b>	<b>Mobile Application Development</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Students are expected to have basic knowledge of JAVA, HTML, JavaScript and CSS
<b>Course Objectives</b>	Students will learn the basics of the programming language, designing mobile interfaces, using libraries to build applications, user input and other aspects.
<b>Course Outcome (CO)</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Define multimedia to potential clients.</li> <li>2. Identify and describe the function of the general skill sets in the multimedia industry.</li> <li>3. Identify the basic components of a multimedia project.</li> <li>4. Identify the basic hardware and software requirements for multimedia development and playback.</li> </ol>

## SYLLABUS

### UNIT-I

#### *Introduction To Mobile Devices*

Mobile devices vs. desktop devices, ARM and intel architectures, Power Management, Screen resolution, Touch interfaces, Application deployment, App Store, Google Play, Windows Store, Development environments, Xcode, Eclipse, VS2012, PhoneGap, Native vs. web applications

#### ***Mobile Applications***

Introduction to mobile computing, mobile applications, Embedded systems, Market and business drivers for mobile applications, Publishing and delivery of mobile applications, Requirements gathering and validation for mobile applications

### UNIT-II

#### *Mobile OS Architectures*

Comparing and Contrasting architectures of all three – Android, iOS and Windows, Underlying OS (Darwin vs. Linux vs. Win 8), Kernel structure and native level programming, Runtime (Objective-C vs. Dalvik vs. WinRT), Approaches to power management, Security

#### ***Basic Design***

Introduction, Basics of embedded systems design, Embedded OS Design constraints for mobile applications, both hardware and software related, architecting mobile applications, user interfaces for mobile applications, touch events and gestures, Achieving quality constraints, performance, usability, security, availability and modifiability.

### UNIT-III

#### *Advanced Design*

Designing applications with multimedia and web access capabilities, Integration with GPS and social media

networking applications, accessing applications hosted in a cloud computing environment, Design patterns for mobile applications.

### ***Technology I - Android***

Introduction, Establishing the development environment, Android architecture, Activities and views , Interacting with UI , Persisting data using SQLite , Packaging and deployment , Interaction with server side applications, Using Google Maps, GPS and Wi-Fi, Integration with social media applications.

## **UNIT-IV**

### **Technology II - iOS**

Introduction to Objective C, iOS features, UI implementation, Touch frameworks , Data persistence using Core Data and SQLite , Location aware applications using Core Location and Map Kit , Integrating calendar and address book with social media application, Using Wi-Fi iPhone marketplace.

### ***Mobile Device Security***

Mobile malware, Device protections, iOS “Jailbreaking”, Android “rooting” and Windows’ “defenestration”.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Professional Mobile Application Development	Jeff McWherter and Scott Gowell	Wrox
2	Android in Practice	Charlie Collins, Michael Galpin and Matthias Kappler	DreamTech
3	Beginning iOS 6 Development: Exploring the iOS SDK	David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson	Apress

<b>Course Code</b>	<b>CSE371</b>
<b>Course Title</b>	<b>Introduction to Internet of Things</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	NIL
<b>Course Objectives</b>	The Internet is evolving to connect people to physical things and physical things to other physical things all in real time. It's becoming the Internet of Things (IoT). The course enables students to understand the basics of Internet and protocols. It introduces some of the application areas where Internet of Things can be applied.
<b>Course Outcome (CO)</b>	At the end of the course the learner will be able to-  <ol style="list-style-type: none"> <li>1. Understand and describe Functional blocks of IOT</li> <li>2. Explain MAC protocol and various routing protocols</li> <li>3. Describe data aggregation and data dissemination</li> <li>4. Evaluate and explain challenges in IoT design</li> <li>5. Demonstrate the ability to develop applications through IoT tools</li> </ol>

## UNIT-I

Introduction to IoT Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models and APIs

## UNIT-II

IoT & M2M Machine to Machine, Difference between IoT and M2M, Software define Network, Network and Communication aspects Wireless medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment and Node discovery, Data aggregation and dissemination

## UNIT-III

Challenges in IoT Design challenges, Development challenges, Security challenges, other challenges Domain specific applications of IoT Home automation, Industry applications, Surveillance applications, Other IoT applications

## UNIT-IV

Developing IoTs Introduction to Python, Introduction to different IoT tools, developing applications through IoT tools, developing sensor-based application through embedded system platform, Implementing IoT concepts with python

## RECOMMENDED BOOKS

<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	The Internet of Things in the Cloud:A Middleware Perspective	Honbo Zhou	CRC Press,2012

<b>Course Code</b>	<b>CAI307</b>
<b>Course Title</b>	Pattern Recognition and neural network
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	NIL
<b>Course Objectives(CO)</b>	The learner will be able to 1. Understand the requirements of the project 2. Prepare Report 3. Present Findings before the department

## SYLLABUS

### UNIT I

Principles of pattern recognition: Uses, mathematics, Classification and Bayesian rules, Clustering vs classification, Basics of linear algebra and vector spaces, Eigen values and eigen vectors, Rank of matrix and SVD

### UNIT II

Bayesian decision theory, Classifiers, Discriminant functions, Decision surfaces, Parameter estimation methods, Hidden Markov models, dimension reduction methods, Fisher discriminant analysis, Principal component analysis, non-parametric techniques for density estimation, non-metric methods for pattern classification, unsupervised learning, algorithms for clustering: K-means, Hierarchical and other methods

### UNIT III

Problem statement and uses, Branch and bound algorithm, Sequential forward and backward selection, Cauchy Schwartz inequality, Feature selection criteria function: Probabilistic separability based and Interclass distance based, Feature Extraction: principles

Human visual recognition system, Recognition methods: Low-level modelling (e.g. features), Mid-level abstraction (e.g. segmentation), High-level reasoning (e.g. scene understanding); Detection/Segmentation methods; Context and scenes, Importance and saliency, Large-scale search and recognition, Egocentric vision, systems, Human-in-the-loop interactive systems, 3D scene understanding.

### UNIT IV

Comparison between performance of classifiers, Basics of statistics, covariance and their properties, Data condensation, feature clustering, Data visualization, Probability density estimation, Visualization and Aggregation, FCM and soft-computing techniques, Examples of real-life datasets.

<b>Course Code</b>	<b>CSE373</b>
<b>Course Title</b>	Cloud Computing
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Distributed System, Operating Systems and Networking
<b>Course Objectives</b>	This Course work provides the complete understanding of Cloud system, its implementation techniques and its various applications in the field of computer Science.
<b>Course Outcome (CO)</b>	The learner will be able to- <ol style="list-style-type: none"> <li>1. Understand characteristics and types of cloud computing</li> <li>2. Describe architecture of cloud computing</li> <li>3. Explain applications of cloud</li> <li>4. Demonstrate their knowledge of cloud computing to real world examples</li> </ol>

## **SYLLABUS**

### **UNIT-I**

Cloud Computing Basics, History of Cloud Computing, Importance, Characteristics of Cloud Computing, Benefits and Challenges to Cloud architecture.

### **UNIT-II**

Types of Cloud: Public Cloud, Private Cloud, Hybrid and Community Cloud. Differences between public and private cloud, Status of Cloud Computing in India, Cloud Service Models, Role of virtualization in enabling the cloud; Differences between Grid computing and cloud computing, differences between grid computing and utility computing, Cloud Computing security concerns and proposed security model for future cloud computing.

### **UNIT-III**

Cloud Computing- Logical architecture, developing holistic Cloud Computing Reference Models- Seven step model of migrating to cloud. Virtualization types, Virtual Machine Life Cycle, Virtualization applications, Pitfalls of Virtualization, CPU Virtualization

### **UNIT-IV**

Case Study of Cloud Computing, Cloud Computing Risks. Cloud Tools, Cloud Applications, Future Trends, Mobile cloud, Jungle Computing, Big Data –Features and applications

<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Cloud Computing – A Practical Approach	Anthony T. Velte, Toby J. Velte and Robert E	TMH
2	Cloud Computing – Web based Applications	Michael Miller	Pearson Publishing

<b>Course Code</b>	<b>CAI309</b>
<b>Course Title</b>	Generative AI & Large Language Models
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic Knowledge of AI
<b>Course Objectives</b>	<ul style="list-style-type: none"> <li>• To understand the fundamentals of Generative AI and Large Language Models (LLMs).</li> <li>• To study neural networks, transformer architecture, and text generation techniques.</li> <li>• To explore training, fine-tuning, and deployment of generative AI models.</li> <li>• To analyze applications, ethical issues, and emerging trends in Generative AI systems.</li> </ul>
<b>Course Outcome (CO)</b>	<ul style="list-style-type: none"> <li>• Understand the concepts and architecture of Generative AI and LLMs.</li> <li>• Apply transformer-based techniques for text and content generation tasks.</li> <li>• Analyze training, fine-tuning, and deployment methods for LLMs.</li> <li>• Evaluate ethical considerations and real-world applications of Generative AI.</li> </ul>

## Syllabus

### Unit I: Introduction to Generative AI

Basics of Generative AI, Evolution of AI and Generative Models, Introduction to Large Language Models, Neural Networks and Deep Learning Basics, Natural Language Processing Fundamentals, Applications of Generative AI, AI Tools and Platforms

### Unit II: Transformer Architecture and LLMs

Introduction to Transformers, Attention Mechanism, Encoder and Decoder Models, Pre-trained Language Models, Text Generation Techniques, Prompt Engineering Basics, Fine-Tuning of LLMs

### Unit III: Generative AI Applications

Conversational AI and Chatbots, Text Summarization and Translation, Image and Content Generation, Code Generation Systems, Retrieval-Augmented Generation (RAG), AI Assistants and Automation, Evaluation Metrics for Generative Models

### Unit IV: Ethics, Security, and Future Trends

Bias and Fairness in AI, Privacy and Security Issues, Responsible AI Practices, Hallucination and AI Reliability, Deployment of Generative AI Models, Multimodal AI Systems, Future Trends in Generative AI and LLMs

<b>Sr. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	<i>Generative Deep Learning</i>	David Foster	O'Reilly Media
2	<i>Natural Language Processing with Transformers</i>	Lewis Tunstall, Leandro von Werra, Thomas Wolf	O'Reilly Media
3	<i>Deep Learning</i>	Ian Goodfellow, Yoshua Bengio, Aaron Courville	MIT Press

# **6<sup>th</sup> Semester**

<b>Course Code</b>	<b>CSE352</b>
<b>Course Title</b>	<b>Internet Web Programming</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of Program Development and Programming Language Constructs
<b>Course Objectives</b>	This course introduces advanced programming skills for website design. Dynamic content development will be explored through state-of-the-art programming languages for the creation of interactive websites. Students will create web pages that utilize the most current advances in web development.
<b>Course Outcomes (CO)</b>	The students will be able to: CO1: Understand concepts of Internet, WWW, Email, and HTML. CO2: Perform programs related to forms, table, and CSS using HTML tags. CO3: Implement the concepts of JavaScript, and DOM. CO4: Implement PHP programs, and MySQL commands using PHPMyAdmin.

## SYLLABUS

### UNIT-I

**Internet and WWW:** Introduction to internet and its applications, Email, telnet, FTP, ecommerce, video conferencing, e-business. Internet service providers, domain name server, internet address World Wide Web (WWW): World Wide Web and its evolution, uniform resource locator (URL), browsers - internet explorer, Netscape navigator, opera, Firefox, chrome, Mozilla. Search engine, web saver - Apache, IIS, proxy server, HTTP protocol.

**HTML and Graphics:** HTML Tag Reference, Global Attributes, Event Handlers, Document Structure Tags, Formatting Tags, Text Level formatting, Block Level formatting, List Tags, Hyperlink tags, Image and Image maps, Table tags, Form Tags, Frame Tags, Executable content tags.

**Image maps:** Client-side Image maps, Server-side Image maps, Using Server-side and Client-side Image maps together, alternative text for Image maps,

**Tables:** Introduction to HTML tables and their structure, the table tags, Alignment, Aligning entire Table, Alignment within a row, Alignment within a cell, Attributes, Content Summary, Background colour, adding a Caption, Setting the width, adding a border, Spacing within a cell, Spacing between the cells, spanning multiple rows or columns, Elements that can be placed in a table, Table Sections and column properties, Tables as a design tool

### UNIT-II

**Frames:** Introduction to Frames, Applications, Frames document, The <FRAMESET> tag, Nesting <FRAMESET> tag, placing content in frames with the <FRAME> tag, Targeting named frames, creating floating frames, Using Hidden frames,

**Forms:** Creating Forms, the <FORM> tag, Named Input fields, The <INPUT> tag, Multiple lines text windows, drop down and list boxes, Hidden, Text, Text Area, Password, File Upload, Button, Submit, Reset, Radio, Checkbox, Select, Option, Forms and Scripting, Action Buttons, Labelling input files, Grouping related fields, Disabled and read-only fields, Form field event handlers, Passing form data

**Style Sheets:** What are style sheets? Why are style sheets valuable? Different approaches to style sheets, Using Multiple approaches, linking to style information in separate file, Setting up style information, Using the <LINK> tag, embedded style information, Using <STYLE> tag, Inline style information.

### UNIT-III

**Java Script:** Introduction, Client-Side JavaScript, Server-Side JavaScript, JavaScript Objects, JavaScript Security, Operators: Assignment Operators, Comparison Operators, Arithmetic Operators, % (Modulus), ++ (Increment), -- (Decrement), - (Unary Negation), Logical Operators, Short-Circuit Evaluation, String Operators, Special Operators, (Conditional operator),(Comma operator), delete, new, this, void.

