



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC Accredited with A+ grade

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ACADEMIC REGULATIONS

COURSE STRUCTURE AND SYLLABUS

For UG-R24

B.Tech – COMPUTER SCIENCE AND ENGINEERING

(Applicable for batches admitted from 2024-2025)



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113

Table: Program-B.Tech CSE (R24) curriculum grouping based on curriculum components semester wise

S.No	Course Component	I	II	III	IV	V	VI	VII	VIII	Total Credits
1	Basic Sciences Course(BS)	7	7	-	3	-	-	-	-	17
2	Engineering Science Courses (ES)	8.5	9	3	-	-	-	-	-	20.5
3	Humanity Science and Management Courses(HS)	3	-	1	2	-	-	2	-	9
4	Professional Core Courses (PC)	-	4.5	15	13.5	12	9	6	-	60
5	Program Elective Courses (PE)	-	-	-	-	3	6	3	3	15
6	Open Elective Courses (OE)	-	-	-	-	3	3	3	3	12
7	Internships/Project(s)	-	-	-	-	2	-	4	10	16
8	Skill Oriented Courses(SC)	-	-	1.5	2	2	2	2	-	9.5
9	Mandatory Courses (MC)	0.5	0.5	-	-	-	-	-	-	1
Total		19	21	20.5	21.5	22	20	20	16	160



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY **(Autonomous)**

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC Accredited with A+ grade

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113.

<https://avanthienggcollege.ac.in/>, E-mail: principal@avanthienggcollege.ac.in

Department of COMPUTER SCIENCE AND ENGINEERING

Proposed Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Induction Programme

S.No	Course Title	Category	L-T-P-C
1	Physical Activities--Sports, Yoga and Meditation, Plantation	MC	0-0-6-0
2	Career Counseling	MC	2-0-2-0
3	Orientation to all branches -- career options, tools, etc	MC	3-0-0-0
4	Orientation on admitted Branch -- corresponding labs, tools and platforms	EC	2-0-3-0
5	Proficiency Modules & Productivity Tools	ES	2-1-2-0
6	Assessment on basic aptitude and mathematical skills	MC	2-0-3-0
7	Remedial Training in Foundation Courses	MC	2-1-2-0
8	Human Values & Professional Ethics	MC	3-0-0-0
9	Communication Skills -- focus on Listening, Speaking, Reading, Writing skills	BS	2-1-2-0
10	Concepts of Programming	ES	2-0-2-0

Department of COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

I Year I Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				Lecture	Tutorial	Practical	Credits
1	BS	R24BS01	Linear Algebra and Calculus	3	0	0	3
2	HS	R24HS01	Communicative English	2	0	0	2
3	BS	R24BS02	Engineering Physics	3	0	0	3
4	ES	R24ES01	Basic Civil & Mechanical Engineering	3	0	0	3
5	ES	R24ES02	Problem Solving and Programming with C	3	0	0	3
6	BS	R24BS03	Engineering Physics Lab	0	0	2	1
7	HS	R24HS02	Communicative English Lab	0	0	2	1
8	ES	R24ES03	Problem Solving and Programming with C Lab	0	0	3	1.5
9	ES	R24ES04	IT Workshop	0	0	2	1
10	MC	R24MC01	Health and Wellness, Yoga, and Sports	0	0	1	0.5
Total				14	00	10	19

Category	Courses	Credits
BS-Basic Science Course	3	7
ES-Engineering Science Course	4	8.5
HS-Humanities and Social Science including Management Course	2	3
MC-Mandatory Course	1	0.5
Total	10	19

Department of COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

I Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	BS	R24BS04	Differential Equations and Vector Calculus	3	0	0	3
2	BS	R24BS05	Applied Chemistry	3	0	0	3
3	ES	R24ES06	Engineering Graphics	1	0	4	3
4	ES	R24ES05	Basic Electrical and Electronics Engineering	3	0	0	3
5	PC	R24CSPC01	Data Structures through C	3	0	0	3
6	BS	R24BS06	Applied Chemistry Lab	0	0	2	1
7	ES	R24ES07	Basic Electrical and Electronics Engineering Lab	0	0	3	1.5
8	PC	R24CSPC02	Data Structures through C Lab	0	0	3	1.5
9	ES	R24ES08	Engineering Workshop	0	0	3	1.5
10	MC	R24MC02	NSS/NCC/Scouts & Guides/Community Service	0	0	1	0.5
Total				13	00	16	21

Category	Courses	Credits
BS- Basic Science Course	3	7
ES-Engineering Science Course	4	9
PC-Professional Core Course	2	4.5
MC-Mandatory Course	1	0.5
Total	10	21

**Chairperson
Board of Studies (CSE)**



AVANTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113. www.avanthienggcollege.ac.in,
mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Program: **B.Tech Computer Science and Engineering**

Regulation: R24

II Year I Semester-Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	ES	R24CSES09	Mathematical Foundations of Computer Science	3	0	0	3
2	PC	R24CSPC03	Design and Analysis of Algorithms	3	0	0	3
3	PC	R24CSPC04	Object Oriented Programming using JAVA	3	0	0	3
4	PC	R24CSPC05	Digital Logic and Computer Organization	3	0	0	3
5	PC	R24CSPC06	Software Engineering and Project Management	3	0	0	3
6	PC	R24CSPC07	Object Oriented Programming using JAVA Lab	0	0	3	1.5
7	PC	R24CSPC08	Algorithms For Efficient coding Lab	0	0	3	1.5
8	SC	R24CSSC01	Full Stack Development	0	0	3	1.5
9	HS	R24CSHS05	Soft skills& Verbal Ability	0	0	2	1
10	MC	R24MC04	Indian Traditional Knowledge	2	0	0	-
Total				17	00	11	20.5

Category	Courses	Credits
ES- Engineering Science Course	1	03
PC-Proficitional Core Course	6	15
HS- Humanity Science and Management Course	1	01
SC- Skill Oriented Course	1	1.5
MC-Mandatory Course	1	00
Total	10	20.5



AVANTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113. www.avanthienggcollege.ac.in,
mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

II Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	BS	R24CSBS07	Probability & Statistics	3	0	0	3
2	PC	R24CSPC09	Formal Languages and Automata Theory	3	0	0	3
3	PC	R24CSPC10	Database Management Systems	3	0	0	3
4	HS	R24HS03	Universal Human Values-Understanding Harmony & Human Ethical Conduct	2	0	0	2
5	PC	R24CSPC11	Operating Systems	3	0	0	3
6	PC	R24CSPC12	Database Management Systems Lab	0	0	3	1.5
7	PC	R24CSPC13	Operating Systems Lab	0	0	3	1.5
8	PC	R24CSPC14	Advanced Coding Lab-I	0	0	3	1.5
9	SC	R24CSSC02	Python Programming	0	1	2	2
10	HS	R24HS04	Quantitative aptitude & Logical Reasoning	0	0	2	1
11	MC	R24MC03	Environmental Science	2	0	0	-
Total				16	01	13	21.5
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation							

Category	Courses	Credits
BS- Basic Sciences Course	1	03
PC-Proficitional Core Course	6	13.5
SC- Skill Oriented Course	1	2
HS- Humanity Science and Management Course	2	03
MC-Mandatory Course	1	--
Total	11	21.5

**Chairperson
Board of Studies (CSE)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B.Tech Computer Science and Engineering

Regulation: R24

III Year I Semester -Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC15	Compiler Design	3	0	0	3
2	PC	R24CSPC16	Computer Networks	3	0	0	3
3	HS	R24HS05	Managerial Economics and Financial Analysis	2	0	0	2
4	PE-1	R24CSPE1.1 R24CSPE1.2 R24CSPE1.3 R24CSPE1.4	1. Data Warehousing and Data Mining 2. Software Testing Methodologies 3. Human Computer Interaction 4. Information Retrieval and Search Engines 5. 12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
5	OE-1		OPEN ELECTIVE-1	3	0	0	3
6	PC	R24CSPC17	Compiler Design Lab	0	0	3	1.5
7	PC	R24CSPC18	Computer Networks Lab	0	0	3	1.5
8	SC	R24CSSC03	Full Stack Development-2	0	0	2	1
9	IN	R24IN01	Evaluation of Community Service Project	2	0	0	2
10	HS	R24HS06	Design Thinking & Innovation	1	0	2	2
11	HS	R24HS09	Advanced Aptitude and Communication Skills	2	0	0	1
12	MC	R24MC05	Constitution of India	2	0	0	0
Total				21	00	10	23