**Statements:** Break, comment, continue, delete, do ... while, export, for, for...in, function, if...else, import, labelled, return, switch, var, while, with, Core JavaScript (Properties and Methods of Each): Array, Boolean, Date, Function, Math, Number, Object, String, reg Exp

**Document and its associated objects:** document, Link, Area, Anchor, Image, Applet, Layer Events and Event Handlers: General Information about Events, Defining Event Handlers, event, on Abort, on Blur, on Change, on Click, on Dbl Click, on Drag Drop, on Error, on Focus, on Key Down on Keypress, on Key Up, on Load, on Mouse Down, on Mouse Move, on Mouse Out, on Mouse Over, on Mouse Up, on Move, on Reset, on Resize, on Select, on Submit, on Unload

### UNIT-IV

**XML:** Introduction to XML, Anatomy of an XML, document, Creating XML Documents, Creating XML DTDs, XML Schemas, XSL

**PHP:** Why PHP and MySQL? Server-side web scripting, Installing PHP, Adding PHP to HTML, Syntax and Variables, Passing information between pages, Strings, Arrays and Array Functions, Numbers, Basic PHP errors / problems.

**Advanced PHP and MySQL:** PHP/MySQL Functions, Displaying queries in tables, Building Forms from queries, String and Regular Expressions, Sessions, Cookies and HTTP, Type and Type Conversions, E-Mail.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	HTML 5 in simple steps Kogent Learning	Dreamtech press Solutions Inc.	<b>Dreamtech Press</b>
2	HTML: Beginner's guide Wendy Willard Mc Graw Hill	<u>Wendy Willard</u>	<b>Osborne/McGraw-Hill</b>
3	Managing software process	Watts Humphrey	Pearson education
4	Software Engineering – An Engineering Approach	James F. Peters and WitoldPedrycz	Wiley

<b>Course Code</b>	<b>CSE354</b>
<b>Course Title</b>	<b>Software Engineering</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Overview of Structure and Software Analysis and Design
<b>Course Objectives</b>	This Course Work provides the thorough understanding of the software engineering concepts and it also gives the ideas of handling the projects in the organizations and in institutes
<b>Course Outcome (CO)</b>	The learner will be able to- 1. Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics 2. Communicate effectively with a range of audiences 3. Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors 4. Analyze, design, verify, validate, implement, apply, and maintain software systems

### SYLLABUS

#### UNIT-I

**Introduction:** Software Engineering definition, history, evolution of software, software components, applications, software myths, software crisis.

**Software Development Lifecycle:** Requirements analysis, software design, coding, testing, maintenance

**Software Process Models:** Waterfall model, prototyping, interactive enhancement, spiral model. Role of Management in software development. Role of metrics and measurement.

#### UNIT-II

**Software Requirement Specification:** Problem analysis, requirement specification, validation, metrics, monitoring and control, SRS

#### UNIT-III

**Coding:** Top-down and bottom-up, structured programming, information hiding, programming style, and internal documentation. Verification, Metrics, monitoring and control.

**Testing:** Levels of testing functional testing, structural testing, test plan, test cases specification, and reliability assessment.

#### UNIT-IV

**Software Project Management:** Cost estimation, Project scheduling, Staffing, Software configuration management, Quality assurance, SIX SIGMA, Project Monitoring, Risk management, reverse engineering CASE Tools

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Engineering: A Practitioners Approach	Roger Pressman	McGraw Hill
2	Software Engineering	Sommerville	Adison Wesley
3	Managing software process	Watts Humphrey	Pearson education
4	Software Engineering – An Engineering Approach	James F. Peters and WitoldPedrycz	Wiley

<b>Course Code</b>	<b>CSE356</b>
<b>Course Title</b>	Programming in Java
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Knowledge of OOPs
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.</li> <li>2. Be aware of the important topics and principles of software development.</li> <li>3. Be able to use the Java SDK environment to create, debug and run simple Java programs.</li> <li>4. Understand the principles of inheritance, packages and interfaces</li> </ol>
<b>Course Outcome</b>	<p>The learner will be able to-</p> <ol style="list-style-type: none"> <li>1. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.</li> <li>2. Read and make elementary modifications to Java programs that solve real-world problems.</li> <li>3. Validate input in a Java program.</li> <li>4. Identify and fix defects and common security issues in code.</li> </ol>

## SYLLABUS

### UNIT-I

Object oriented programming concepts, objects, classes, methods and messages, abstraction a encapsulation, inheritance, abstract classes, polymorphism, Objects a classes in Java, defining classes, methods, access specifiers, static members, constructors, finalizemeth

### UNIT-II

Arrays, Strings, Packages, Java-Doc comments, Inheritance, class hierarchy, polymorphism, dynam binding, final keyword, abstract classes

### UNIT-III

The Object class, Reflection, interfaces, object cloning, inner classes, proxies, I/O Streams, Graphi programming, Frame, Components, working with 2D shapes.

### UNIT-IV

Basics of event handling, event handlers, adapter classes, actions, mouse events – AWT event hierarchy, introduction to Swing, Model-View-Controller design pattern – buttons, layout management, Swing Components, exception handling, exception hierarchy, throwing and catching exceptions.

<b>Course Code</b>	<b>MDC018</b>
<b>Course Title</b>	<b>Gender, Culture and Development</b>
<b>Type of Course</b>	OE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisite</b>	None
<b>Course Objectives (CO)</b>	The objective of this course is to build an understanding and initiate and strengthen programs combating gender-based violence and discrimination
<b>Course Outcomes</b>	Upon completion of this course, students will be able to- 1. Understand basic gender concepts. 2. Explain gender roles and relationships matrix. 3. Identify Gender-based violence from a human rights perspective 4. Develop relationship between gender, development and violence

### *SYLLABUS*

#### **UNIT-I**

##### **Introduction to Gender**

1. Definition of Gender
2. Basic Gender Concepts and Terminology
3. Exploring Attitudes towards Gender
4. Social Construction of Gender

#### **UNIT-II**

##### **Gender Roles and Relations**

1. Types of Gender Roles
2. Gender Roles and Relationships Matrix
3. Gender-based Division and Valuation of Labour

#### **UNIT-III**

##### **Gender Development Issues**

1. Identifying Gender Issues
2. Gender Sensitive Language
3. Gender, Governance and Sustainable Development
4. Gender and Human Rights

##### **Gender-based Violence**

1. The concept of violence
2. Types of Gender-based violence
3. The relationship between gender, development and violence
4. Gender-based violence from a human rights perspective

#### **UNIT-IV**

##### **Gender and Culture**

1. Gender and Film
2. Gender and Electronic Media
3. Gender and Advertisement
4. Gender and Popular Literature

<b>Course Code</b>	<b>CAI302</b>
<b>Course Title</b>	<b>Deep Learning</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Overview of Structure and Software Analysis and Design
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. Make students familiar with basic concepts and tool used in neural networks</li> <li>2. Teach students structure of a neuron including biological and artificial</li> <li>3. Teach learning in network (Supervised and Unsupervised)</li> <li>4. Teach concepts of learning rules.</li> </ol>
<b>Course Outcomes</b>	<p>The learner will be able to</p> <ol style="list-style-type: none"> <li>1. Design single and multi-layer feed-forward neural networks</li> <li>2. Understand supervised and unsupervised learning concepts &amp; understand unsupervised learning using Kohonen networks</li> <li>3. Understand training of recurrent Hopfield networks and associative memory concepts.</li> </ol>

## SYLLABUS

### Unit I: Introduction

Structure of biological neurons relevant to ANNs., Models of ANNs; Feedforward & feedback networks; learning rules; Hebbian learning rule, perception learning rule, delta learning rule, Widrow-Hoff learning rule, correction learning rule, Winner-take-all learning rule, etc.

### Unit II: Single layer Perception Classifier and Multi-layer Feed forward Networks

Classification model, Features & Decision regions; training & classification using discrete perceptron, algorithm, single layer continuous perceptron networks for linearly separable classifications, linearly non-separable pattern classification, Delta learning rule for multi-perceptron layer, Generalized delta learning rule, Error back-propagation training, learning factors, Examples.

### Unit III: Single layer feedback Networks

Basic Concepts, Hopfield networks, Training & Examples. Associative memories: Linear Association, Basic Concepts of recurrent.

### Unit IV: Auto associative memory

Retrieval algorithm, storage algorithm; Bidirectional associative memory, Architecture, Association encoding & decoding, Stability.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Introduction to Artificial Neural systems	Jacek M. Zurada, 1994	Jaico Publ. house
2	Neural Network Fundamentals	N.K. Bose, P. Liang, 2002	M.H

<b>Course Code</b>	<b>CSE380</b>
<b>Course Title</b>	<b>Software Engineering Laboratory</b>
<b>Type of Course</b>	PC
<b>L T P</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisites</b>	Knowledge of Program Development Constructs
<b>Course Objectives</b>	This practical coursework allows the students to efficiently design a working software model.
<b>Course Outcome (CO)</b>	The students will be able to: CO1: Understand real time business requirements and design SRS documents and Use Case model CO2: Understand notations used in UML diagrams and design UML Class Diagram, Interaction diagrams

### **SYLLABUS**

Develop Use Case diagrams for selected Mini project

#### **1: Use Case Models**

- 1.1: To develop a problem statement.
- 1.2: Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).
- 1.3: Identify Use Cases and develop the Use Case model.

#### **2: UML Diagrams.**

- 2.1: Identify the conceptual classes and develop a domain model with UML Class diagram
- 2.2: Using the identified scenarios find the interaction between objects and represent them using Object diagrams.
- 2.3: Draw UML Interaction diagrams: Collaboration and sequence diagrams.
- 2.4: Draw the State Chart diagram.
- 2.5: Identify the business activities and develop an UML Activity diagram

#### **3: Implementations of Layers**

- 3.1: Draw Component diagrams.
- 3.2: Draw Deployment diagrams.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	UML 2 and the Unified Process: Practical Object- Oriented Analysis and Design	Jim Arlow, Ila Neustadt	Pearson Education
2	Practical Object-Oriented Design With UML	Priestley	Tata McGraw-Hill Education
3	Object-Oriented Software Engineering: Practical Software Development Using UML and Java	Lethbridge	Tata McGraw-Hill Education

<b>Subject Code</b>	<b>CSE382</b>
<b>Course Title</b>	<b>Internet Web Programming-Laboratory</b>
<b>Type of Course</b>	PC
<b>LTP</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisite</b>	Basic knowledge of Program Development and Programming Language Constructs
<b>Course Objectives</b>	This course introduces advanced programming skills for website design. Dynamic content development will be explored through state-of-the-art programming languages for the creation of interactive web sites. Students will create web pages that utilize the most current advances in web development.
<b>Course Outcomes (CO)</b>	The learner will be able to- 1. Implement interactive web page(s) using HTML, CSS and JavaScript. 2. Design a responsive web site using HTML5 and CSS3. 3. Describe and differentiate different Web Extensions and Web Services. 4. Build Dynamic web site using server-side PHP Programming and Database connectivity.

## SYLLABUS

### List of Practical's

1. Configuration and administration Apache Web Server.
2. Develop an HTML page to demonstrate the use of basic HTML tags,
3. Develop an HTML page to demonstrate Link to different HTML pages and link within a page, Insertion of images.
4. Implement HTML List tags
5. Implement HTML table tags.
6. Develop a registration form by using various form elements like input box, text area, radio buttons, Check boxes etc.
7. Develop HTML webpage for implementation of Frames.
8. Design an HTML page by using the concept of internal, inline, external style sheets.
9. Create an HTML file to implement the styles related to text, fonts, links using cascading style sheets
10. Create an HTML file to implement the concept of document object model using JavaScript
11. Create an HTML page including JavaScript that takes a given set of integer numbers and shows
12. them after sorting in descending order.
13. Create a PHP file to print any text using variable.
14. Demonstrate the use of Loops and arrays in PHP
15. Create a PHP file using GET and POST methods.
16. A simple calculator web application that takes two numbers and an operator (+, -, /, \* and %) from An HTML page and returns the result page with the operation performed on the operands.
17. Demonstrate the use of web site designing tools such as Joomla, WordPress. Implement at least one minor project using different technologies mentioned in theory of the subject.

<b>Subject Code</b>	<b>CSE384</b>
<b>Course Title</b>	<b>Programming in Java Laboratory</b>
<b>Type of Course</b>	PC
<b>LTP</b>	0 0 2
<b>Credits</b>	1
<b>Course Prerequisite</b>	NA

## LIST OF PRACTICALS

- Implement a Java program demonstrating the difference between procedure-oriented programming and object-oriented programming paradigms.
- Create a Java class to demonstrate the concepts of classes, objects, and object references.
- Develop a Java application showcasing abstraction and encapsulation principles.
- Design a Java program illustrating inheritance, including different types like single, multilevel, hierarchical, and hybrid.
- Write a Java application demonstrating method overriding and overloading for achieving polymorphism.
- Develop a Java program to showcase the use of constructors, constructor overloading, and constructor overriding.
- Implement a Java class demonstrating the usage of access modifiers (private, public, protected, default).
- Create a Java program illustrating the use of this keyword for referring to the current object.
- Develop a Java application showcasing the usage of the super keyword for invoking superclass constructors and methods.
- Write a Java program to demonstrate exception handling using try, catch, finally, throw, and throw keywords.
- Develop a Java application to showcase error and exception types, such as checked exceptions, unchecked exceptions, and errors.
- Implement a Java program demonstrating runtime polymorphism through overriding methods.
- Write a Java application illustrating compile-time polymorphism through method overloading.
- Create a Java program demonstrating the implementation of multiple inheritance using interfaces.
- Design a Java application showcasing the usage of abstract classes and interfaces, highlighting their differences and similarities.

<b>Course Code</b>	<b>CSE366</b>
<b>Course Title</b>	<b>Digital Image Processing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	This course has no prerequisite other than knowledge of probability and statistics, and Computer graphics.
<b>Course Objectives</b>	The objective of this course is to teach students the architecture of image processing. By taking this course, the students are expected to understand the basic algorithms and be able to apply these techniques.
<b>Course Outcome (CO)</b>	The learner will be able to- 1. Understand digital image processing 2. Understand the image enhancement.

## SYLLABUS

### *Unit-I*

Introduction to the DIP areas and applications; Components of Digital Image Processing; Elements of Visual Perception; Image Sensing and Acquisition; Image Sampling and Quantization; Relationships between pixels; color models.