Category	Courses	Credits
PC-Professional Core Courses	4	9
PE- Professional Elective Courses	1	3
OE-Open Elective Courses	1	3
Community Service Project	1	2
SEC-Skill Enhancement course	1	1
HS-Humanities and Social Science including Management Course	3	5
MC-Mandatory Course	1	0
Total	12	23

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

III Year II Semester - Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC19	Artificial Intelligence & Machine Learning	3	0	0	3
2	PC	R24CSPC20	Cryptography & Network Security	3	0	0	3
3	PE-2	R24CSPE2.1 R24CSPE2.2 R24CSPE2.3 R24CSPE2.4	1. Cloud Computing 2. Cyber Security 3. DevOps 4. Augmented Reality and Virtual Reality 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	PE-3	R24CSPE3.1 R24CSPE3.2 R24CSPE3.3 R24CSPE3.4	1. Software Project Management 2. Big Data Analytics 3. Distributed Systems 4. Foundation of Data Science 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	OE-2		OPEN ELECTIVE-2	3	0	0	3
6	PC	R24CSPC21	Artificial Intelligence & Machine Learning	0	0	3	1
7	PC	R24CSPC22	Cryptography & Network Security Lab	0	0	3	1
8	SC	R24CSSC04	Competitive Coding Lab	0	0	2	1
9	HS	R24HS15	Corporate Readiness for Emerging Technologies	2	0	0	1
10	MC	R24MC07	Value Education	2	0	0	0
				19	00	8	19

Mandatory Industry Internship of 08 weeks duration during summer vacation

Category	Courses	Credits
PC-Professional Core Courses	4	8
PE- Professional Elective	2	6
OE-Open Elective	1	3
SEC-Skill Enhancement course	1	1
HS-Humanities and Social Science including Management Course	1	1
MC-Mandatory Course	1	0
Total	10	19

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B.Tech Computer Science and Engineering

Regulation: R24

IV Year I Semester -Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC23	Neural Networks & Deep Learning	3	0	0	3
2	PC	R24CSPC24	Data Warehousing and Data Mining	3	0	0	3
3	HS	R24HS14	Human Resources & Project Management	2	0	0	2
4	PE-4	R24CSPE4.1 R24CSPE4.2 R24CSPE4.3 R24CSPE4.4	1. Software Architecture & Design Patterns 2. Block chain Technology 3. Quantum Computing 4. AGENTIC AI SYSTEMS 5.12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	OE-3		OPEN ELECTIVE-3	3	0	0	3
6	SC	R24CSSC05	Prompt Engineering/ SWAYAM Plus - Certificate program in Prompt Engineering and ChatGPT	0	0	2	2
7	IN	R24INO2	Evaluation of Industry Internship (Done during III-II Summer vacation- 8 weeks)	2	0	0	2
8	MC	R24MC06	Technical paper writing and IPR	2	0	0	0
TOTAL				18	00	06	18

Category	Courses	Credits
PC-Professional Core Courses	2	6
PE- Professional Elective	1	3
OE-Open Elective	1	3
SC-Skill Enhancement course	1	2
MC-Mandatory Course	1	0
HS-Humanities and Social Science including Management Course	1	2
IN- Evaluation of Industry Internship	1	2
Total	9	18

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

IV Year II Semester - Course Structure

S. No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PE-5	R24CSPE5.1 R24CSPE5.2 R24CSPE5.3 R24CSPE5.4	1. Soft Computing 2. Mobile Ad-hoc Networks 3. Natural Language Processing 4. UI/UX Design 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
2	OE-4		OPEN ELECTIVE-4	3	0	0	3
3	PR	R24PR01	Internship & Project Work	0	0	20	12
Total				06	0	20	18

Category	Courses	Credits
PE- Professional Elective	1	3
OE-Open Elective	1	3
PJ- Internship & Project work	1	10
Total	03	16

Note: Student need to do at least ONE MOOC/NPTEL Course (of 3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R24 Regulations); they are allowed to register one semester in advance.

OPEN ELECTIVES

Course Code	Course Name	L	T	P	C
R24CSOE01	Data Structures	3	0	0	3
R24CSOE02	Database Management Systems				
R24CSOE03	Agile Methodologies				
R24CSOE04	Operating System				
R24CSOE05	Scripting Languages				
R24CSOE06	Computer Networks				
R24CSOE07	Mobile Computing				
R24CSOE08	Software Project Management				
R24CSOE09	OOPS through JAVA				
R24CSOE10	Fundamental of Cloud Computing				
R24CSOE11	Python Programming				
R24CSOE12	Object orient Analysis and Design				
R24CSOE13	Big Data Analytics				
R24CSOE14	Web Services				
R24CSOE15	Quantum Computing				
R24CSOE16	Computer Architecture				
R24CSOE17	Fundamentals of Information Security				
R24CSOE18	Ethical Hacking				
R24CSOE19	Data Mining				
R24CSOE20	Biometrics				

Chairperson
Board of Studies (CSE)



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC Accredited with A+ grade

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ACADEMIC REGULATIONS

COURSE STRUCTURE AND SYLLABUS

For UG-R24

B.Tech – COMPUTER SCIENCE AND ENGINEERING

(Applicable for batches admitted from 2024-2025)



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC Accredited with A+ grade

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113.

www.avanthienggcollege.ac.in, mail: principal@avanthienggcollege.ac.in

Academic Regulations 2024 (R24) for B. Tech (Regular-Full time)

(Effective for the students admitted into I year from the

Academic Year: 2024-2025 onwards)

1. Award of the B.Tech Degree

(a) **Award of the B.Tech. Degree/ B.Tech. Degree with a Minor:** If he/ she fulfils the following:

- (i) Pursues a course of study for not less than four academic years and not more than eight academic years. However, for the students availing Gap year facility this period shall be extended by two years at the most and these two years would be in addition to the maximum period permitted for graduation (Eight Years).
- (ii) Registers for 160 credits and secures all 160 credits.

(b) **Award of B.Tech. Degree with Honors:** If he / she fulfils the following:

- (i) Student secures additional 15 credits fulfilling all the requisites of a B.Tech. Program i.e., 160 credits.
- (ii) Registering for Honors is optional.
- (iii) Honors are to be completed simultaneously with B.Tech. Programme.

2. Students, who fail to fulfill all the academic requirements for the award of the degree within eight academic years from the year of their admission, shall forfeit their seat in B.Tech. Course and their admission stands cancelled. This clause shall be read along with clause 1 a) i).

3. Courses of study:

The following courses of study are offered at present with specialization in the B.Tech

S.No.	Branch Code - Abbreviation	Branch
1.	02-EEE	Electrical and Electronics Engineering
2.	03-ME	Mechanical Engineering
3.	04-ECE	Electronics and Communication Engineering
4.	05-CSE	Computer Science and Engineering

5.	42-CSM	Computer Science and Engineering (Artificial Intelligence and Machine Learning)
6.	44-CSD	Computer Science and Engineering (Data Science)

4. Admissions

Admission to the B. Tech Program shall be made subject to the eligibility, qualifications and specialization prescribed by the A.P. State Government/ Institution from time to time. Admissions shall be made either based on the merit rank obtained by the student in the common entrance examination conducted by the A.P. Government /Institution or to any other order of merit approved by the A.P. Government / Institution, subject to reservations as prescribed by the Government/ Institution from time to time.

5. Program related terms

Credit: A unit by which the course work is measured. It determines the number of hours of instruction required per week. One credit is equivalent to one hour of teaching (Lecture/Tutorial) or two hours of practical work/field work per week.

Credit Definition:

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

- a) **Semester:** A semester comprises 90 working days.
- b) **Academic Year:** Two consecutive (one odd + one even) semesters constitute one academic year.
- c) **Choice Based Credit System (CBCS):** The CBCS provides a choice for students to select from the prescribed courses.

6. Semester / Credits:

- i) A semester comprises 90 working days and an academic year is divided into two semesters.
- ii) The summer term is for eight weeks during summer vacation. Internship / apprenticeship / work-based vocational education and training can be carried out during the summer term, especially by students who wish to exit after two semesters or four semesters of study.
- iii) Regular courses may also be completed well in advance through MOOCs satisfying prerequisites.