### *Unit-II*

Image Enhancement Spatial Domain: Gray level transformations; Histogram processing; Basics of Spatial Filtering; Smoothing and Sharpening Spatial Filtering Frequency Domain: Introduction to Fourier Transform; Smoothing and Sharpening frequency domain filters; Ideal, Butterworth and Gaussian filters

### *Unit-III*

Image Restoration Course Code: Noise models; Mean Filters; Order Statistics; Adaptive filters; Band reject Filters; Band pass Filters; Notch Filters; Optimum Notch Filtering; Inverse Filtering; Wiener filtering

### *Unit-IV*

Feature Extraction and Image Segmentation Feature Extraction: Contour and shape dependent feature extraction, Extraction of textural features

Segmentation: Detection of Discontinuities; Edge Linking and Boundary detection; Region based segmentation; Morphological processing- erosion and dilation.

Image Compression and Encoding Entropy-based schemes, Transform-based encoding, Predictive encoding and DPCM, Vector quantization, Huffman coding.

<b>Course Code</b>	<b>CSE314</b>
<b>Course Title</b>	<b>Computer Vision</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Computer Graphics
<b>Course Objectives (CO)</b>	To familiarize the student with specific, well-known computer vision methods, algorithms and results. To understand the roles of image transformations and there in variances in pattern recognition and classification.
<b>Course Outcomes</b>	<p><b>The learner will be able to-</b></p> <ol style="list-style-type: none"> <li>1. Identify basic concepts, terminology, theories, models and methods in the field of computer vision</li> <li>2. Describe basic methods of computer vision related to multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition.</li> <li>3. Assess which methods to use for solving a given roblem.</li> <li>4. Analyze the accuracy of the methods</li> </ol>

## SYLLABUS

### UNIT I: Digital Image Formation and Low-Level Processing

This unit provides an overview of the state-of-the-art in digital image formation and processing. It covers the fundamentals of image formation and various geometric transformations including orthogonal, Euclidean, affine, and projective. Core concepts such as Fourier Transform, convolution, filtering, image enhancement, restoration, and histogram processing are discussed. The unit also delves into depth estimation and multi-camera views, addressing concepts such as perspective, binocular stereopsis, camera and epipolar geometry, homography, rectification, DLT, RANSAC, and 3D reconstruction frameworks including auto-calibration.

### UNIT II: Feature Extraction, Image Segmentation, and Pattern Analysis

This unit introduces various feature extraction methods such as edge detection (Canny, LOG, DOG), line detectors using Hough Transform, and corner detection using Harris and Hessian Affine methods. Advanced descriptors like SIFT, SURF, HOG, GLOH, and scale-space analysis techniques using Gaussian derivative filters, Gabor filters, and DWT are covered. In image segmentation, methods like region growing, edge-based segmentation, graph cuts, mean-shift, MRFs, texture segmentation, and object detection are explored. Pattern analysis includes clustering techniques (K-Means, K-Medoids, Mixture of Gaussians), classification (discriminant functions, supervised, unsupervised, semi-supervised), and classifiers like Bayes, KNN, ANN. Dimensionality reduction methods like PCA, LDA, ICA, and non-parametric techniques are also Introduced

### UNIT III: Motion Analysis and Shape Reconstruction

This unit focuses on motion analysis techniques including background subtraction and modeling, optical flow, KLT tracking, spatio-temporal analysis, dynamic stereo, and motion parameter estimation. It also explores shape-from-X techniques, such as shape from shading, texture, color, motion, and edges. Concepts like the Phong model, reflectance maps, albedo estimation, photometric stereo, and the use of surface smoothness constraints are discussed. The unit concludes with perceptual organization and cognition, modeling vision as a graphic process in the brain, insights from neurological trauma and visual deficits, and phenomena like visual agnosias and illusions.

### UNIT IV: Model Estimation and Applications

This unit explores model estimation using machine learning and statistical methods in computer vision. It discusses both discriminative and generative models and their applications in areas like content-

based image retrieval (CBIR). Additional applications include content-based video retrieval (CBVR), activity recognition, computational photography, biometrics, image stitching, and document processing. The unit also highlights modern trends such as super-resolution, GPU-based acceleration, augmented reality, cognitive models, fusion techniques, and sparse representation and compressed sensing (SR & CS).

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Computer Vision: Algorithms and Applications	Richard Szeliski	Springer
2	Computer Vision: A Modern Approach	D. A. Forsyth, J.Ponce	Prentice Hall
3	Introductory Techniques for 3D Computer Vision	Trucco and Verri	Prentice Hall
4	Computer vision	Shapiro, L. &Stockman, G	Prentice Hall
5	Three-dimensional Computer Vision: A geometric approach	Olivier Faugeras	Olivier Faugeras

<b>Course Code</b>	<b>CSE362</b>
<b>Course Title</b>	<b>Compiler Construction</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic understanding of programming languages, data structures, and machine architecture.
<b>Course Objectives (CO)</b>	<ol style="list-style-type: none"> <li>1. To understand and list the different stages in the process of compilation.</li> <li>2. Identify different methods of lexical analysis.</li> <li>3. Design top-down and bottom-up parsers.</li> <li>4. Identify synthesized and inherited attributes.</li> <li>5. Develop syntax -directed translation schemes.</li> <li>6. Develop algorithms to generate code for a target machine.</li> </ol>
<b>Course Outcome(CO)</b>	<p>The learner will be able to:</p> <ol style="list-style-type: none"> <li>1. For a given grammar specification, develop the lexical analyser.</li> <li>2. For a given parser specification, design top-down and bottom-up parsers.</li> <li>3. Develop syntax-directed translation schemes.</li> <li>4. Develop algorithms to generate code for a target machine.</li> </ol>

## SYLLABUS

### UNIT I

Introduction: Phases of compilation and overview. Lexical Analysis (Scanner): Regular languages, finite automata, regular expressions, converting regular expressions to finite automata, scanner generators (lex, flex). Syntax Analysis (Parser): Context-free languages and grammars, push-down automata, LL(1) grammars and top-down parsing, operator grammars, LR(0), SLR(1), LR(1), LALR(1) grammars and bottom-up parsing, ambiguity and LR parsing, LALR(1) parser generators (yacc, bison).

### UNIT II

Semantic Analysis: Attribute grammars, syntax-directed definitions, evaluation and flow of attributes in a syntax tree. Symbol Table: Structure, symbol attributes, and management. Run-Time Environment: Procedure activation, parameter passing, value return, memory allocation, and scope.

### UNIT III

Intermediate Code Generation: Translation of different language features, types of intermediate representations.

Code Improvement (Optimization): Control-flow analysis, data-flow analysis, dependence analysis; Local optimization, global optimization, loop optimization, peephole optimization.

### UNIT IV

Architecture Dependent Code Improvement: Instruction scheduling (for pipelines), loop optimization (for cache memory), etc.

Register Allocation and Target Code Generation

Advanced Topics: Type systems, data abstraction, compilation of object-oriented features, and non-imperative programming languages

**RECOMMENDED BOOKS**

<b>Sr.no.</b>	<b>Name</b>		<b>Publisher</b>
1	Compilers Principles, Techniques, &Tools	A.V.Aho, R. Sethi &J.D. Ullman	Pearson Education
2	Engineering a Compiler	Keith Cooper and LindaTorczon,	Morgan-Kaufman Publishers

<b>Course Code</b>	<b>CAI306</b>
<b>Course Title</b>	<b>MLOps (Machine Learning Operations)</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic understanding of ML
<b>Course Objectives (CO)</b>	<ul style="list-style-type: none"> <li>• To understand the fundamentals of Machine Learning Operations and deployment pipelines.</li> <li>• To study data management, model training, monitoring, and automation techniques in MLOps.</li> <li>• To explore cloud platforms, CI/CD practices, and containerization for ML systems.</li> <li>• To analyze scalable, reliable, and secure deployment of machine learning applications.</li> </ul>
<b>Course Outcome(CO)</b>	<ul style="list-style-type: none"> <li>• Understand the concepts and workflow of Machine Learning Operations.</li> <li>• Apply tools and techniques for model deployment, monitoring, and automation.</li> <li>• Analyze CI/CD pipelines and cloud-based infrastructure for ML applications.</li> <li>• Evaluate performance, scalability, and security aspects of deployed ML systems.</li> </ul>

### Unit I: Introduction to MLOps

Basics of Machine Learning Operations, MLOps Lifecycle, ML Workflow and Pipelines, Data Collection and Preprocessing, Version Control for ML, Introduction to DevOps and MLOps, Tools and Platforms for MLOps

### Unit II: Model Development and Deployment

Model Training and Validation, Experiment Tracking, Model Packaging and Deployment, Containerization using Docker, Virtualization and Cloud Platforms, API-Based Model Serving, Continuous Integration and Continuous Deployment (CI/CD)

### Unit III: Monitoring and Automation

Model Monitoring Techniques, Performance Evaluation Metrics, Data Drift and Model Drift, Workflow Automation, Pipeline Orchestration, Logging and Alerting Systems, Model Retraining Strategies

### Unit IV: Scalability, Security, and Emerging Trends

Scalable ML Infrastructure, Cloud-Based MLOps, Security and Privacy in MLOps, Governance and Responsible AI, Edge Deployment of ML Models, AutoML and Automated Pipelines, Future Trends in MLOps and AI Deployment

<b>Sr. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	<i>Introducing MLOps</i>	Mark Treveil, Nicolas Omont, Clément Stenac	O'Reilly Media
2	<i>Practical MLOps</i>	Noah Gift, Alfredo Deza	O'Reilly Media
3	<i>Machine Learning Engineering</i>	Andriy Burkov	True Positive Inc.

<b>Course Code</b>	<b>CSE348</b>
<b>Course Title</b>	<b>Digital Marketing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Nil
<b>Course Objective</b>	The main objective of this course is to provide learners with the knowledge of business advantages of digital marketing and its importance for marketing success; to develop a digital marketing plan; to make SWOT analysis; to define a target group; to get introduced to various digital channels, their advantages and ways of integration;
<b>Course Outcomes</b>	The learner will be able to- <ol style="list-style-type: none"> <li>1. Identify the importance of digital marketing for marketing success,</li> <li>1. Manage customer relationships across all digital channels and build better customer relationships,</li> <li>2. Create a digital marketing plan, starting from the SWOT analysis and defining a target group, then identifying digital channels, their advantages and limitations.</li> <li>3. Perceiving ways of integration taking into consideration the available budget.</li> </ol>

## **SYLLABUS**

### **UNIT I**

Introduction: Marketing and its definition, Digital Marketing, How we do Marketing, Benefits of Digital marketing, Digital marketing platforms and Strategies, Defining Marketing Goals, Latest Digital marketing trends, introduction to traditional and new methods of marketing Requirement: Requirements for digital marketing, its uses.

### **UNIT II**

Search Engine Optimization: Introduction to Search Engines, How the search engine works, Components of Search Engines. Keyword Research and Competition: Introduction to Keyword Research, Types of Keywords, Keyword Research Methodology, Business Analysis & Categorization, Google Keyword Planner, Market Research and Analysis, New Keyword Ideas, Competition Analysis, Finalizing the Keywords List.

### **UNIT III**

Onpage Optimization: Introduction to Onpage ,What is Webmaster Tools, Selecting Target Location, Onpage Analysis Methodology, Fundamental On-page Factors , Website Speed , Domain name in SEO, URL Optimization , Title Tag Optimization , Meta Tags Optimization, Content Optimization , Sitemaps Generation , Using Robot.txt in Site URL , Redirecting Techniques , Canonical Links , Rich Snippets.

### **UNIT IV**

Offpage Optimization : What is Link Building , Types of Linking Methods , DoFollow Vs. NoFollow Link building Guidelines , Linking Building Methodology , Links Analysis Tools , Directory Submissions , Local Business Directories , Social Bookmarking , Using Classifieds for Inbound traffic , Question and Answers , Blogging & Commenting , Guest Blogging Local SEO: What is Local SEO, Importance of Local SEO , Submission to Google My Business , Completing the Profile , Local SEO Ranking Signals , Local SEO, Negative Signals, Citations and Local Submissions

<b>RECOMMENDED BOOKS</b>			
<b>S.No</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
<b>1</b>	<b>Digital Marketing For Dummies</b>	<b>Ryan Deiss &amp; Russ Henneberry</b>	<b>John Wiley &amp; Sons, Inc.</b>
<b>2</b>	<b>Social Media Marketing All-in-one Dummies</b>	<b>Jan Zimmerman, Deborah Ng</b>	<b>John Wiley &amp; Sons Inc, 4<sup>th</sup> edition</b>

<b>Course Code</b>	<b>CSE378</b>
<b>Course Title</b>	<b>Advanced Parallel Computing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of Computer System Architecture
<b>Course Objectives</b>	Students become familiar with parallel computer architecture and algorithms.
<b>Course Outcome (CO)</b>	The learner will be able to- 1. Understand basic terms used in parallel computing 2. Classify parallel computers 3. Describe parallel computer architecture 4. Analyze parallel algorithms

## SYLLABUS

### UNIT-I

**Introduction to Parallel Computing:** Basic concepts about program/process/ thread, Concurrent Execution, Parallel Execution, granularity, Potential of Parallelism; Need of Parallel Computation; Levels of parallel processing; Parallel processing Vs. Parallel computing; Dataflow Computing concept; Applications of parallel processing-Scientific Applications / Image processing, Engineering Application, Database query / Answering applications, AI Applications, Mathematical simulations and modeling

**Classification of Parallel Computers:** Types of Classification; Flynn's/ Handler classification; UMA / NUMA /COMA; Loosely coupled / tightly coupled; Classification based grain size and Instruction level parallelism.

### UNIT-II

**Interconnection Network:** Need of Interconnection Network; Concept Bandwidth Nod degree diameter bisection bandwidth, In degree and Out degree; Static and Dynamic Interconnection network; Omega, Parallel Shifter, Bens, permutation, hypercube, butterfly; Shuffle exchange Network

**Parallel Computer Architecture:** Introduction to various computer architecture; Pipeline processing; Vector / Array processing; VLIW and Super scalar architecture; Associative architecture -Multithreaded architecture.