7. Structure of Undergraduate Programme

All courses offered for the undergraduate program (B. Tech.) are broadly classified as follows:

S.No.	Category	Breakup of Credits (Total 160)	Percentage of total credits	AICTE Recommendation(%)
1.	Humanities and Social Science including Management (HM)	13	8 %	8 – 9%
2.	Basic Sciences (BS)	20	13 %	12 - 16%
3.	Engineering Sciences (ES)	23.5	14%	10 – 18%
4.	Professional Core (PC)	54.5	34 %	30 – 36%
5.	Electives – Professional (PE) & Open (OE); Domain Specific Skill Enhancement Courses (SEC)	33	21 %	19 - 23%
6.	Internships & Project work (PR)	16	10 %	8 – 11%
7.	Mandatory Courses (MC)	Non-credit	Non-credit	-

8. Course Classification:

All subjects/ courses offered for the undergraduate programme in Engineering (B.Tech. Degree programmes) are broadly classified as follows:

S. No.	Broad Course Classification	Course Category	Description
1.	Foundation Core Courses	Foundation courses	Includes Mathematics, Physics and Chemistry; fundamental engineering courses; Humanities, Social sciences and Management courses
2.	Professional Core Courses	Professional Core Courses (PC)	Includes subjects related to the parent discipline/department/branch of Engineering
3.	Open Elective Courses	Professional Elective Courses (PE)	Include selective subjects related to the parent discipline/department/ branch of Engineering
		Open Elective Courses (OE)	Elective subjects which include interdisciplinary Subjects or subjects in an area outside the parent discipline/ department/ branch of Engineering
		Domains Specific Skill Enhancement Courses (SEC)	Interdisciplinary/ job-oriented / domain courses which are relevant to the industry
4.	Project Internships	Project	B.Tech. Project (or) Major Project
		Internships	Summer Internships–Community based and Industry Internships; Industry oriented Full Semester Internship
5.	Audit Courses	Mandatory non-credit courses	Covering subjects of developing desired attitude among the learners

9. Programme Pattern

- i. Total duration of the B.Tech (Regular) Programme is four academic years.
- ii. Each academic year of study is divided into two semesters.
- iii. Minimum number of instruction days in each semester is 90 days
- iv. There shall be mandatory student induction program for fresher's, with three- week duration before the commencement of first semester. Physical activity, Creative Arts, Universal Human Values, Literary, Proficiency Modules, Lectures by Eminent People, Visits to local Areas, Familiarization to Dept. /Branch & Innovations etc., are included as per the guidelines issued by AICTE.
- v. Health/wellness/yoga/sports and NCC / NSS / Scouts & Guides / Community service activities are made mandatory as credit courses for all the undergraduate students
- vi. Courses like Environmental Sciences, Indian Constitution, and Technical Paper Writing & IPR are offered as non-credit mandatory courses for all the undergraduate students.
- vii. Design Thinking for Innovation & Tinkering Labs is made mandatory as credit courses for all the undergraduate students.
- viii. Increased flexibility for students through an increase in the elective component of the curriculum, with 05 Professional Elective courses and 04 Open Elective courses.
- ix. Professional Elective Courses, include the elective courses relevant to the chosen specialization/branch. Proper choice of professional elective courses can lead to students specializing in emerging areas within the chosen field of study.
- x. A total of 04 Open Electives are offered in the curriculum. A student can complete the requirement for B.Tech. Degree with a Minor within the 160 credits by opting for the courses offered through various verticals/tracks under Open Electives.
- xi. While choosing the electives, students shall ensure that they do not opt for the courses with syllabus contents similar to courses already pursued.
- xii. A pool of interdisciplinary/job-oriented/domain skill courses which are relevant to the industry are integrated into the curriculum of all disciplines. There shall be 05 skill-oriented courses offered during III to VII semesters. Among the five skill courses, four courses shall focus on the basic and advanced skills related to the domain/interdisciplinary courses and the other shall be a soft skills course.
- xiii. Students shall undergo mandatory summer internships, for a minimum of eight weeks duration at the end of second and third year of the programme. The internship at the end of second year shall be community oriented and industry internship at the end of third year
- xiv. There shall also be mandatory full internship in the final semester of the programme along with the project work.

- xv. Undergraduate degree with Honors is introduced by the Institution for the students having good academic record.
- xvi. Each college shall take measures to implement Virtual Labs (<https://www.vlab.co.in>) which provide remote access to labs in various disciplines of Engineering and will help student in learning basic and advanced concept through remote experimentation. Student shall be made to work on virtual lab experiments during the regular labs.
- xvii. Each college shall assign a faculty advisor/ mentor after admission to a group of students from same department to provide guidance in courses registration/ career growth/ placements/ opportunities for higher studies/ GATE/ other competitive exams etc.
- xviii. Preferably 25% of course work for the theory courses in every semester shall be conducted in the blended mode of learning.

10. Evaluation Process

The performance of a student in each semester shall be evaluated subject wise with a maximum of 100 marks for theory and 100 marks for practical subject. Summer Internships shall be evaluated for 50 marks, Full Internship & Project work in final semester shall be evaluated for 200 marks, and mandatory courses with no credits shall be evaluated for 30 MID semester marks.

A student has to secure not less than 35% of marks in the end examination and a minimum of 40% of marks in the sum total of the mid semester and end examination marks taken together for the theory, practical, design, drawing subject or project etc. In case of a mandatory course, he / she should secure 40% of the total marks.

Theory Courses

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

- i) For theory subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End-Examination.
- ii) For practical subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End- Examination.
- iii) If any course contains two different branch subjects, the syllabus shall be written in two parts with 2.5 units each (Part-A and Part-B) and external examination question paper shall be set with two parts each for 35 marks.

- iv) If any subject is having both theory and practical components, they will be evaluated separately as theory subject and practical subject. However, they will be given same subject code with an extension of “T for theory subject and P for practical” subject.

a) Continuous Internal Evaluation

- i) For theory subjects, during the semester, there shall be two midterm examinations. Each midterm examination shall be evaluated for 30 marks of which 10 marks for objective questions / short answer questions (20 minutes duration), 15 marks for subjective paper (90 minutes duration) and 5 marks for assignment.
- ii) Objective paper shall contain for 05 short answer questions with 2 marks each or maximum of 20 bits / multiple choice questions (MCQ's) for 10 marks. Subjective paper shall contain 3 questions and each question carries 10 marks. The marks obtained in the subjective paper are condensed to 15 marks.

Note:

- The objective paper shall be prepared in line with the quality of competitive examinations questions.
 - The subjective paper shall contain 3 questions of equal weightage of 10 marks. Any fraction shall be rounded off to the next higher mark.
 - The objective paper shall be conducted either online or offline by the respective department on the day of subjective paper test.
 - Assignments shall be in the form of problems, mini projects, design problems, slip tests, quizzes etc., depending on the course content. It should be continuous assessment throughout the semester and the average marks shall be considered.
- iii) If the student is absent for the mid semester examination, no re-exam shall be conducted and mid semester marks for that examination shall be considered as zero.
- iv) First midterm examination shall be conducted for 2.5 (I Unit, II Unit and III Unit half part) units of syllabus the second midterm examination shall be conducted for remaining 2.5 Units (III Unit half part, IV and V units).
- v) Final mid semester marks shall be arrived at by considering the marks secured by the student in both the mid examinations with 80% weightage given to the better mid exam and 20% to the other.

For Example:

- Marks obtained in first mid: 25
- Marks obtained in second mid: 20
- Final mid semester Marks: $(25 \times 0.8) + (20 \times 0.2) = 24$

If the student is absent for any one midterm examination, the final mid semester marks shall be arrived at by considering 80% weightage to the marks secured by the student in the appeared examination and zero to the other. For Example:

- Marks obtained in first mid: Absent
- Marks obtained in second mid: 25
- Final mid semester Marks: $(25 \times 0.8) + (0 \times 0.2) = 20$

b) End Examination Evaluation:

End examination of theory subjects shall have the following pattern:

- (i) There shall be 6 questions and all questions are compulsory.
- (ii) Question I shall contain 10 compulsory short answer questions for a total of 20 marks such that each question carries 2 marks. There shall be 2 short answer questions from each unit
- (iii) In each of the questions from 2 to 6, there shall be either / or type questions of 10 marks each and each question have internal choice. Student shall answer any one of them.
- (iv) The questions from 2 to 6 shall be set by covering one unit of the syllabus for each question.

End examination of theory subjects consisting of two parts of different subjects, for Example: Basic Electrical & Electronics Engineering shall have the following pattern:

- (i) Question paper shall be in two parts viz., Part A and Part B with equal weightage of 35 marks each.
- (ii) In each part, question 1 shall contain 5 compulsory short answer questions for a total of 5 marks such that each question carries 1 mark.
- (iii) In each part, questions from 2 to 4, there shall be either/or type questions of 10 marks each. Student shall answer any one of them.
- (iv) The questions from 2 to 4 shall be set by covering one unit of the syllabus for each question.

Practical Courses

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

- a) For practical courses, there shall be a continuous evaluation during the semester for 30 sessional marks and end examination shall be for 70 marks.
- b) Day-to-Day work in the laboratory shall be evaluated for 15 marks by the concerned laboratory teacher based on the regularity / record / viva and 15 marks for the internal test.
- c) The end examination shall be evaluated for 70 marks, conducted by the

concerned laboratory teacher and a senior expert in the subject from the same department.

- Procedure: 20 marks
- Experimental work & Results: 30 marks
- Viva Voce: 20 marks.

In a practical subject consisting of two parts (Eg: Basic Electrical & Electronics Engineering Lab), the examination shall be conducted for 70 marks as a single laboratory in 3 hours. Mid semester examination shall be evaluated 30 marks in each part. Mid semester examination shall be evaluated as above for 30 marks in each part and final mid semester marks shall be arrived by considering the average of marks obtained in two parts.

- d) For the subject having design and/or drawing, such as Engineering Drawing, the distribution of marks shall be 30 for mid semester evaluation and 70 for end examination.

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

Day-to-Day work shall be evaluated for 15 marks by the concerned subject teacher based on the reports/submissions prepared in the class. And there shall be two midterm examinations in a semester for duration of 2 hours each for 15 marks with weightage of 80% to better mid marks and 20% for the other. The subjective paper shall contain 3 either or type questions of equal weightage of 5 marks. There shall be no objective paper in mid semester examination. The sum of day-to-day evaluation and the mid semester marks will be the final sessional marks for the subject.

The end examination pattern for Engineering Graphics, shall consists of 5 questions, either/or type, of 14 marks each. There shall be no objective type questions in the end examination.

- e) There shall be no external examination for mandatory courses with zero credits. However, attendance shall be considered while calculating aggregate attendance and student shall be declared to have passed the mandatory course only when he/she secures 40% or more in the internal examinations. In case, the student fails, a re-examination shall be conducted for failed candidates for 30 marks satisfying the conditions mentioned in item 1 & 2 of the regulations.
- f) The laboratory records and mid semester test papers shall be preserved for a minimum of 3 years in the respective institutions as per the University norms and shall be produced to the Committees of the University as and when the same are asked for.

11. Skill oriented Courses

- i) There shall be five skill-oriented courses offered during III to VII semesters.
- ii) Out of the five skill courses, two shall be skill-oriented courses from the same domain of their main three skill courses, one shall be a soft skill course and the remaining two shall be skill-advanced courses from the same domain/Interdisciplinary/Job oriented.
- iii) The course shall carry 100 marks and shall be evaluated through continuous assessments during the semester for 30 sessional marks and end examination shall be for 70 marks. Day-to-Day work in the class / laboratory shall be evaluated for 30 marks by the concerned teacher based on the regularity/ assignments / viva/ mid semester test. The end examination similar to practical examination pattern shall be conducted by the concerned teacher and an expert in the subject nominated by the principal.
- iv) The Head of the Department shall identify a faculty member as coordinator for the course. A committee consisting of the Head of the Department, coordinator and a senior Faculty member nominated by the Head of the Department shall monitor the evaluation process. The marks / grades shall be assigned to the students by the above committee based on their performance.
- v) The student shall be given an option to choose either the skill courses being offered by the college or to choose a certificate course being offered by industries/Professional bodies or any other accredited bodies. If a student chooses to take a Certificate Course offered by external agencies, the credits shall be awarded to the student upon producing the Course Completion Certificate from the agency. A committee shall be formed at the level of the college to evaluate the grades/marks given for a course by external agencies and convert to the equivalent marks/grades.
- vi) If a student prefers to take a certificate course offered by external agency, the department shall mark attendance of the student for the remaining courses in that semester excluding the skill course in all the calculations of mandatory attendance requirements upon producing a valid certificate as approved by the Institute.

12. Massive Open Online Courses (MOOCs):

A Student has to pursue and complete one course compulsorily through MOOCs approved by the Institute. A student can pursue courses other than core through MOOCs and it is mandatory to complete one course successfully through MOOCs for awarding the degree. A student is not permitted to register and pursue core courses through MOOCs.

A student shall register for the course (Minimum of either 8 weeks or 12 weeks) offered through MOOCs with the approval of Head of the Department. The Head of the Department shall appoint

one mentor to monitor the student's progression. The student needs to earn a certificate by passing the exam. The student shall be awarded the credits assigned in the curriculum only by submission of the certificate. Examination fee, if any, will be borne by the student.

Students who have qualified in the proctored examinations conducted through MOOCs platform can apply for credit transfer as specified and are exempted from appearing internal as well as external examination (for the specified equivalent credit course only) conducted by the Institute.

Necessary amendments in rules and regulations regarding adoption of MOOC courses would be proposed from time to time.

13. Credit Transfer Policy

Adoption of MOOCs is mandatory, to enable Blended model of teaching-learning as also envisaged in the NEP 2020. As per University Grants Commission (Credit Framework for Online Learning Courses through SWAYAM) Regulation, 2016, the Institution shall allow up to a maximum of 20% of the total courses being offered in a particular programme i.e., maximum of 32 credits through MOOCs platform.

- i) The AIET (A) shall offer credit mobility for MOOCs and give the equivalent credit weightage to the students for the credits earned through online learning courses.
- ii) Student registration for the MOOCs shall be only through the respective department of the institution, it is mandatory for the student to share necessary information with the department.
- iii) The Credit transfer policy will be applicable to the Professional & Open Elective courses only.
- iv) The concerned department shall identify the courses permitted for credit transfer.
- v) The Institution shall notify at the beginning of semester the list of the online learning courses eligible for credit transfer.
- vi) The institution shall designate a faculty member as a Mentor for each course to guide the students from registration till completion of the credit course.
- vii) The Institute shall ensure no overlap of MOOC exams with that of the Institute examination schedule. In case of delay in results, the Institute will re-issue the marks sheet for such students.
- viii) Student pursuing courses under MOOCs shall acquire the required credits only after successful completion of the course and submitting a certificate issued by the competent authority along with the percentage of marks and grades.
- ix) The institution shall submit the following to the examination section of the Institute:
 - (a) List of students who have passed MOOC courses in the current semester along with the certificate of completion.

- (b) Undertaking form filled by the students for credit transfer.
- x) The Institution` shall resolve any issues that may arise in the implementation of this policy from time to time and shall review its credit transfer policy in the light of periodic changes brought by UGC, SWAYAM, NPTEL and state government.

Note: Students shall be permitted to register for MOOCs offered through online platforms approved by the Institute from time to time.

14. Academic Bank of Credits (ABC)

The Institute has implemented Academic Bank of Credits (ABC) to promote flexibility in curriculum as per NEP 2020 to

- i) Provide option of mobility for learners across the universities of their choice.
- ii) Provide option to gain the credits through MOOCs from approved digital platforms.
- iii) Facilitate award of certificate/diploma/degree in line with the accumulated credits in ABC
- iv) Execute Multiple Entry and Exit system with credit count, credit transfer and credit acceptance from students account.

15. Mandatory Internships

Summer Internships: Two summer internships either onsite or virtual each with a minimum of 08 weeks duration, done at the end of second and third years, respectively are mandatory. It shall be completed in collaboration with local industries, Govt. Organizations, construction agencies, Power projects, software MNCs or any industries in the areas of concerned specialization of the Undergraduate program. One of the two summer internships at the end of second year (Community Service Project) shall be society oriented and shall be completed in collaboration with government organizations/ NGOs & others. The other internship at the end of third year is Industry Internship and shall be completed in collaboration with Industries. The student shall register for the internship as per course structure after commencement of academic year. The guidelines issued by the APSCHE / Institute shall be followed for carrying out and evaluation of Community Service Project and Industry Internship.