### UNIT-III

#### **Parallel Algorithm & Parallel Programming**

**Parallel Algorithm:** Introduction to Parallel Algorithms; Analysis of Parallel Algorithms; Different models of computation- Combinational circuit, Permutation Circuit, Sorting circuit, Matrix computation.

**PRAM Algorithms:** Message passage programming -Shared memory, Message passing libraries, Data Parallel programming; Data Structures for parallel algorithms-Link list, Arrays pointers, Hypercube network.

**Parallel Programming:** Introduction to Parallel Programming; Types of parallel programming - Programming based on message passing, data parallelism, Programming forshared memory systems, Example programs for parallel systems.

### UNIT- IV

#### **Advanced Topics**

**Operating System for Parallel Computers:** Basic issues of Operating Systems for Parallel Computers; Process Management; Resource Management; Memory management; I/O Management; Inter-Processor Communication; Vectorization Compiler

**Performance Evaluation:** Introduction to performance evaluation; Metric of Parallel overhead; Law

Speedup; Measurement Tools

**Recent Trends:** Multi-component CPU; Apex architecture IA 64; Hyper threading

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Advanced Computer Architecture: Parallelism, Scalability, Programmability	Hwang, K	Tata McGraw Hills
2	Introduction to Parallel Processing	Sasikumar M., Shikhare, D., Ravi Prakash	Prentice Hall of India pvt.ltd. New Delhi
3	Computer Architecture and Parallel Processing	Hwang, K., Briggs, F. A.	McGraw Hill

<b>Course Code</b>	<b>CAI304</b>
<b>Course Title</b>	<b>Machine Learning</b>
<b>Type of Course</b>	<b>PE</b>
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	<b>Machine Learning</b>
<b>Course Objectives</b>	To understand learning models and learning algorithms
<b>Course Outcomes(CO)</b>	The learner will be able to- 1. Recognize the characteristics of machine learning that make it useful to real-world problems. 2. Characterize and differentiate between supervised and unsupervised learning techniques. 3. Explain Reinforcement learning and its control 4. Represent concepts of Decision trees.

## SYLLABUS

### UNIT I

Introduction- Basic concepts, machine learning problems, types of learning, designing a learning system, Goals and applications of machine learning

Learning Theory- Bias/variance tradeoff. Union and Chernoff/Hoeffding bounds. VC dimension, Worst case (online) learning, learning algorithms.

### UNIT II

Supervised learning - Supervised learning setup, LMS, Logistic regression, Perceptron, Exponential family, Generative learning algorithms, Gaussian discriminant analysis, Naive Bayes, Support vector machines, Model selection and feature selection.

Unsupervised learning - Clustering. K-means, EM, Mixture of Gaussians, Factor analysis, PCA (Principal components analysis), ICA (Independent components analysis).

### UNIT III

Reinforcement learning and control - MDPs, Bellman equations, Value iteration and policy iteration, Linear quadratic regulation (LQR), LQG, Q-learning. Value function approximation, Policy search, Reinforce, POM.

### UNIT IV

Decision Tree Learning, Representing concepts as decision trees, Recursive induction of decision trees, Picking the best splitting attribute: entropy and information gain, Searching for simple trees and computational complexity, Occam's razor, Overfitting, noisy data.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Pattern Recognition and Machine Learning	Bishop, C.	Berlin: Springer-Verlag.
2	Elements of Statistical Learning	Hastie, Tibshirani, and Friedman	Springer
3	Machine Learning	Tom Mitchell	Mc-GrawHill

<b>Course Code</b>	<b>CSE322</b>
<b>Course Title</b>	<b>Distributed System</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of object-oriented programming, data structures, threads, operating system concepts.
<b>Course Objectives</b>	This Course provides the complete understanding of distributed system and its various applications in the field of Computer Science.
<b>Course Outcome(CO)</b>	<b>The learner will be able to –</b> <ol style="list-style-type: none"> <li>1. Identify characteristics of distributed system.</li> <li>2. Explain the system models of distributed processing and communication.</li> <li>3. Explain distributed deadlock detection.</li> <li>4. Explain distributed transaction and its types.</li> </ol>

## SYLLABUS

### UNIT-I

**Characterization of Distributed Systems:** Introduction, Examples of distributed Systems, Issues in Distributed Operating Systems, Resource sharing and the Web, Challenges.

**System Models:** Architectural models, Fundamental Models.

**Theoretical Foundation for Distributed System:** Limitation of Distributed system, absence of global clock, shared memory, Logical clocks, Lamport's & vector logical clocks, Causal ordering of messages, global state, and termination detection.

### UNIT-II

**Distributed Mutual Exclusion:** Classification of distributed mutual exclusion, requirement of mutual exclusion theorem, Token based and non-token based algorithms, performance metric for distributed mutual exclusion algorithms.

**Distributed Deadlock Detection:** System model, resource vs communication deadlocks, deadlock prevention, avoidance, detection & resolution, centralized deadlock detection, distributed deadlock detection, path pushing algorithms, edge chasing algorithms.

### UNIT-III

**Agreement Protocols:** Introduction, System models, classification of Agreement Problem – Interactive consistency Problem, Applications of Agreement algorithms.

**Distributed Objects and Remote Invocation:** Communication between distributed objects, Remote procedure call, Events and notifications, Java RMI case study.

**Transactions and Concurrency Control:** Transactions, Nested transactions, Locks, Optimistic, Concurrency control, Timestamp ordering, Comparison of methods for concurrency control.

### UNIT-IV

**Distributed Transactions:** Introduction, Flat and nested distributed transactions, Atomic commit protocols, concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery.

**Distributed Shared Memory:** Design and Implementation issues, consistency models, CORBA Case Study: CORBA RMI, CORBA services.

**File System:** File service components, design issues, interfaces, implementation techniques, Sun Network File System – architecture and implementation, other distributed filesystems – AFS, CODA.

**Name Services:** SNS name service model.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Advanced Concepts in Operating Systems	Mukesh Singhal &Niranjan G Shivaratri	TataMc GrawHill
2	Distributed System: Concepts and Design	Coulouris, Kindberg Dollimore,	Pearson Education

<b>Course Code</b>	<b>CSE324</b>
<b>Course Title</b>	<b>Wireless Communications</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Nil
<b>Course Objective</b>	To gain an understanding of the principles behind the design of wireless communication systems and technologies.
<b>Course Outcomes</b>	<p>1. <b>The learner will be able to</b> -</p> <p><b>1. Understand and explain the Classification of mobile communication systems.</b></p> <p><b>2. Examine state-of-the-art distributed systems, such as Google File System.</b></p> <p><b>3. Learn the principles, architectures, algorithms and programming models used in distributed systems.</b></p>

## SYLLABUS

### UNIT-I

Introduction: A basic cellular system, performance criteria, operation of cellular systems, planning a cellular system, analog & digital cellular systems.

Examples of Wireless Communication Systems: Paging Systems, Cordless Telephone Systems, Cellular Telephone Systems.

GSM system: Architecture and features; GSM Services; Authentication; Incoming & outgoing call flow; Handover in GSM.

### UNIT-II

Digital Communication through Fading Multipath Channels: Fading types and their characteristics. Concept of Diversity: Branches and signal paths - Combining methods - Selective diversity combining - Pre-detection and post-detection combining - Switched combining - Maximal ratio combining - Equal gain combining.

Different Types of Channels: Control & Traffic channels.

BTS Hardware: Introduction of BTS 3900 series; Baseband Unit (BBU); Radio Frequency Unit (RFU); Description of Cards; Log into BTS 3900.

### UNIT-III

Multiple Access Techniques for Wireless Communications: Introduction, Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), Spread Spectrum Multiple Access, CDMA (Code Division Multiple Access), Space Division Multiple Access.

WCDMA: Features and architecture, handoff and its types

### UNIT-IV

Wireless Systems & Standards: GPRS/EDGE specification features and architecture.

3G Systems: Application of 3G & UMTS & CDMA 2000 standards, specifications and architecture of UMTS, Forward CDMA Channel, Reverse CDMA Channel.

BSC Hardware: Introduction to 6900 series; MPR & EPR; Description of Cards; Log into BSC6900.

Future Trends: Bluetooth technology, 4G mobile techniques, Wi-Fi technology, advanced systems, ZigBee.

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>

1	Wireless Communications	T.S. Rappaport,	Principles Edition, and Practice, 2 <sup>nd</sup> Pearson Education Asia, 2010.
2	Mobile Cellular Telecommunications	William CYLee	2ndEdition, MGH.
3	Mobile and Personal Communication systems and services	Raj Pandya	Prentice Hall of India.
4	Wireless and Digital Communications	Dr.Kamilo Feher	TMH

<b>Course Code</b>	<b>CSE326</b>
<b>Course Title</b>	<b>Block Chain</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	This course has no prerequisite other than knowledge of probability and statistics, and programming skills.
<b>Course Objectives</b>	The objective of this course is to teach students the architecture of blockchain. By taking this course, the students are expected to understand the basic algorithms, and be able to apply these techniques to financial service, supply chain.
<b>Course Outcome (CO)</b>	The learner will be able to- 1. Understand the basic architecture of blockchain. 2. Understand theory of bitcoin. 3. Describe components of blockchain. 4. Explain applications of blockchain in financial service, supply chain.

## SYLLABUS

### **Unit-I**

Introduction to Block chain-I :Basics ,History, Architecture, Conceptualization ,Bitcoin basics.

### **Unit-II**

Consensus in Bitcoin – I (The Basics, PoW and Beyond, The Miners), Permissioned Blockchain (Basics, Consensus)

### **Unit-III**

Blockchain for Enterprise – Overview, Blockchain Components and Concepts, Hyperledger Fabric Transaction Flow, Hyperledger Fabric Details. Fabric – Membership and Identity Management

### **Unit-IV**

Blockchain Use Cases. Blockchain in Financial Service (Payments and Secure Trading, Compliance and Mortgage, Financial Trade). Blockchain in Supply Chain  
Blockchain in Other Industries. Blockchain in Government (Advantages, Use Cases, Digital Identity)

<b>Course Code</b>	<b>CSE376</b>
<b>Course Title</b>	<b>ADVANCE DATABASE MANAGEMENT SYSTEM</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of Database and relational database management system
<b>Course Objectives</b>	This course is intended to provide an understanding of the current theory and practice of database management systems, a solid technical overview of database management systems. In addition to technical concerns, more general issues are emphasized. These include data independence, integrity, security, recovery, performance, database design principles and database administration.
<b>Course Outcome (CO)</b>	The learner will be able to- CO1: Explain the features of database management systems and Relational database. CO2: Analyze the existing design of a database schema using ER diagrams and apply concepts of normalization to design an optimal database. CO3: Identify the need for Concurrent transactions and locking and explain their types, advantages and disadvantages CO4: Formulate query, using SQL, solutions to a broad range of queries and data update problems. CO5: Explain Spatial and Multimedia databases

## Syllabus

### Unit-1

Introduction: Introduction to DBMS, RDBMS, Types of DBMS and their advantages and disadvantages, E-R Diagram, Keys, Normalization, Query optimization

Transaction management and Concurrency control: Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Two-phase locking, Multiple Granularity, Timestamp based Protocol. Time stamping methods, optimistic methods, database recovery management

### Unit-2

Distributed Databases: Basic concepts, structure, trade-offs Methods of data distribution – fragmentation, replication, design & advance concepts of DDBMS like Two-phase commit protocol, distributed transaction, distributed concurrency control, distributed deadlock handling.

Introduction to object-oriented databases: Object Oriented Data model, Object Oriented Database Management System, Object Query Language, Object Oriented Relational Database Management System and its concepts.

### Unit-3

Data warehousing Concepts: Architecture, Data-flows, Tools & Technologies, Data Marts, Data Mining and Online Analytical Processing.

### Unit-4

Emerging Database Technologies: Spatial & Multimedia databases, Mobile Computing & Mobile Databases, OLAP, data mining, data warehouse, multimedia database, geographical database, spatial database.

# **7<sup>TH</sup> SEMESTER**

<b>Course Code</b>	<b>CAI401</b>
<b>Course Title</b>	<b>Data Visualization</b>
<b>Type of Course</b>	PC
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of computer system
<b>Course Objective (CO)</b>	The students will be able to represent any type of dataset in visual form. They will also be able to draw insights from the data. They will also learn about different python visualization libraries.
<b>Course Outcome (CO)</b>	The students will able to: CO1: Apply data visualizations in order to derive more meaning out of data. CO2: Understand python visualization libraries. CO3: Apply data visualization on different types of data. CO4: Perceive hidden meanings from data using data visualization.

## SYLLABUS

### UNIT I

The Computer and the Human, Overview of Visualization, 2-D Graphics, SVG example, 2-D Drawing, 3-D Graphics, Photorealism, Non-Photorealism, the human retina: Perceiving Two Dimensions, Perceiving, Perspective

### UNIT II

Visualization tools, Line plots, area plots, histogram, bar charts, pie charts, scatter plots, bubble plots, waffle charts, word clouds

### UNIT III

Visualization of numerical data, Introduction, Data, Mapping, Charts, Glyphs, parallel coordinates, Parallel coordinates, Stacked graphs, Tufte's Design Rules, Using Color

### UNIT IV

Visualization of non-numerical data, Graphs and Networks, Embedding Planar Graphs, Graph Visualization, Tree Maps, Principal Component Analysis, Multidimensional Scaling, Python visualization libraries, matplotlib, pandas, seaborn, ggplot, plotly

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Taming Python by Programming	Jeeva Jose	Khanna Book Publishing House
2.	Data Visualization with Python and JavaScript: Scrape, Clean, Explore & Transform Your Data	Kyran Dale	O'Reilly, 2016

<b>Course Code</b>	<b>CSE403</b>
<b>Course Title</b>	<b>Cyber security</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of computer system
<b>Course Objective (CO)</b>	The main aim of this course is to provide knowledge about how to secure our data on the Internet.
<b>Course Outcome (CO)</b>	The students will be able to: <ol style="list-style-type: none"> <li>1. Implement cyber security best practices and risk management</li> <li>2. Integrate network monitoring and present real-time solutions</li> <li>3. Impact cyber security risk in an ethical, social, and professional manner.</li> <li>4. Learning basics of cyber laws and cyber forensic</li> </ol>

### SYLLABUS

#### UNIT I:

**Introduction to Cyber Security:** Overview of Cyber Security, Cyber Threats: - Cyber Warfare-Cyber Crime- Cyber Terrorism-Cyber Espionage

**Cyber Security Vulnerabilities and Cyber Security Safeguards:** Cyber Security Vulnerabilities-Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards- Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.