Evaluation of the summer internships shall be through the departmental committee. A student will be required to submit a summer internship report to the concerned department and appear for an oral presentation before the departmental committee comprising of Head of the Department, supervisor of the internship and as senior faculty member of the department. A certificate of successful completion from industry shall be included in the report. The report and the oral presentation shall carry 50% weightage each. It shall be evaluated for 50 external marks. There shall be no internal

marks for Summer Internship. A student shall secure minimum 40% of marks for successful completion. In case, if a student fails, he/she shall reappear as and when semester supplementary examinations are conducted by the Institute.

Full Semester Internship and Project work:

In the final semester, the student should mandatorily register and undergo internship (onsite/virtual) and in parallel he/she should work on a project with well-defined objectives. At the end of the semester the candidate shall submit an internship completion certificate and a project report. A student shall also be permitted to submit project report on the work carried out during the internship. The project report shall be evaluated with an external examiner. The total marks for project work 200 marks and distribution shall be 60 marks for internal and 140 marks for external evaluation. The supervisor assesses the student for 30 marks (Report: 15 marks, Seminar: 15 marks). At the end of the semester, all projects shall be show cased at the department for the benefit of all students and staff and the same is to be evaluated by the departmental Project Review Committee consisting of supervisor, a senior faculty and HOD for 30 marks. The external evaluation of Project Work is a Viva-Voce Examination conducted in the presence of internal examiner and external examiner appointed by the Institute and is evaluated for 140 marks.

The institute shall facilitate and monitor the student internship programs. Completion of internships is mandatory, if any student fails to complete internship, he/she will not be eligible for the award of degree. In such cases, the student shall repeat and complete the internship.

16. Guidelines for offering Minor

To promote inter disciplinary knowledge among the students; the students admitted into B.Tech.in a major stream / branch are eligible to obtain degree in Minor in another stream.

- i) The Minor program requires the completion of 12 credits in Minor stream chosen.
- ii) Two courses for 06 credits related to a Minor are to be pursued compulsorily for them in or degree, but may be waived for students who have done similar / equivalent courses. If waived for a student, then the student must take an extra elective course in its place. It is recommended that students should complete the compulsory courses (or equivalents) before registering for the electives.
- iii) Electives (minimum of 2 courses) to complete a total of 12 credits.

Note: A total of 04 Open Electives are offered in the curriculum. A student can complete the requirement for Minor within the 160 credits by opting for the courses offered through various verticals/tracks under Open Electives.

17. Guidelines for offering Honors

The objective of introducing B.Tech. (Honors) is to facilitate the students to choose additionally the specialized courses of their choice and build their competence in a specialized area in the UG level. The programme is a best choice for academically excellent students having good academic record and interest towards higher studies and research.

- i) Honors are introduced in the curriculum of all B.Tech. Programs offering a major degree and is applicable to all B.Tech (Regular and Lateral Entry) students admitted in Engineering.
- ii) A student shall earn additional 15 credits for award of B.Tech. (Honors) degree from same branch/ department/ discipline registered for major degree. This is in addition to the credits essential for obtaining the under graduate degree in Major Discipline (i.e., 160 credits).
- iii) A student is permitted to register for Honors in IV semester after the results of III Semester are declared and students may be allowed to take maximum two subjects per semester pertaining to the Honors from V Semester onwards.
- iv) The concerned head of the department shall arrange separate class work and timetable of the courses offered under Honors program.
- v) Courses that are used to fulfill the student's primary major may not be double counted towards the Honors. Courses with content substantially equivalent to courses in the student's primary Major may not be counted towards the Honors.
- vi) Students can complete the courses offered under Honors either in the college or in online platforms like SWAYAM with a minimum duration of 12 weeks for a 3-credit course and 8 weeks duration for a 2-credit course satisfying the criteria for credit mobility. If the courses under Honors are offered in conventional mode, then the teaching and evaluation procedure shall be similar to regular B. Tech courses.
- vii) The attendance for the registered courses under Honors and regular courses offered for Major degree in a semester are to be considered separately.
- viii) A student shall maintain an attendance of 75% in all registered courses under Honors to be eligible for attending semester end examinations.
- ix) A student registered for Honors shall pass in all subjects that constitute the requirement for the Honors degree program. No class/division (i.e., second class, first class and distinction, etc.) shall be awarded for Honors degree programme.
- x) If a student drops or is terminated from the Honors program, the additional credits so far earned cannot be converted into open or core electives; they will remain extra. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.

- xi) The Honors will be mentioned in the degree certificate as Bachelor of Technology (Honors) in XYZ. For example, B.Tech. (Honors) in Mechanical Engineering.

Enrolment into Honors:

- i) Students of a Department/Discipline are eligible to opt for Honors program offered by the same Department/Discipline
- ii) The enrolment of student into Honors is based on the CGPA obtained in the major degree program. CGPA shall be taken up to III semester incase of regular entry students and only III semester in case of lateral entry students. Students having 7 CGPA without any backlog subjects will be permitted to register for Honors.
- iii) If a student is detained due to lack of attendance either in Major or in Honors, registration shall be cancelled.
- iv) Transfer of credits from Honors to regular B.Tech degree and vice-versa shall not be permitted.
- v) Honors are to be completed simultaneously with a Major degree program.

Registration for Honors:

- i) The eligible and interested students shall apply through the HOD of his / her parent department. The whole process should be completed within one week before the start of every semester. Selected students shall be permitted to register the courses under Honors.
- ii) The selected students shall submit their willingness to the principal through his/her parent department offering Honors. The parent department shall maintain the record of student pursuing the Honors.
- iii) The students enrolled in the Honors courses will be monitored continuously. An advisor/mentor from parent department shall be assigned to a group of students to monitor the progress.
- iv) There is no fee for registration of subjects for Honors program offered in offline at the respective institutions.

18. Attendance Requirements:

- i) A student shall be eligible to appear for the Institution's / Institute's external examinations if he/she acquires a minimum of 40% attendance in each subject and 75% of attendance in aggregate of all the subjects. b) Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester may be granted by the College's Academic Committee.
- ii) Shortage of Attendance below 65% in aggregate shall in NO CASE be condoned.

- iii) A stipulated fee shall be payable towards condonation of shortage of attendance to the Institute.
- iv) Students whose shortage of attendance is not condoned in any semester are not eligible to take their end examination of that class and their registration shall stand cancelled.
- v) A student will not be promoted to the next semester unless he/she satisfies the attendance requirements of the present semester. They may seek readmission for that semester from the date of commencement of class work.
- vi) If any candidate fulfils the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.
- vii) If the learning is carried out in blended mode (both offline & online), then the total attendance of the student shall be calculated considering the offline and online attendance of the student.
- viii) For induction programme attendance shall be maintained as per AICTE norms.

19. Promotion Rules:

The following academic requirements must be satisfied in addition to the attendance requirements mentioned in section 18.

- i) A student shall be promoted from first year to second year if he/she fulfils the minimum attendance requirement as per college norms.
- ii) A student will be promoted from II to III year if he/she fulfils the academic requirement of securing 40% of the credits (any **decimal** fraction should be **rounded off** to **lower** digit) up to in the subjects that have been studied up to III semester.
- iii) A student shall be promoted from III year to IV year if he/she fulfils the academic requirements of securing 40% of the credits (any **decimal** fraction should be **rounded off** to **lower** digit) in the subjects that have been studied up to V semester. And in case a student is detained for want of credits for a particular academic year by ii) & iii) above, the student may make up the credits through supplementary examinations and only after securing the required credits he/she shall be permitted to join in the V semester or VII semester respectively as the case may be.
- iv) When a student is detained due to lack of credits/ shortage of attendance he/she may be re-admitted when the semester is offered after fulfillment of academic regulations. In such case, he/she shall be in the academic regulations into which he/she is readmitted.

20. Grading:

As measure of the student's performance, a 10-point Absolute Grading System using the following Letter Grades and corresponding percentage of marks shall be followed: After each course is evaluated for 100 marks, the marks obtained in each course will be converted to a corresponding

letter grade as given below, depending on the range in which the marks obtained by the student fall.

Structure of Grading of Academic Performance

Range in which the marks in the subject fall	Grade	Grade points Assigned
90 & above	S (Superior)	10
80 - 89	A (Excellent)	9
70 - 79	B (Very Good)	8
60 - 69	C (Good)	7
50 - 59	D (Average)	6
40-49	E (Pass)	5
<40	F (Fail)	0
Absent	Ab (Absent)	0

- i) A student obtaining Grade “F” or Grade “Ab” in a subject shall be considered failed and will be required to reappear for that subject when it is offered the next supplementary examination.
- ii) For non-credit audit courses, “Satisfactory” or “Unsatisfactory” shall be indicated instead of the letter grade and this will not be counted for the computation of SGPA/CGPA/ Percentage.

Computation of Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA):

SGPA: The Semester Grade Point Average (SGPA) is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses under gone by a student, i.e.,

$$SGPA = \frac{\sum (C_i \times G_i)}{\sum C_i}$$

Where, C_i is the number of credits of the i^{th} subject and G_i is the grade point scored by the student in the i^{th} course.

CGPA: The Cumulative Grade Point Average (CGPA) will be computed in the same manner considering all the courses under gone by a student over all the semesters of a program, i.e.

$$CGPA = \frac{\sum (C_i \times S_i)}{\sum C_i}$$

Where “ S_i ” is the SGPA of the i^{th} semester and C_i is the total number of credits up to that semester.

Both SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts. While computing the SGPA the subjects in whom the student is awarded Zero grade points will also be included.

Grade Point: It is a numerical weight allotted to each letter grade on a 10-point scale.

Letter Grade: It is an index of the performance of students in a said course. Grades are denoted by the letters S, A, B, C, D, E and F.

Award of Class:

After a student has satisfied the requirements prescribed for the completion of the program and are eligible for the award of B.Tech. Degree, he/she shall be placed in one of the following four classes:

Class Awarded	CGPA Secured
First Class with Distinction	≥ 7.5 (Without any supplementary appearance)
First Class	$\geq 6.5 < 7.5$
Second Class	$\geq 5.5 < 6.5$
Pass Class	$\geq 5.0 < 5.5$

Note: Students who have written supplementary examinations to fulfill the credit requirement will not be awarded First Class with Distinction. For such students the highest degree that is awarded will be First Class Only.

CGPA to Percentage conversion Formula: $(CGPA - 0.75) \times 10$

21. With-holding of Results

If the candidate has any dues not paid to the Institute or if any case of indiscipline or malpractice is pending against him/her, the result of the candidate shall be with held in such cases.

22. Multiple Entry/ Exit Option

(a) Exit Policy:

The students can choose to exit the four-year programme at the end of First / Second/ Third year.

- i) **UG Certificate (in Field of study/ discipline)** - Programme duration: First year (first two semesters) of the undergraduate programme, 40 credits followed by an additional exit 10-credit bridge course(s) lasting two months, including at least 6- credit job-specific internship / apprenticeship that would help the candidates acquire job-ready competencies required to

enter the workforce.

- ii) **UG Diploma (in Field of study/ discipline)** – Programme duration: First two years (first four semesters) of the undergraduate programme, 80 credits followed by an additional exit 10-credit bridge course(s) lasting two months, including at least 6- credit job-specific internship / apprenticeship that would help the candidates acquire job-ready competencies required to enter the workforce.
- iii) **Bachelor of Science (in Field of study/discipline) i.e., B.Sc. Engineering in (Field of study/discipline)** - Programme duration: First three years (first six semesters) of the undergraduate programme, 120 credits.

(b) Entry Policy:

Modalities on multiple entry by the student into the B.Tech. Programme will be provided in due course of time.

Note: The Institute shall resolve any issues that may arise in the implementation of Multiple Entry and Exit policies from time to time and shall review the policies in the light of periodic changes brought by UGC, AICTE/APSCHE and State government.

23. Gap Year Concept:

Gap year concept for Student Entrepreneur in Residence is introduced and outstanding students who wish to pursue entrepreneurship / become entrepreneur are allowed to take a break of one year at any time after II year to pursue full-time entrepreneurship programme/ to establish startups. This period may be extended to two years at the most and these two years would not be counted for the time for the maximum time for graduation. The principal of the respective college shall forward such proposals submitted by the students to the Institute. An evaluation committee constituted by the Institute shall evaluate the proposal submitted by the student and the committee shall decide whether to permit the student(s) to avail the Gap Year or not.

24. Transitory Regulations

Discontinued, detained, or failed candidates are eligible for readmission as and when the semester is offered after fulfillment of academic regulations. Candidates who have been detained for want of attendance or not fulfilled academic requirements or who have failed after having undergone the course in earlier regulations or have discontinued and wish to continue the course are eligible for admission into the unfinished semester from the date of commencement of class work with the same or equivalent subjects as and when subjects are offered, subjected to Section 2 and they will follow the academic regulations into which they are readmitted.

Candidates who are permitted to avail Gap Year shall be eligible for re-joining into the succeeding year of their B. Tech from the date of commencement of class work, subjected to Section 2 and they

will follow the academic regulations in to which they are readmitted.

25. Minimum Instruction Days for a Semester:

The minimum instruction days including internal exams for each semester shall be 90 days.

26. Medium of Instruction:

The medium of instruction of the entire B.Tech undergraduate programme in Engineering (including examinations and project reports) will be in English only.

27. Student Transfers:

Student transfers shall be as per the guidelines issued by the Government of Andhra Pradesh / JNTU-GV and the Institute from time to time.

28. General Instructions:

- i) The academic regulations should be read as a whole for purpose of any interpretation.
- ii) Malpractices rules-nature and punishments are appended.
- iii) Where the words “he”, “him”, “his”, occur in the regulations, they also include “she”, “her”, “hers”, respectively.
- iv) In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Principal / Dean-Academics of the institution is final.
- v) The Institute may change or amend the academic regulations or syllabi at any time and the changes or amendments shall be made applicable to all the students on rolls with effect from the dates notified by the Institute.
- vi) In the case of any doubt or ambiguity in the interpretation of the guidelines given, the decision of the Principal / Dean-Academics of the institution is final.

ACADEMIC REGULATIONS (R24) for B.Tech (LATERAL ENTRY SCHEME)

(Effective for the students getting admitted into II year through Lateral Entry Scheme from the
Academic Year **2025-2026** onwards)

1. Award of the Degree

- (a) Award of the B.Tech. Degree / B.Tech. Degree with a Minor if he/she fulfills the following:
- (i) Pursues a course of study for not less than three academic years and not more than six academic years. However, for the students availing Gap year facility this period shall be extended by two years at the most and these two years would in addition to the maximum period permitted for graduation (Six years).
 - (ii) Registers for 120 credits and secures all 120 credits.

(b) Award of B.Tech. Degree with Honors

A student will be declared eligible for the award of the B.Tech. with Honors if he / she fulfills the following:

- i) Student secures additional 15 credits fulfilling all the requisites of a B.Tech. Program i.e., 120 credits.
 - ii) Registering for Honors is optional.
 - iii) Honors are to be completed simultaneously with B.Tech programme.
- 2.** Students, who fail to fulfill the requirement for the award of the degree within six consecutive academic years from the year of admission, shall forfeit their seat.

3. Minimum Academic Requirements

The following academic requirements have to be satisfied in addition to the requirements mentioned in item no.2

- i. A student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory, practical, design, drawing subject or project if he secures not less than 35% of marks in the end examination and a minimum of 40% of marks in the sum total of the mid semester evaluation and end examination taken together.
- ii. A student shall be promoted from III year to IV year if he/she fulfills the academic requirements of securing 40% of the credits (any decimal fraction should be rounded off to lower digit) in the subjects that have been studied up to V semester.

And in case if student is already detained for want of credits for particular academic year, the student may make up the credits through supplementary exams of the above exams before the commencement of IV year I semester class work of next year.

4. Course Pattern

- i) The entire course of study is three academic years on semester pattern.
 - ii) A student eligible to appear for the end examination in a subject but absent at it or has failed in the end examination may appear for that subject at the next supplementary examination offered.
 - iii) When a student is detained due to lack of credits/shortage of attendance the student may be re-admitted when the semester is offered after fulfillment of academic regulations, the student shall be in the academic regulations into which he/she is readmitted.
5. All other regulations as applicable for B. Tech. Four-year degree course (Regular) will hold good for B. Tech. (Lateral Entry Scheme).