#### UNIT II:

**Securing Web Application, Services and Servers:** Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges.

**Intrusion Detection and Prevention:** Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, Systems, System Integrity Validation.

#### UNIT III:

**Cryptography and Network Security:** Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security Security Protocols: - security at the Application Layer- PGP and S/MIME, Security at Transport Layer- SSL and TLS, Security at Network Layer-IPSec.

#### UNIT IV:

**Cyberspace and the Law:** Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013.

RECOMMENDED BOOKS			
Sr. no.	Name	AUTHOR(S)	PUBLISHER
1.	Cyber security and Cyberwar: What Everyone Needs to Know®	Allan Friedman and P. W. Singer	Oxford University Press
2.	Cyber security for Beginners	Raef Meeuwisse	Cyber Simplicity Limited
3.	Cyber security Essentials	Charles J. Brooks, Christopher Grow, Donald Short, and Philip Craig	Sybex

<b>Course Code</b>	<b>CAI403</b>
<b>Course Title</b>	Big Data Analytics and its Applications
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	NIL
<b>Course Objective (CO)</b>	Study the requirements of non-traditional large-scale data applications
<b>Course Outcome (CO)</b>	The Learner will be able to : CO1: Understand concepts of Big Data. CO2: Analyze data used in Hadoop Architecture. CO3: Perform Text Analytics and understand JSON, NOSQL.

## **SYLLABUS**

### **UNIT-I**

DBMS Overview Introduction to big data, Handling and Processing Big Data, Methodological Challenges and Problems, Benefits and challenges of big data, Examples.

### **UNIT-II**

Analysis of data at Rest- Hadoop analytics: Limitations of existing distributing systems, Hadoop Approach, Hadoop Architecture, Distributed file system: HDFS and GPFS, Internals of Hadoop MR engine, Need for High level language- JAQL and PIG.

### **UNIT-III**

Introduction to Text Analytics: Using Regular expressions, Using AQL, Sentiment Analysis No SQL: JSON store, MongoDB, RDF, HBASE

### **UNIT-IV**

Analytics: Clustering, Classification, Segmentation, Linear regression, ML Search: Indexing and Indexing Techniques, Create inverted index using JAQL, Lab using Data Explorer Bundling Hadoop job: Application, Use BI tooling to create application, Publish applications. Analysis of data in motion – Real time analytics

Introduction to streams computing, Challenges/limitations of conventional Systems, Solving a real time analytics problem using conventional system, Challenges to be solved - scalability, thread pooling, etc., Understanding the challenges in handling streaming data from the real world and how to address those using stream computing, Benefits of stream computing in Big Data world, Realtime Analytics Platform(RTAP).

<b>Course Code</b>	MDC007
<b>Course Title</b>	MANAGING INNOVATION AND ENTREPRENEURSHIP
<b>Type of Course</b>	MDC
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	NIL
<b>Course Objectives</b>	Foundations of Entrepreneurship Concept, innovation management- definition and process of innovation management methods of man agent innovation Women entrepreneurs& Entrepreneurship Development
<b>Course Outcomes (CO)</b>	

## SYLLABUS

### UNIT-I

Foundations of Entrepreneurship Concept, Need, Definition& role of Entrepreneurship, Definition, characteristics& scope of Entrepreneur, Innovation, Invention, Creativity, Opportunities . Concepts of Entrepreneur, Manager, Intrapreneur / Corporate Entrepreneur comparative study, Roles& Responsibilities. Role of entrepreneur in Indian economy,

### UNIT-II

Women entrepreneurs & Entrepreneurship Development Meaning, role, problems& reasons for less women entrepreneurs. Various institutes & Govt schemes to help & uplift women entrepreneurs. Case studies for successful women entrepreneurs. Concept, need & role of Entrepreneurship Development

### UNIT-III

Small& Medium Enterprises: Small & Medium Industry: Meaning and importance Definition of SME –role & importance in India Economy, Steps for Starting Small Industry: Decisions to become entrepreneur -Steps to be taken-Search for a business idea, source of ideas, idea processing, selection idea, input requirements

### UNIT-IV

Innovation management- definition and process of innovation management methods of management innovation. Entrepreneurship as a career, Sustaining Competitiveness Maintaining competitive advantage

<b>RECOMMENDED BOOKS</b>			
<b>S. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Renu arora, S.K sood	Fundamentals of Entrepreneurship	Kalyani Publishers
2	Richard Branson	Entrepreneurship and Business	Pearson

<b>Course Code</b>	CSE405
<b>Course Title</b>	<b>Theory of Automata and Computation</b>
<b>Type of Course</b>	PC
<b>L T P</b>	4 0 0
<b>Credits</b>	4
<b>Course Prerequisites</b>	Basic knowledge of Discrete mathematics and System programming,
<b>Course Objectives</b>	<ul style="list-style-type: none"> <li>• Develop a formal notation for strings, languages and machines.</li> <li>• Design finite automata to accept a set of strings of a language.</li> <li>• Prove that a given language is regular and apply the closure properties of languages.</li> <li>• Distinguish between computability and non-computability and decidability and undecidability.</li> </ul>
<b>Course Outcome (CO)</b>	<p><b>The student will be able to-</b></p> <ul style="list-style-type: none"> <li>• Write a formal notation for strings, languages and machines.</li> <li>• Design finite automata to accept a set of strings of a language.</li> <li>• For a given language determine whether the given language is regular or not.</li> <li>• Distinguish between computability and non-computability and Decidability and undecidability.</li> </ul>

## SYLLABUS

### UNIT-I

Basic Theory of Automata: Sets, Relation, Functions, Alphabet, String, Languages Finite Automata: Formal Languages, Deterministic and Non-Deterministic Finite Automata, Finite Automata with  $\epsilon$ - moves, Equivalence of NFA and DFA, Minimization of finite automata, Two- way finite automata, Moore and Mealy machines, Applications of finite automata b) Regular Expression: Definition, Algebraic Laws, Conversion of R.E to F.A, F.A to R.E , Applications, Regular grammar for F.A.

### UNIT-II

Regular Sets and Context Free Grammars: Properties of regular sets, Context-Free Grammars – Derivation trees, Chomsky Normal Forms and Greibach Normal Forms, Ambiguous and unambiguous grammars. Pushdown Automata and Parsing Algorithms: Pushdown Automata and Context-Free Languages; Top-down parsing and Bottom-up parsing, Properties of CFL, Applications of Pumping Lemma, Closure properties of CFL and decision algorithms, Chomsky hierarchy.

### UNIT-III

Turing Machines: Turing machines (TM) – computable languages and functions – Turing Machine constructions – Storage in finite control.

Variations of TMs: Variations of TMs – Recursive and Recursive enumerable languages, Recursive Function, Partial and Total Recursive Function, Primitive Recursive Function.

### UNIT-IV

Introduction to Computational Complexity: Time and Space complexity of TMs –Complexity classes – Introduction to NP-Hardness and NP-Completeness, PCP Problem, Concept of decidability & undecidability. Undecidability: Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages.

<b>Sr. no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
<b>1</b>	Introduction to Automata Theory, Languages and Computation	John E. Hopcroft and Jeffrey D. Ullman	Narosa Publishers
<b>2</b>	Theory of Computer Science (Automata, Languages & Computation)	K.L.P. Mishra & N.C. handshekaran	PHI
<b>3</b>	Elements of the Theory of Computation	Harry R. Lewis and Christos H. Papadimitriou	Pearson Education Asia
<b>4</b>	Automata and Computability	Dexter C. Kozen	Undergraduate Texts in Computer Science, Springer

<b>Course Code</b>	<b>CSE481</b>
<b>Course Title</b>	<b>Major Project</b>
<b>Type of Course</b>	PROJ
<b>L T P</b>	0 0 4
<b>Credits</b>	2
<b>Course Prerequisites</b>	Nil
<b>Course Objectives</b>	The objective of Major Project is to enable the student to work on a project, either fully theoretical/practical or involving both theoretical and practical work, under the guidance of a Supervisor from the Department alone or jointly with a Supervisor drawn from R&D laboratory/Industry. This is expected to provide a good training for the student(s) in R&D work and technical leadership.
<b>Course Outcomes (CO)</b>	The learner will be able to 1. Understand the requirements of the project 2. Prepare Report 3. Present Findings before the department

The assignment to normally include:

1. In depth study of the topic assigned in the light of the study done.
2. Review and finalization of the Approach to the Problem relating to the assigned topic preferably in the area in which six weeks industrial / institutional training was taken after 6<sup>th</sup> semester.
3. Preparing an Action Plan for conducting the investigation, including teamwork.
4. Detailed Analysis/Modelling/Simulation/Design/Problem Solving/Experiment as needed.
5. Final development of product/process, testing, results, conclusions and future directions.
6. Preparing a paper for Conference presentation/Publication in Journals, if possible.
7. Preparing a project report with running code in the standard format for being evaluated by the Department.
8. Final Seminar Presentation before the Departmental Committee.

<b>Course Code</b>	<b>CSE485</b>
<b>Course Title</b>	<b>Four Weeks Industrial Training Evaluation (Undertaken after 6<sup>th</sup> semester)</b>
<b>Type of Course</b>	SEC III
<b>L T P</b>	
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basics of programming and software development
<b>Course Objectives</b>	To enhance programming skills of a learner, so that the learner finds solutions to problems. He also gets industrial experience of software development
<b>Course Outcomes- (CO)</b>	The learner will be able to- 1. implement software using proper software life cycle models 2. works with the latest IT tools 3. Develop team leadership

The six weeks industrial training will give exposure to the practical aspects of the discipline, in the real-time working scenario. In addition, the student may also work on a specified task or project which may be assigned to him/her, by the industry person. The student will maintain the daily diary which will have signature of an industry expert, assigned to him/ her. This daily diary will be produced by the student during practical examinations, as and when scheduled by the institute. The department will evaluate student performance based on his/her project report, running software code, CD containing code and daily diary

<b>Course Code</b>	<b>CSE451</b>
<b>Course Title</b>	<b>Cryptography</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic Knowledge of complexity theory, algorithms, game theory, machine learning
<b>Course Objectives</b>	This course work provides the thorough understanding of the network security and various cryptography techniques to obtain security on network and a computer.
<b>Course outcome(CO)</b>	The learner will be able to- 1. Understand concepts related to security attacks, encryption, decryption techniques, and substitution and transposition techniques. 2. Describe principles of public key cryptography, RSA algorithm. 3. Explain authentication requirements and use of hash function

## SYLLABUS

### UNIT-I

**Introduction:** Attacks, Services and Mechanisms, Security attacks, security services, model for internet network security. Conventional Encryption: Conventional Encryption Model, steganography, Classical Encryption Techniques: Substitution Techniques, Transposition Techniques.

### UNIT-II

**Modern Encryption Techniques:** Simplified Data Encryption Standard, Block Cipher Principles.

*The Data Encryption Standard, Strength of DES.*

**Encryption Algorithms:** Triple DES, International Data Encryption Algorithm, Blowfish.

### UNIT-III

**Confidentiality using Conventional Encryption:** Placement of Encryption Function, Traffic Confidentiality, Key distribution, Random Number Generation.

**Public- Key Cryptography:** Principles of Public- Key Cryptosystems, RSA algorithm, Key Management, Diffie-Hellman Key.

### UNIT-IV

**Message Authentication and Hash Functions:** Authentication Requirements, Authentication Functions, Message Authentication Codes, Hash Functions, Security of hash Functions and MACs, Digital Signatures, Authentication Protocols, SHA-1, RC-4, RC-5.

<b>Course Code</b>	<b>CAI405</b>
<b>Course Title</b>	Human-Computer Interaction
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of Computer Graphics & Image Processing.
<b>Course Objectives</b>	The main objective of the subject is to impart knowledge about animation execution, workflow & post-production.
<b>Course Outcomes (CO)</b>	<p><b>The learner will be able to:</b></p> <ol style="list-style-type: none"> <li>1. Understand fundamentals of animation.</li> <li>2. Get knowledge of 3D modeling tools.</li> <li>3. Compare between Polygon Modeling and NURBS modeling.</li> </ol>

### SYLLABUS

#### UNIT 1: Introduction to HCI

This unit covers the basics of Human-Computer Interaction (HCI), including what HCI is, its importance, and its scope. It explores the evolution of user interfaces over time and discusses key human factors involved in HCI such as perception, memory, and learning. The goals of HCI, including usability, user experience (UX), and accessibility, are introduced. Finally, different types of interfaces are covered, including command-line, graphical, web, voice, and touch interfaces.

#### UNIT 2: Human-Centered Design

This unit focuses on design principles essential to creating effective interfaces, such as consistency, feedback, affordance, and visibility. It introduces the user-centered design process, including stages like analysis, design, prototyping, and evaluation. Key techniques like task analysis and creating user personas are explained. Students also learn about wireframing and low-fidelity prototyping methods, along with an introduction to usability testing to evaluate design effectiveness.

#### UNIT 3: Interaction Models & Design

This unit covers different models of interaction including Norman's Model and various interaction frameworks. It discusses input and output devices and technologies that facilitate interaction. Different types of user interactions are explored, such as direct manipulation, menu selection, and form fill-in. Dialogue design and navigation principles are taught, along with strategies for making interfaces accessible and inclusive for diverse users.

#### UNIT 4: Evaluation and Emerging Trends

This unit introduces usability evaluation methods such as heuristic evaluation, cognitive walkthroughs, and surveys to assess interface quality. It covers A/B testing and techniques for analyzing user feedback. The unit also explores emerging applications of HCI in mobile, wearable, and VR/AR devices. Ethical issues like privacy and accessibility are discussed, along with future directions in HCI, including AI integration and affective computing.