(Dr. R Prasad Rao)
Dean(Academics) &
Member Secretary (AC)

(Dr.C P V N J Mohan Rao)
Chairman
Academic Council



AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY **(Autonomous)**

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC Accredited with A+ grade

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113.

<https://avanthienggcollege.ac.in/>, E-mail: principal@avanthienggcollege.ac.in

Department of COMPUTER SCIENCE AND ENGINEERING

Proposed Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Induction Programme

S.No	Course Title	Category	L-T-P-C
1	Physical Activities--Sports, Yoga and Meditation, Plantation	MC	0-0-6-0
2	Career Counseling	MC	2-0-2-0
3	Orientation to all branches -- career options, tools, etc	MC	3-0-0-0
4	Orientation on admitted Branch -- corresponding labs, tools and platforms	EC	2-0-3-0
5	Proficiency Modules & Productivity Tools	ES	2-1-2-0
6	Assessment on basic aptitude and mathematical skills	MC	2-0-3-0
7	Remedial Training in Foundation Courses	MC	2-1-2-0
8	Human Values & Professional Ethics	MC	3-0-0-0
9	Communication Skills -- focus on Listening, Speaking, Reading, Writing skills	BS	2-1-2-0
10	Concepts of Programming	ES	2-0-2-0

Department of COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

I Year I Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				Lecture	Tutorial	Practical	Credits
1	BS	R24BS01	Linear Algebra and Calculus	3	0	0	3
2	HS	R24HS01	Communicative English	2	0	0	2
3	BS	R24BS02	Engineering Physics	3	0	0	3
4	ES	R24ES01	Basic Civil & Mechanical Engineering	3	0	0	3
5	ES	R24ES02	Problem Solving and Programming with C	3	0	0	3
6	BS	R24BS03	Engineering Physics Lab	0	0	2	1
7	HS	R24HS02	Communicative English Lab	0	0	2	1
8	ES	R24ES03	Problem Solving and Programming with C Lab	0	0	3	1.5
9	ES	R24ES04	IT Workshop	0	0	2	1
10	MC	R24MC01	Health and Wellness, Yoga, and Sports	0	0	1	0.5
Total				14	00	10	19

Category	Courses	Credits
BS-Basic Science Course	3	7
ES-Engineering Science Course	4	8.5
HS-Humanities and Social Science including Management Course	2	3
MC-Mandatory Course	1	0.5
Total	10	19

Department of COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

I Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	BS	R24BS04	Differential Equations and Vector Calculus	3	0	0	3
2	BS	R24BS05	Applied Chemistry	3	0	0	3
3	ES	R24ES06	Engineering Graphics	1	0	4	3
4	ES	R24ES05	Basic Electrical and Electronics Engineering	3	0	0	3
5	PC	R24CSPC01	Data Structures through C	3	0	0	3
6	BS	R24BS06	Applied Chemistry Lab	0	0	2	1
7	ES	R24ES07	Basic Electrical and Electronics Engineering Lab	0	0	3	1.5
8	PC	R24CSPC02	Data Structures through C Lab	0	0	3	1.5
9	ES	R24ES08	Engineering Workshop	0	0	3	1.5
10	MC	R24MC02	NSS/NCC/Scouts & Guides/Community Service	0	0	1	0.5
Total				13	00	16	21

Category	Courses	Credits
BS- Basic Science Course	3	7
ES-Engineering Science Course	4	9
PC-Professional Core Course	2	4.5
MC-Mandatory Course	1	0.5
Total	10	21

**Chairperson
Board of Studies (CSE)**

Linear Algebra and Calculus
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: Linear Algebra and Calculus	Course Code: R24BS01
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: Linear algebra is a prerequisite for calculus, and that you should have a deep understanding of linear algebra before moving on to calculus.	

COURSE OVERVIEW:

A course on linear algebra and calculus typically covers fundamental concepts like vectors, matrices, linear systems, differentiation, and integration.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To equip the students with standard concepts and tools of mathematics to handle various real-world problems and their applications.
2. To enable the students to apply linear algebra to solve engineering problems.
3. To enable the students to apply calculus to solve engineering problems.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Develop matrix algebra techniques that are needed by engineers for practical applications.
CO2	To find the eigen values and eigen vectors and solve the problems by using linear transformation.
CO3	Apply the knowledge of mean value theorems, solve inequality.
CO4	Familiarize with functions of several variables which is useful in optimization.
CO5	Familiarize with double and triple integrals of functions of several variables in two and three dimensions.

COURSE CONTENT (SYLLABUS)**UNIT-I: Matrices and Linear System of Equations****10 Hours**

Matrices: Vector Space, Linear independent, dependent (only definitions).

Rank of a matrix by echelon form, normal form. Cauchy-Binet formulae (without proof). Inverse of Non-singular matrices by Gauss- Jordan method.

System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations by Gauss elimination method.

COs-CO1

Self-Learning Topic: Encoding and Decoding messages by using matrices

UNIT- II: Linear Transformation and Orthogonal Transformation **10 Hours**

Eigen values and Eigen vectors and their properties(without proof), Diagonalization of a matrix, Cayley-Hamilton Theorem (without proof), finding inverse and power of a matrix by Cayley–Hamilton Theorem, Quadratic forms and Nature of the Quadratic Forms, Reduction of Quadratic form to canonical forms by Orthogonal Transformation.

COs-CO2

Self-Learning Topic: Google’s page rank Algorithm.

UNIT–III: Calculus **10 Hours**

Mean Value Theorems: Rolle’s Theorem, Lagrange’s mean value theorem with their geometrical interpretation, Cauchy’s mean value theorem, Taylor’s and Maclaurin theorems with remainders (without proof), problems on the above theorems.

COs-CO3

Self-Learning Topic: Application of mean value theorems

UNIT- IV: Partial differentiation and Applications **10 Hours**

Partial derivatives, total derivatives, chain rule, change of variables, Taylor’s and Maclaurin’s series expansion of functions of two variables. Jacobian, maxima and minima of functions of two variables, method of Lagrange multipliers.

COs-CO4

Self-Learning Topic: Jacobian of implicit functions.

UNIT-V: Multiple Integrals **10 Hours**

Double integrals - change of variables (Cartesian and Polar coordinates), change of order of integration, Cylindrical and Spherical coordinates, triple integrals. Finding areas (by double integrals) and volumes (by double integrals and triple integrals).

COs-CO5

Self-Learning topic: Calculating Centers of Mass and Moment of inertia

Text Books:

1. B.S.Grewal, Higher Engineering Mathematics, 44/e, Khanna publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.

Reference Books:

1. Dennis G. Zill and Warren S. Wright , Advanced Engineering Mathematics, Jones and Bartlett, 2018.
2. Michael Green berg, Advanced Engineering Mathematics, 9th edition, Pearson edn.
3. George B. Thomas, Maurice D. Weir and Joel Hass, Thomas Calculus, 14/e, Pearson Publishers, 2018.
4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5/e, Alpha Science
5. International Ltd., 2021 (9th reprint).
6. B.V. Ramana, Higher Engineering Mathematics, McGraw Hill Education, 2017.

Web References:

1. <http://onlinecourses.nptel.ac.in>
2. <https://nptel.ac.in/courses/111105121>
3. https://onlinecourses.nptel.ac.in/noc24_ma91/course
4. https://onlinecourses.nptel.ac.in/noc24_ma53/course
5. https://onlinecourses.nptel.ac.in/noc24_ma11/course

COMMUNICATIVE ENGLISH
I B.TECH- I SEMESTER (Common to CSE,CSE(AIML),CSE(DS))

Course Title: COMMUNICATIVE ENGLISH	Course Code: R24HS01
Teaching Scheme (L:T:P): 2 0 0	Credits: 2
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: To excel in a <i>Communicative English</i> course, certain foundational skills and prerequisites are helpful such as Basic Grammar Knowledge, Listening Skills , Basic Vocabulary, Reading Comprehension, Confidence in Speaking, Writing Skills.	