RECOMMENDED BOOKS			
Sr.no.	Name	Author(s)	Publisher
1	3D Animation for the Raw Beginner Using Maya	Roger King	Chapman and Hall

2	Editing Digital Video - The Complete Creative and Technical Guide	Robert Goodman	McGraw-Hill
3	Maya Documentation	<a href="https://knowledge.autodesk.com">https://knowledge.autodesk.com</a>	Autodesk

<b>Course Code</b>	<b>CAI407</b>
<b>Course Title</b>	AI agents and Autonomous System
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of AI
<b>Course Objectives</b>	<ul style="list-style-type: none"> <li>• To understand the fundamental concepts of Artificial Intelligence, intelligent agents, and autonomous systems.</li> <li>• To study various agent architectures, search techniques, and knowledge representation methods.</li> <li>• To explore machine learning approaches and multi-agent system coordination techniques.</li> <li>• To analyze real-world applications, ethical issues, and emerging trends in autonomous technologies</li> </ul>
<b>Course Outcomes (CO)</b>	<ul style="list-style-type: none"> <li>• Understand the concepts and working principles of intelligent agents and autonomous systems.</li> <li>• Apply search and knowledge representation techniques for AI-based problem solving.</li> <li>• Analyze machine learning and multi-agent approaches in autonomous environments.</li> <li>• Evaluate real-world autonomous applications along with ethical and societal considerations.</li> </ul>

## **SYLLABUS**

### **Unit I: Introduction to AI and Intelligent Agents**

- Basics of Artificial Intelligence
- Evolution of AI
- Characteristics of Intelligent Agents
- Types of Intelligent Agents
- Agent Architecture
- Agent Environment and Rationality
- Perception and Action Mechanisms
- Applications of AI Agents

### **Unit II: Agent Design and Knowledge Representation**

- Problem Solving in AI
- State Space Representation
- Uninformed Search Techniques
- Informed and Heuristic Search
- Optimization Techniques
- Knowledge Representation Methods
- Rule-Based Systems and Logical Reasoning
- Planning and Decision Making

### **Unit III: Machine Learning and Multi-Agent Systems**

- Introduction to Machine Learning
- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Neural Networks and Decision Trees
- Learning in Autonomous Agents
- Multi-Agent Systems
- Cooperation and Communication among Agents
- Applications of Multi-Agent Systems

### **Unit IV: Autonomous Systems and Emerging Trends**

- Autonomous Systems and Robotics
- Drones and Self-Driving Vehicles
- Sensors and Perception Systems
- Localization and Navigation
- Path Planning Techniques
- Human-Agent Interaction
- Ethics and Responsible AI
- Generative AI and Conversational Agents
- Edge AI and Future Trends

<b>Sr. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	<i>Artificial Intelligence: A Modern Approach</i>	Stuart Russell, Peter Norvig	Pearson
2	<i>Artificial Intelligence</i>	Elaine Rich, Kevin Knight	McGraw Hill
3	<i>Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations</i>	Yoav Shoham, Kevin Leyton-Brown	Cambridge University Press

<b>Course Code</b>	<b>CSE455</b>
<b>Course Title</b>	<b>Natural Language Processing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Objectives</b>	The objective of this course is to provide knowledge of the fundamentals of speech and text processing
<b>Course Outcomes(CO)</b>	The learner will be able to- 1. Understand basic concepts of Natural language processing 2. Explain Machine translation and speech recognition

SYLLABUS

**UNIT-I**

**Introduction:** Natural Language Processing (NLP), Challenges of NLP, NLP applications, Processing of Indian Languages.

**UNIT-II**

**Structures:** Theories of Parsing, Parsing Algorithms; Robust and Scalable Parsing on Noisy Texts in Web documents; Hybrid of Rule Based and Probabilistic Parsing; Scope Ambiguity and Attachment Ambiguity resolution.

**UNIT-III**

Understanding Part of Speech or Text Processing: Tokenization, Sentence segmentation or Splitting, Normalization

**UNIT-IV**

**Words and Word Forms:** Morphology fundamentals; Morphological Diversity of Indian Languages; Morphology Paradigms; Finite State Machine Based Morphology; Automatic Morphology Learning; Shallow Parsing; Named Entities; Maximum Entropy Models; Random Fields, Scope Ambiguity and Attachment Ambiguity resolution

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Natural Language Understanding	Allen, James	Second Edition, Benjamin/Cumming
2	Statistical Language Learning	Charniack, Eugene	MIT Press
3	Speech and Language Processing	Jurafsky, Dan and Martin, James	Second Edition, Prentice Hall
4	Foundations of Statistical Natural Language Processing	Manning, Christopher and Heinrich, Schutze	MIT Press.

<b>Course Code</b>	<b>CSE477</b>
<b>Course Title</b>	<b>Data mining in Business Intelligence</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of Computer Graphics & Image Processing.
<b>Course Objectives</b>	Students will be enabled to understand and implement classical models and algorithms in data mining.
<b>Course Outcomes (CO)</b>	The learner will be able to- 1. Understand Data mining and its scope. 2. Understand various data mining techniques 3. Describe supervised and unsupervised clustering techniques 4. Illustrate applications of data mining using real life examples

## SYLLABUS

### UNIT-I

Introduction to Data Mining: Introduction: Scope of Data Mining: What is Data Mining; How does Data Mining Works, Predictive Modeling: Data Mining and Data Warehousing: Architecture for Data Mining: Profitable Applications: Data Mining Tools: Data Preprocessing: Introduction, Data Preprocessing Overview, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

### UNIT-II

**Data Mining Techniques-** An Overview: Introduction, Data Mining, Data Mining Versus Database Management System, Data Mining Techniques- Association rules, Classification, Regression, Clustering, Neural networks.

### UNIT-III

**Clustering:** Introduction, Clustering, Cluster Analysis, Clustering Methods- K means, Hierarchical clustering, Agglomerative clustering, Divisive clustering, clustering and segmentation software, evaluating clusters.

### UNIT-IV

Applications of data mining: Introduction, Business applications using data mining- Risk Management and targeted marketing, Customer profiles and feature construction, Medical applications, Scientific applications using data mining

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Introduction to Data Mining	Pang-Ning Tan, Michael Steinbach, Vipin Kumar	Pearson Education India
2	Data Mining	Pieter Adrians, Dolf zantinge	Pearson Education India
3	Database Management Systems	R. Ramakrishnan, J.	McGraw Hill

*8<sup>th</sup>*

*Semester*

<b>Course Code</b>	<b>CSE466</b>
<b>Course Title</b>	<b>Six Months Industrial Training</b>
<b>Type of Course</b>	Training
<b>L T P</b>	
<b>Credits</b>	20
<b>Course Prerequisites</b>	Basics of programming and software development
<b>Course Objectives</b>	To enhance programming skills of a learner, so that the learner finds solutions to problems. He also gets industrial experience of software development
<b>Course Outcomes- (CO)</b>	The learner will be able to- 1. implement software using proper software life cycle models 2. works with latest IT tools 3. Develop team leadership

The six months industrial training will give exposure to the practical aspects of the discipline, in real time working scenario. In addition, the student may also work on a specified task or project which may be assigned to him/her, by the industry person. The student will maintain the daily diary which will have signature of industry expert, assigned to him/her. This daily diary will be produced by the student during mid semester viva voce and internal and external end semester practical examinations, as and when scheduled by the institute. The department will get the marks assigned by the industry expert, against student performance or evaluation. The outcome of the internship should be presented in the form of a project report, running software code, CD containing code and project report, daily diary.

# *Open Electives*

<b>Course Code</b>	<b>CSE391</b>
<b>Course Title</b>	<b>Basics of AI</b>
<b>Type of Course</b>	PC
<b>L: T:P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Nil
<b>Course Objective(s)</b>	To get introduced to the basic knowledge representation, problem solving, and learning methods of Artificial Intelligence. To solve problems in Artificial Intelligence using Python. To familiarize yourself with knowledge processing in expert systems.
<b>Course Outcome (CO)</b>	CO1 Understand the informed and uninformed problem types and apply search strategies to solve them. CO2 Apply difficult real-life problems in a state space representation to solve them using AI techniques like searching and game playing. CO3 Design and evaluate intelligent expert models for perception and prediction from an intelligent environment. CO4 Understand Reinforcement Learning and Ethical Implications.

## SYLLABUS

### Unit 1: Foundations of Artificial Intelligence

Introduction to AI: Definition and brief history of Artificial Intelligence, AI applications across different domains, Overview of AI subfields: symbolic AI, machine learning, and deep learning.

Search Algorithms and Knowledge Representation: Problem-solving agents, Uninformed search algorithms: Breadth-first search, Depth-first search, Informed search algorithms: A\* search, Heuristic search, Propositional logic and first-order logic (Syntax and semantics, Resolution and inference), Semantic networks and frames (Representation and inference).

### Unit 2: Machine Learning Basics

Introduction to Machine Learning: Basics of machine learning, Supervised, unsupervised, and reinforcement learning, Evaluation metrics in machine learning.

Classical Machine Learning Algorithms: Linear regression (Simple and multiple linear regression, Gradient descent optimization), Logistic regression (Binary and multinomial logistic regression, Sigmoid function and probability estimation), Decision trees and ensemble methods (Decision tree construction, Bagging, boosting, and random forests).

### Unit 3: Deep Learning and Neural Networks

Neural Networks: Introduction to artificial neural networks (Perceptron's and activation functions, Feed forward and back propagation), multi-layer perceptron's (Hidden layers and network architecture, Activation functions), Training neural networks (Gradient descent and stochastic gradient descent, Regularization techniques).

### Unit 4: Applications and Ethical Considerations

Natural Language Processing: Basics of natural language processing (NLP), Text preprocessing and tokenization, NLP applications (Sentiment analysis, Named entity recognition, Part-of-speech tagging, Word embeddings and semantic similarity).

Reinforcement Learning and Ethical Implications: Introduction to reinforcement learning (Markov decision processes, Policy iteration and value iteration), Q-learning and deep Q-networks (DQN) (Experience replay and target networks, Deep reinforcement learning algorithms), Privacy concerns and data ethics (Data anonymization and de-identification techniques, Case studies of AI regulation worldwide).

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Artificial Intelligence	E. Rich	McGrawHill
2	Introduction to Artificial Intelligence	E. Charniak and D. McDermott	Addison Wesley

<b>Course Code</b>	<b>CSE393</b>
<b>Course Title</b>	<b>Introduction to Cloud Computing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Distributed System, Operating Systems and Networking
<b>Course Objectives (CO)</b>	This Course work provides the complete understanding of Cloud system, its implementation techniques and its various applications in the field of computer Science.
<b>Course Outcome</b>	The learner will be able to- 1. Understand characteristics and types of cloud computing 2. Describe architecture of cloud computing 3. Explain applications of cloud 4. Demonstrate their knowledge of cloud computing to real world examples

### SYLLABUS

#### UNIT-I

Cloud Computing Basics, History of Cloud Computing, Importance, Characteristics of Cloud Computing, Benefits and Challenges to Cloud architecture.

#### UNIT-II

Types of Cloud: Public Cloud, Private Cloud, Hybrid and Community Cloud. Differences between public and private cloud, Status of Cloud Computing in India, Cloud Service Models, Role of virtualization in enabling the cloud; Differences between Grid computing and cloud computing, differences between grid computing and utility computing, Cloud Computing security concerns and proposed security model for future cloud computing.

#### UNIT-III

Cloud Computing- Logical architecture, Developing Holistic Cloud Computing Reference Models- Seven step model of migrating to cloud.

Virtualization types, Virtual Machine Life Cycle, Virtualization applications, Pitfalls of Virtualization, CPU Virtualization.

#### UNIT-IV

Case Study of Cloud Computing, Cloud Computing Risks. Cloud Tools, Cloud Applications, Future Trends, Mobile cloud, Jungle Computing, Big Data –Features and applications

<b>RECOMMENDED BOOKS</b>			
<b>Sr.no.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	Cloud Computing –A Practical Approach	Anthony Teletubby J.Velte and RobertE	TMH
2	Cloud Computing –Web based Applications	Michael Miller	Pearson Publishing

<b>Course Code</b>	<b>CSE491</b>
<b>Course Title</b>	<b>Introduction to Operating Systems</b>
<b>Type of Course</b>	PC
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Overview of Computer Architecture
<b>Course Objectives</b>	To learn the fundamentals of Operating Systems. To learn the mechanisms of OS to handle processes and threads and their communication To learn the mechanisms involved in memory management in contemporary OS To gain knowledge on distributed operating system concepts that include architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols To know the components and management aspects of concurrency management Learning to implement simple OS mechanisms
<b>Course Outcome (CO)</b>	<b>The learner will be able to-</b> <ul style="list-style-type: none"> <li>• Create processes and threads.</li> <li>• Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.</li> <li>• For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and improving the access time.</li> <li>• Design and implement file management system.</li> <li>• For a given I/O devices and OS (specify) develop the I/O management functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/ O controllers.</li> </ul>

## SYLLABUS

### UNIT-I

**Introduction:** Operating Systems functions, Types of operating systems, Multiprogramming systems, Batch systems, Time-sharing systems, Operating system operations, Special purpose operating systems, distributed systems, Different computing environments.

### UNIT-II

**Operating System Organization:** Processor and user modes, user operating system interface, Kernels, System calls and its types, System programs, Operating system structures, Virtual machines.

**Process Management:** Process states, Process Scheduling, Process hierarchy, Threads, Threading issues, multi-threading models, non-preemptive and pre-emptive scheduling algorithms, Concurrent processes, Critical section, Semaphores, methods for inter-process communication, Deadlocks.

### UNIT-III

**Memory Management:** Physical and virtual address space, Memory allocation strategies, Paging, Segmentation, Virtual memory and Demand paging, Page replacement algorithms.

**File and I/O Management:** Directory structure, File operations, Files system mounting, File allocation methods, Device management, Disk scheduling algorithms.

### UNIT-IV

**OS and Security:** Security breaches, types of attacks, attack prevention methods, security policy and access control, OS design considerations for security, access control lists and OS support, internet and general network security, Policy mechanism, Program, network and system threats, Authentication.

**RECOMMENDED BOOKS**

<b>Sr.no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	Operating System Concepts Essentials	9 <sup>th</sup> Edition by Avi Silberschatz, Peter Galvin, Greg Gagne	Wiley Asia Student Edition.
2	Operating Systems: Internals and Design Principles	5 <sup>th</sup> Edition, William Stallings	Prentice Hall of India
3	Operating System: A Design-oriented Approach	1 <sup>st</sup> Edition by Charles Crowley	Irwin Publishing
4	Operating Systems: A Modern Perspective	2 <sup>nd</sup> Edition by Gary J. Nutt	Addison-Wesley

<b>Course Code</b>	<b>CSE493</b>
<b>Course Title</b>	<b>Basics of Networking</b>
<b>Types of Course</b>	<b>OE</b>
<b>L T P</b>	<b>3: 0: 0</b>
<b>Credits</b>	<b>3</b>
<b>Course Prerequisites</b>	Nil
<b>Course Objectives</b>	It aims to introduce students to the fundamental techniques used in implementing secure network communications, and to give them an understanding of common threats and attacks.
<b>Course Outcomes (CO)</b>	The student will be able to- CO1: Understand basic concepts and security in network technology CO2: Explain IPv6 CO3: Explain classical encryption techniques CO4: Illustrate applications of Network Security

## Syllabus

### UNIT-I

**Introduction to Network Technology:** SLIP/PPP Dedicated lines, BOOTP, DHCP, Domain management (DNS), Transport Layer issues, TCP/IP, Gateway, Dial-up, Internet networking TCP/IP protocols, IP addressing.

### UNIT-II

**Basics of Network security-** Fundamentals of network security, Basics of IPv6, IPsec: overview of IPsec, IP and IPv6, Authentication header (AH), Encapsulating Security Payload (ESP).

**Security Trends** – Attacks and services, Classical crypto systems, Different types of ciphers, LFSR sequences, Basic Number theory, Congruences, Chinese Remainder theorem, Modular exponentiation, Fermat and Euler's theorem, Legendre and Jacobi symbols, Finite fields, continued fractions.

### UNIT-III

**Model of Network security-** Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Block Cipher and the Data Encryption Standard - Modes of operation, Triple DES, AES, RC4, RSA, Attacks, Primality test, Factoring.

**Discrete Logarithms** – Digital signatures, RSA, ElGamal, DSA, Unwanted traffic: denial of service attacks.

### UNIT-IV

**Authentication applications** – Kerberos, X.509, PKI, Electronic Mail security, PGP, S/MIME, IP security, Web Security, SSL, TLS, SET.

**System Security** – Intruders, Malicious software, viruses, Firewalls and filters, Security

<b>BOOKS RECOMMENDED</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Network Security and Ethical Hacking	Rajat Khare	Luniver Press
2.	Cryptography and Network Security	Atul Kahate	Tata Mc-Graw Hill
3.	Computer Networks	A.S Tanenbaum	Pearson

<b>Course Code</b>	<b>CSE495</b>
<b>Course Title</b>	<b>Introduction to Digital Marketing</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Nil
<b>Course Objective</b>	The main objective of this course is to provide learners with the knowledge of business advantages of digital marketing and its importance for marketing success; to develop a digital marketing plan; to make SWOT analysis; to define a target group; to get introduced to various digital channels, their advantages and ways of integration;
<b>Course Outcomes</b>	The learner will be able to- Identify the importance of digital marketing for marketing success, Manage customer relationships across all digital channels and build better customer relationships, Create a digital marketing plan, starting from the SWOT analysis and defining a target group, then identifying digital channels, their advantages and limitations. Perceiving ways of integration taking into consideration the available budget.

### SYLLABUS

#### UNIT I

Introduction: Marketing and its definition, Digital Marketing, How we do Marketing, Benefits of Digital marketing, Digital marketing platforms and Strategies, Defining Marketing Goals, Latest Digital marketing trends, introduction to traditional and new methods of marketing Requirement: Requirements for digital marketing, its uses.

#### UNIT II

Search Engine Optimization: Introduction to Search Engines, How the search engine works, Components of Search Engines. Keyword Research and Competition: Introduction to Keyword Research, Types of Keywords, Keyword Research Methodology, Business Analysis & Categorization, Google Keyword Planner, Market Research and Analysis, New Keyword Ideas, Competition Analysis, Finalizing the Keywords List.

#### UNIT III

On page Optimization: Introduction to On page ,What is Webmaster Tools, Selecting Target Location, On page Analysis Methodology, Fundamental On-page Factors , Website Speed , Domain name in SEO, URL Optimization , Title Tag Optimization , Meta Tags Optimization , Content Optimization , Sitemaps Generation, Using Robot.txt in Site URL , Redirecting Techniques , Canonical Links, Rich Snippets.

#### UNIT IV

Off page Optimization : What is Link Building , Types of Linking Methods , Do Follow Vs. No Follow Link building Guidelines , Linking Building Methodology , Links Analysis Tools , Directory Submissions, Local Business Directories , Social Bookmarking , Using Classifieds for Inbound traffic ,Question and Answers , Blogging & Commenting , Guest Blogging Local SEO: What is Local SEO, Importance of Local SEO , Submission to Google My Business , Completing the Profile , Local SEO Ranking Signals , Local SEO Negative Signals , Citations and Local Submissions

<b>Sr. No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
1	<i>Digital Marketing for Dummies</i>	Ryan Deiss, Russ Henneberry	Wiley
2	<i>Digital Marketing: Strategy, Implementation and Practice</i>	Dave Chaffey, Fiona Ellis-Chadwick	Pearson
3	<i>The New Rules of Marketing and PR</i>	David Meerman Scott	Wiley

<b>Course Code</b>	<b>CSE497</b>
<b>Course Title</b>	<b>Basics Concepts of IOT</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	NIL
<b>Course Objectives</b>	The Internet is evolving to connect people to physical things and physical things to other physical things all in real time. It's becoming the Internet of Things (IoT). The course enables students to understand the basics of Internet and protocols. It introduces some of the application areas where Internet of Things can be applied.
<b>Course Outcome(CO)</b>	At the end of the course the learner will be able to- <ol style="list-style-type: none"> <li>1. Understand and describe Functional blocks of IOT</li> <li>2. Explain MAC protocol and various routing protocols</li> <li>3. Describe data aggregation and data dissemination</li> <li>4. Evaluate and explain challenges in IoT design</li> <li>5. Demonstrate the ability to develop applications through IoT tools</li> </ol>

### SYLLABUS

#### UNIT-I

Introduction to IoT Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models and APIs

#### UNIT-II

IoT & M2M Machine to Machine, Difference between IoT and M2M, Software define Network, Network and Communication aspects Wireless medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment and Node discovery, Data aggregation and dissemination

#### UNIT-III

Challenges in IoT Design challenges, Development challenges, Security challenges, other challenges Domain specific applications of IoT Home automation, Industry applications, Surveillance applications, Other IoT applications

#### UNIT-IV

Developing IoTs Introduction to Python, Introduction to different IoT tools, developing applications through IoT tools, developing sensor-based application through embedded system platform, Implementing IoT concepts with python

#### BOOKS RECOMMENDED

<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1	The Internet of Things in the Cloud:A Middleware Perspective	Honbo Zhou	CRC Press,2012

<b>Course Code</b>	<b>CSE489</b>
<b>Course Title</b>	<b>ECOMMERCE</b>
<b>Type of Course</b>	PE
<b>L T P</b>	3 0 0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basics of Internet
<b>Course Objectives (CO)</b>	This course examines the evolution of enterprise resource planning(ERP) systems - from internally focused client/server systems to externally focused e-business.
<b>Course Outcomes</b>	<ul style="list-style-type: none"> <li>• Understand the fundamental concepts, models, and applications of E-Commerce systems.</li> <li>• Apply online business strategies and electronic payment technologies in digital commerce.</li> <li>• Analyze security, legal, and ethical issues related to E-Commerce platforms.</li> <li>• Evaluate the role of E-Commerce in modern business and digital marketing environments.</li> </ul>

### SYLLABUS

#### UNIT I

**Introduction and Concepts Networks and commercial transactions** - Internet and other novelties; Networks and electronic transactions today, Model for commercial transactions; Internet environment - internet advantage, world wide web and other internet sales venues; Online commerce solutions. Security Technologies: Why is internet insecure? A brief introduction to Cryptography; Public key solution. Digital payment systems; First virtual internet payment system; cyber cash model Operational process of Digicash, Ecash Trail; Using Ecash; Smart cards; Electronic Data Interchange: Its basics; EDI versus Internet and EDI over Internet.

#### UNIT II

**Introduction ERP An Overview, Enterprise-An Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering (BPR), Data Warehousing, Data Mining, On-line Analytical Processing (OLAP), Supply Chain Management, Management Information systems (MIS), Decision support system (DSS), Executive Information systems (EIS). ERP – A Manufacturing Perspective Materials Requirement Planning (MRP), Bill of Material (Bom), Distribution Requirements Planning (DRP), JIT & Kanban, CAD/CAM.**

#### UNIT III

**ERP Implementation** - ERP Implementation Lifecycle, Implementation Methodology, Not all Packages are Created Equal!, ERP Implementation-The Hidden Costs, Organizing the Implementation, Vendors, Consultants and Users, Contracts with Vendors, Consultants and Employees, Project Management and Monitoring, After ERP Implementation.

#### UNIT IV

**The Business Modules-** Business Modules in an ERP Package, Finance, Manufacturing (Production), Human Resources, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution

<b>RECOMMENDED BOOKS</b>			
<b>S.No.</b>	<b>Name</b>	<b>Author(s)</b>	<b>Publisher</b>
<b>1</b>	Enterprise Resource Planning	S. Sadagopan	Tata McGraw Hill 2000
<b>2</b>	E-Commerce: The Cutting Edge of Business	Bajaj, Kamlesh K. and Nag, Debjani	Tata McGraw-Hill Publishing Company
<b>3</b>	Enterprise Resource Planning	Alexis Leon	Tata McGraw Hill 2001
<b>4</b>	Electronic Commerce	Loshin, Pete and Murphy, Paul	Second edition, 1990, Jaico Publishing House, Mumbai

<b>Course Code</b>	<b>CSE499</b>
<b>Course Title</b>	<b>Introduction to Cyber security</b>
<b>Type of Course</b>	PC
<b>L T P</b>	3:0:0
<b>Credits</b>	3
<b>Course Prerequisites</b>	Basic knowledge of computer system
<b>Course Objective (CO)</b>	The main aim of this course is to provide knowledge about how to secure our data on the Internet.
<b>Course Outcome (CO)</b>	The students will be able to: <ul style="list-style-type: none"> <li>• Implement cyber security best practices and risk management</li> <li>• Integrate network monitoring and present real-time solutions</li> <li>• Impact cyber security risk in an ethical, social, and professional manner.</li> <li>• Learning basics of cyber laws and cyber forensic</li> </ul>

## SYLLABUS

### UNIT I:

**Introduction to Cyber Security:** Overview of Cyber Security, Cyber Threats: - Cyber Warfare-Cyber Crime- Cyber Terrorism-Cyber Espionage

**Cyber Security Vulnerabilities and Cyber Security Safeguards:** Cyber Security Vulnerabilities-Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards- Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.

### UNIT II:

**Securing Web Application, Services and Servers:** Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges.

**Intrusion Detection and Prevention:** Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, Systems, System Integrity Validation.

### UNIT III:

**Cryptography and Network Security:** Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security Security Protocols: - security at the Application Layer- PGP and S/MIME, Security at Transport Layer- SSL and TLS, Security at Network Layer-IPSec.

### UNIT IV:

**Cyberspace and the Law:** Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013.

<b>RECOMMENDED BOOKS</b>			
<b>Sr. no.</b>	<b>Name</b>	<b>AUTHOR(S)</b>	<b>PUBLISHER</b>
1.	Cyber security and Cyberwar: What Everyone Needs to Know®	Allan Friedman and P. W. Singer	Oxford University Press
2.	Cyber security for Beginners	Raef Meeuwisse	Cyber Simplicity Limited
3.	Cybe security Essentials	Charles J. Brooks, Christopher Grow, Donald Short, and Philip Craig	Sybex

# Entrepreneurship Mindset

## Semester 1

<b>Course Name</b>	Entrepreneurship Mindset
<b>Course Code</b>	To be filled by respective Universities
<b>Credits (L-T-P)</b>	02Credits (002)-
<b>Total Marks</b>	100

### Introduction

This course provides students with a holistic framework to develop entrepreneurial skills across the student's journey. The curriculum covers core principles of entrepreneurship, including identifying business opportunities, setting up operations, marketing, scaling, and financial management.

***Throughout the program, students will choose 01 track from 5 specialized tracks:***

- 1. E-commerce:** In this track, students will learn how to build, manage, and scale an online business. From setting up an e-commerce platform to managing digital marketing strategies, inventory, and customer experience, this specialization provides the tools needed to succeed in the rapidly growing digital marketplace.
- 2. Content Creation:** This track focuses on empowering students to create, curate, and market digital content across various platforms. Students will learn how to build a personal or brand identity, develop engaging content, and monetize their platforms, whether through social media, video production, blogging, or influencer marketing.
- 3. Professional Services:** In this track, students will learn how to capitalize on freelance and contract-based opportunities. They will explore strategies for building a sustainable service-based business, manage client relationships, and maximize the flexibility that comes with this track.
- 4. Retail Business:** This track introduces students to the fundamentals of starting and managing small-scale retail ventures. Learners will discover how to identify in-demand products, source suppliers, set up shop (physical or home-based), and attract local customers.

Emphasis will be placed on practical steps like pricing, promotion, customer service, and building repeat sales to establish a steady income stream.

5. **Miscellaneous:** This track gives students the freedom to explore unique and diverse business ideas that don't fit into one category. Learners can identify opportunities around them, test simple solutions, and create small ventures based on local needs or personal interests. The focus is on creativity, problem-solving, and learning how to turn everyday ideas into earning opportunities.

The focus is on practical application, with students engaging in real-world projects that culminate in the creation and scaling of a business.

## Learning Objectives

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

- Understand and apply entrepreneurial principles to real-world business situations.
- Develop and implement business strategies across different industries.
- Use digital tools, including AI, to enhance and automate business operations.
- Build sustainable business models, manage financials, and scale operations.

## Learning Outcomes

After studying this course, students will be able to:

- Launch and manage a business within their chosen track.
- Identify profitable opportunities and develop innovative solutions.
- Implement marketing and sales strategies using both digital and traditional methods.
- Use financial metrics to track performance and make informed business decisions.
- Scale a business using operational systems and automation tools.

The aim of this course is to ensure that, by the end, learners acquire essential entrepreneurial competencies such as **strategic thinking, practical application, founders & growth mindset, operational skills,** and **foundational financial literacy.**

<b>Content Creation</b>		
<b>Week/ Milestone</b>	<b>Milestone</b>	<b>Description</b>
<b>1</b>	<b>Start your journey as a content creator</b>	Learn how to reach thousands of followers online by creating content
<b>2</b>	<b>Decide your content topic</b>	Explore different topics and finalize one topic on which you will create content
<b>3</b>	<b>Start your own content channel</b>	Launch your official channel on YouTube or Instagram and start building your audience
<b>4</b>	<b>Plan your first week content calendar</b>	Learn how to create a simple weekly content calendar that would keep your audience engaged and make them follow you
<b>5</b>	<b>Plan and record your first video</b>	Learn how to create a clear and catchy script for your first video and shoot it using your phone
<b>6</b>	<b>Edit and launch your first video</b>	Learn how to edit your video with trendy music and cool effects and launch your channel with first video
<b>7</b>	<b>Reach first 100 followers</b>	Use WhatsApp and personal Instagram account to tell people about your channel and reach first 100 followers/subscribers
<b>8</b>	<b>Master the skill of engaging people with your content</b>	Learn how to connect with your audience so they stay engaged and feel involved
<b>9</b>	<b>Follow latest trends and famous influencers to grow your channel</b>	Learn how to use trending topics and tag well-known channels or creators to boost your reach and attract more followers
<b>10</b>	<b>Learn how you can make money from your content</b>	Discover how to do brand deals and promote brands or products your followers truly care about
<b>11</b>	<b>Make your first partnership</b>	Reach out to 10 brands for partnership and turn one into your first paid deal
<b>12</b>	<b>Plan your business growth</b>	Create a 1 month plan on what to post on a weekly basis to grow your followers and earn money

<b>E-commerce</b>		
<b>Week/ Milestone</b>	<b>Milestone</b>	<b>Description</b>
<b>1</b>	<b>Learn how to start selling online</b>	Learn what an E-commerce business is and how you can start selling online in less than 30 days
<b>2</b>	<b>Choose your product</b>	Explore different products and finalize what you want to sell online
<b>3</b>	<b>Find a supplier who will deliver the product for you</b>	Identify suppliers who can send your product directly to the customer and help you launch your E-commerce business
<b>4</b>	<b>Create your E-commerce store</b>	Make a simple online store using Instamojo and learn how to upload product images and description on the store
<b>5</b>	<b>Activate payments on your store</b>	Learn how to link your bank account to a payment gateway and integrate that with your Instamojo store to start receiving payments
<b>6</b>	<b>Launch your online store</b>	Connect your domain name to your Instamojo account, create your launch poster, and officially launch your E-commerce store
<b>7</b>	<b>Get first 100 people visit your online store</b>	Learn how to write simple and catchy messages to promote your store and send it to 100 people
<b>8</b>	<b>Make your first sale</b>	Convert one paying customer and learn how to process the order from beginning to delivery
<b>9</b>	<b>Create and post your first ad</b>	Learn how to design a simple promotional ad and post it to bring in more people and sales
<b>10</b>	<b>Take feedback from customers</b>	Engage with your paid customers, take product feedback, and upload success stories or reviews on your website
<b>11</b>	<b>Get a repeat customer or referral</b>	Build trust with your paid customers to get either a repeat order or a referral
<b>12</b>	<b>Plan your business growth</b>	Make a 1 month plan to promote and grow your E-commerce store and earn more money

<b>Professional Service</b>		
<b>Week/ Milestone</b>	<b>Milestone</b>	<b>Description</b>
<b>1</b>	<b>Start your journey in professional services</b>	Learn what is a service and how you can earn money by offering your skill as a service
<b>2</b>	<b>Choose your service skill</b>	Choose one skill you are good at and turn it into a service that people will pay for
<b>3</b>	<b>Find your ideal customer</b>	Talk to people directly or through social media apps, understand who is willing to pay for your service and "WHY"
<b>4</b>	<b>Build your online profile and show what you offer</b>	Explore apps or websites such as WhatsApp Business, Upwork, or Urban Company, and learn how to create an online profile
<b>5</b>	<b>Write your service description</b>	Learn how to write your service in a simple and powerful way that gets people excited to try it
<b>6</b>	<b>Finalize your pricing and launch your service</b>	Learn how to set the right price that your customer finds fair and launch your service
<b>7</b>	<b>Get first 100 leads for your service</b>	Learn how to write catchy messages to promote your service in the market and get first 100 leads
<b>8</b>	<b>Offer a free service and learn from feedback</b>	Give your service for free to real users and use their feedback to make it better
<b>9</b>	<b>Make your first sale</b>	Get one paying customer for your service and work really hard to get a 5 star rating and a video testimonial
<b>10</b>	<b>Build trust with customers and get repeat orders</b>	Use your trust and good relationship with existing customers to get your first repeat customer
<b>11</b>	<b>Earn your first referral</b>	Start offering bonus or other additional services to your existing customers and get your first referral
<b>12</b>	<b>Plan your business growth</b>	Make a 1 month plan on how to get more customers and earn more money

<b>Retail Business</b>		
<b>Week/ Milestone</b>	<b>Milestone</b>	<b>Description</b>
<b>1</b>	<b>Learn how to start retail business</b>	Understand how retail works, the types of business you can start, and talking to local shops
<b>2</b>	<b>Choose your product and ideal customer</b>	Look at products in demand in your area and choose what you'll focus on based on your customer needs
<b>3</b>	<b>Find a supplier for your product</b>	Identify suppliers or wholesalers who can give you quality products/raw materials at good prices
<b>4</b>	<b>Setup your business</b>	Arrange your shop/home or prepare your product stocked with shopkeepers
<b>5</b>	<b>Decide product prices and prepare stock</b>	Decide the right selling price, maintain a simple record of sales, and get ready to handle cash or digital payments
<b>6</b>	<b>Launch your business</b>	Put up posters near your shop, and share the announcement with friends, family, and local community groups
<b>7</b>	<b>Promote your business in your area</b>	Learn how to use simple ways to promote your business locally
<b>8</b>	<b>Make your first sale</b>	Ensure the buying process is smooth and the customer is happy
<b>9</b>	<b>Take feedback from customers</b>	Engage with your paid customers and take product feedback
<b>10</b>	<b>Discover other platforms to grow your sales</b>	Learn how to set up a free WhatsApp Business account for your shop and collect orders there
<b>11</b>	<b>Get a repeat customer or referral</b>	Build trust with your paid customers to get either a repeat order or a referral
<b>12</b>	<b>Plan your business growth</b>	Make a 1 month plan to promote and grow your retail business and earn more money

<b>Miscellaneous</b>		
<b>Week/ Milestone</b>	<b>Milestone</b>	<b>Description</b>
<b>1</b>	<b>Start your journey as an Entrepreneur</b>	Learn what entrepreneurship means and how you can start earning by solving problems around you
<b>2</b>	<b>Identify and validate a problem to solve</b>	Look around yourself, talk to people, and pick one real problem that many people face
<b>3</b>	<b>Define your customer and their pain points</b>	Find out who will buy from you, what difficulties they face, and why they need your solution
<b>4</b>	<b>Generate business ideas and finalize one idea</b>	Think of different ways to solve the problem, compare options, and choose one idea to move ahead with
<b>5</b>	<b>Define your product or service</b>	Decide clearly what product or service you will provide and how it will solve the customer's problem
<b>6</b>	<b>Finalize your pricing and launch your business</b>	Set a fair price for your product or service and take the first step to launch your business
<b>7</b>	<b>Promote your business and get first 100 leads</b>	Tell people about your business using word of mouth, posters, or social media, and collect interest from 100 people
<b>8</b>	<b>Make your first sale</b>	Get your first paying customer and deliver your product or service with full effort
<b>9</b>	<b>Take feedback from customers and improve</b>	Listen to what customers say after using your product/service and make it better step by step
<b>10</b>	<b>Build trust with customers and get your first repeat order</b>	Keep your promise, give good quality, and motivate your customer to buy from you again
<b>11</b>	<b>Earn your first referral and expand your sales</b>	Ask happy customers to recommend you to friends and family so that you can grow your sales
<b>12</b>	<b>Plan your business growth</b>	Make a simple plan for the next month to get more customers, increase sales, and grow your business

## Evaluation Criteria

<b>Evaluation Component</b>	<b>Description</b>	<b>Weightage</b>
<b>Weekly Task Completion</b>	Timely submission of weekly tasks, including activities, reflection prompts, graded quizzes etc	60%
<b>Target Completion</b>	Performance-based evaluation on hitting revenue or profit targets (e.g., generating ₹10,000 revenue)	20%
<b>Final Project</b>	A comprehensive project depending the theme of the semester	20%

### Weekly Component:

Each week of the course follows a structured format designed to guide students from learning to doing, using simple, mobile-accessible components:

<b>Component</b>	<b>Duration</b>	<b>Description</b>
Action Lab	~4 hrs	<ul style="list-style-type: none"><li>- Hands-on task on the weekly concept</li><li>- Includes step-by-step guidance, templates, and worksheets</li><li>- Ends with a submission (e.g., video, reflection, or proof of action)</li></ul>
Learning Resources	Self-paced	<ul style="list-style-type: none"><li>- Videos, short readings, real-life stories, and tools to deepen understanding at their own pace</li></ul>
Check-in	Self-paced	<ul style="list-style-type: none"><li>- Quizzes &amp; Reflection prompts</li></ul>

## Recommended Readings

### **1. Start with Why** - Simon Sinek

Explains how great leaders and entrepreneurs inspire action by starting with a clear sense of “why.” Perfect for understanding purpose-driven entrepreneurship.

### **2. The Lean Startup** - Eric Ries

A must-read on how to test ideas, build fast, and learn quicker ideas for first-time entrepreneurs trying to reduce risk and start smart.

### **3. Contagious: How to Build Word of Mouth in the Digital Age** - Jonah Berger

Introduces how small ideas spread and how anyone can build buzz, useful for content creators and small sellers.

### **4. Shoe Dog** - Phil Knight (Founder of Nike)

An inspiring story of how a college runner built one of the world’s biggest shoe companies with almost nothing. Very relatable in its early struggle phase.

### **5. Rework** - Jason Fried & David Heinemeier Hansson

Offers fresh, simple ideas about doing business differently. Ideal for breaking traditional thinking and seeing how less can be more.

### **6. Ikigai: The Japanese Secret to a Long and Happy Life** - Héctor García & Francesc Miralles

Helps students reflect on passion, purpose, and how to connect what they love with what the world needs perfect for Value Map exercises.

### **7. Tools of Titans (Selected Chapters)** - Tim Ferriss

Pick short, digestible parts from interviews with entrepreneurs, creators, and doers. Recommended as optional deep-dives.

### **8. Zero to One: Notes on Startups, or How to Build the Future** - Peter Thiel co-written with Blake Masters

Indian, relatable, and deeply practical. Breaks down failure, money, motivation, and mindset in a raw and honest way.

### **9. Romancing The Balance Sheet** - Anil Lamba

This book will teach you all the intelligent ways of Good Financial Management.

### **10. Young Entrepreneurs (Series)**- The Better India / YourStory articles

Real Indian stories of youth starting businesses, snackable reads that show what’s possible.

## Syllabus Overview for Semester 1-5

Semester	Learning Focus	Learner's demonstration	Revenue Target
1	Setup & Launch	<b>Understand. Create. Start.</b>	₹10,000
2	Marketing Basics	<b>Engage. Share. Grow.</b>	₹40,000
3	Operations & Scale	<b>Earn. Deliver. Expand.</b>	₹80,000
4	Organic Growth	<b>Attract. Retain. Build.</b>	₹160,000
5	AI Automation & Finance mastery	<b>Simplify. Track. Sustain</b>	₹400,000

### Semester 1: Setup & Launch

In Term 1, students will explore what entrepreneurship means and how it connects to their daily lives. They will learn to identify problems, shape simple business ideas, and test them in real settings. This semester builds the foundation—mindset, observation, value creation, and action.

### Semester 2: Marketing Basics

In Term 2, students will learn how to attract customers and grow their visibility using digital platforms and community-based marketing strategies. Students will also begin to run paid advertising campaigns and learn how to optimize their marketing efforts.

### Semester 3: Operations & Scale

This semester focuses on the day-to-day operations of running a business, including order fulfillment, customer service, and logistics. Students will also focus on scaling operations as demand grows, with an emphasis on managing resources effectively.

### Semester 4: Organic Growth

Students will learn how to grow their businesses organically, using referrals, partnerships, and community engagement. This semester focuses on building a loyal customer base and using word-of-mouth marketing to increase reach and credibility.

### Semester 5: AI Automation & Financial Mastery

The final semester prepares students for long-term sustainability. Students integrate AI to improve productivity, automate routine tasks, and enhance decision-making. They also dive deep into financial planning, learning to set income goals, track expenses, understand profit margins, and create simple financial forecasts. This semester helps students solidify their entrepreneurial identity design systems for financial stability and scalability.