COURSE OVERVIEW:

A *Communicative English* course is designed to develop students' proficiency in spoken and written English through practical and interactive learning methods. The course focuses on improving students' ability to communicate effectively in real-life situations, emphasizing both fluency and accuracy.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To identify the English Communication Skills among the first year B.Tech students and to initiate measures to bridge the gap.
2. To enlighten the students on the necessity of cultivating good language habits through practising LSRW skills.
3. To explain them various topics of grammar and the importance of being grammatically correct in speech and writing.
4. To make them practise Phonetics and impart the nuances of fine speech.
5. To instruct them about the various types of format related to writing letters, paragraph, emails, essays and reports.
6. To make them appreciate English text and deepen their comprehension through reading of textual and non-detailed topics.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	To utilize the text, online resources, and other social, and real time situations with an aim to practice Communicative English
CO2	To apply grammatical knowledge for speaking, and writing purposes
CO3	To analyze and practice various devices of speech for effective conversation and presentations
CO4	Appraising the language competence of the learners and suggesting remedial action
CO5	To make the learners practice writing tasks which are relevant for job training and academic purposes.

COURSE CONTENT (SYLLABUS)**UNIT-I** **12 Hours****Lesson: HUMAN VALUES: A Power of a Plate of Rice by Ifeoma Okoye (Short story)****Listening:** Identifying the topic, the context and specific pieces of information

By listening to short audio texts and answering a series of questions.

Speaking: Asking and answering general questions on familiar topics such as home, family, work, studies and interests; introducing oneself and others.**Reading:** Skimming to get the main idea of a text; scanning to look for specific pieces of information.**Writing:** Mechanics of Writing-Capitalization, Spellings, Punctuation-Parts of Sentences.**Grammar:** Parts of Speech, Basic Sentence Structures-forming questions**Vocabulary:** Synonyms, Antonyms, Affixes (Prefixes/Suffixes), Root words. **COs-C01**

Self learning topics : The Great Indian Scientists-Biography of CV Raman

UNIT-II **10 Hours****Lesson: NATURE: Night of the Scorpion by Nissim Ezekiel (Indian and contemporary)****Listening:** Answering a series of questions about main ideas and supporting ideas after listening to audio texts.**Speaking:** Discussion in pairs/small groups on specific topics followed by short structure talks.**Reading:** Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together.

Writing: Structure of a paragraph - Paragraph writing (specific topics)

Grammar: Cohesive devices-linkers, use of articles and zero article prepositions.**Vocabulary:** Homonyms, Homophones, Homographs. **COs-C02**

Self learning Topics : Seven Ages of Man by William Shakespeare.

UNIT-III **12 Hours****Lesson: BIOGRAPHY: Steve Jobs****Listening:** Listening for global comprehension and summarizing what is listened to.**Speaking:** Discussing specific topics in pairs or small groups and reporting what is discussed**Reading:** Reading a text in detail by making basic inferences-recognizing and interpreting specific context clues; strategies to use text clues for comprehension.**Writing:** Summarizing, Note-making, paraphrasing**Grammar:** Verbs-tenses; Subject-verb agreement; Compound words, Collocations**Vocabulary:** Compound words, Collocations **COs-C03**

Self learning topics: Elon Musk

UNIT-IV **8 Hours****Lesson: INSPIRATION: The Knowledge Society by APJ Abdul Kalam (Ignited minds)****Listening:** Making predictions while listening to conversations/ transactional dialogues without video; listening with video.**Speaking:** Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions.**Reading:** Studying the use of graphic elements in texts to convey information, reveal trends/patterns / relationships, communicate processes or display complicated data.

Writing: Letter Writing: Official Letters and Resumes

Grammar: Reporting verbs, Direct & Indirect speech, Active & Passive Voice

Vocabulary: Words often confused, Jargons

COs-C04

Self learning Topics: The writings of Sudha Murthy- “The day I stopped drinking milk”

UNIT-V

10 Hours

Lesson: MOTIVATION: The Power of Intra personal Communication (An Essay)

Listening: Identifying key terms, understanding concepts and answering a series of relevant questions that test comprehension.

Speaking: Formal Oral Presentation topics from academic contexts

Reading: Reading comprehension.

Writing: Writings structured essays on specific topics.

Grammar: Editing short texts –identifying and correcting common errors in grammar and usage (articles, prepositions, tenses, subject-verb agreement)

Vocabulary: Technical Jargons

COs-C05

Self learning Topics: Body Language (Allan Pease)

Textbooks:

1. Pathfinder: Communicative English for Undergraduate Students, 1stEdition, Orient BlackSwan, 2023 (Units 1,2,3 &5)
2. Empowering English by Cengage Publications, 2023
3. The Great Indian Scientists-Cengage Publications
4. English Essentials- Maruthi Publications.(Unit 4)

Reference Books:

1. P. Elian : A Hand book of English for Engineers and Technologists,
2. Bailey, Stephen. Academic writing: A Handbook for International Students. Routledge, 2014.
3. Murphy Raymond English Grammar in Use, Fourth Edition, Cambridge University Press, 2019.
4. English for Engineers by Shyam Ji Dubey- Vikas Publishing House

Web References:

1. www.bbc.co.uk/learningenglish
2. <https://dictionary.cambridge.org/grammar/british-grammar/>
3. www.eslpod.com/index.html
4. <https://www.learngrammar.net/>
5. <https://english4today.com/english-grammar-online-with-quizzes/>
6. <https://www.talkenglish.com/grammar/grammar.aspx>
7. <https://www.youtube.com/c/DailyVideoVocabulary/videos>
8. https://www.youtube.com/channel/UC4cmBAit8i_NJZE8qK8sfpA

ENGINEERING PHYSICS**I B.TECH- I SEMESTER (Common to CSE,CSE(AI&ML),CSE(DS))**

Course Title: ENGINEERING PHYSICS	Course Code: R24BS02
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: For <i>Engineering Physics</i> , which combines principles of physics with engineering applications, having a solid foundation in certain areas is crucial for understanding advanced concepts. Here's an outline of key prerequisites that can help students succeed in the course.	

COURSE OVERVIEW:

An *Engineering Physics* course is typically designed to bridge the gap between theoretical physics principles and engineering applications, providing students with a solid foundation to analyze and solve complex engineering problems.

COURSE OBJECTIVES:

The objectives of this course are to

1. bridge the gap between the physics in school at 10+2 level and UG level engineering courses.
2. identify the importance of the optical phenomenon i.e. interference and diffraction related to its engineering applications.
3. understand the mechanism of emission of light, utilization of lasers as coherent light sources for low and high energy applications.
4. enlightening the periodic arrangement of atoms in crystalline solids and classify various crystal systems.
5. explain the significant concepts of dielectric and magnetic materials that leads to potential applications in the emerging micro devices.
6. enlightenment of the concepts of quantum mechanics and to provide fundamentals of de-Broglie matter waves and the importance of free electron theory for metals.
7. understand the physics of semiconductors and identify the type of semiconductor using Hall effect.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Analyze the intensity variation of light due to interference, diffraction and classify various types of lasers.
CO2	Identify various crystal systems and analyze the crystalline structure.
CO3	Summarize various types of polarization of dielectrics and classify the magnetic materials.
CO4	Explain fundamentals of quantum mechanics and apply to one dimensional motion of particles.
CO5	Outline the properties of charge carriers in semiconductors

COURSE CONTENT (SYLLABUS)**UNIT-I Wave Optics****12 Hours**

Interference: Introduction - Principle of superposition –Interference of light - Interference in thin films (Reflection Geometry) & applications - Colors in thin films- Newton’s Rings- Determination of wavelength and refractive index

Diffraction: Introduction - Fresnel and Fraunhofer diffractions - Fraunhofer diffraction due to single slit-Fraunhofer diffraction due to N Slits -Diffraction Grating - Dispersive power and resolving power of Grating (Qualitative).

Lasers: Introduction–Characteristics of laser – Spontaneous and Stimulated emissions of radiation – Population inversion – Lasing action - Pumping mechanisms – Ruby laser –He-Ne laser-Applications of lasers.

COs – CO1

Self-Learning Topics: Interference in thin films due to Transmission of light

UNIT-II Crystallography and X-ray diffraction**12 Hours**

Crystallography: Space lattice, Basis, Unit Cell and lattice parameters – Bravais Lattices – crystal systems (3D) – coordination number - packing fraction of SC, BCC & FCC - Miller indices – separation between successive (hkl) planes.

X-ray diffraction: Bragg’s law - X-ray Diffractometer – crystal structure determination by Laue’s and powder methods.

COs– CO2

Self-Learning Topics: Effect of crystallite size on diffracted X-Ray intensity.

UNIT-III Magnetic and Dielectric Materials**12 Hours**

Magnetic Materials: Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability – Atomic origin of magnetism - Classification of magnetic materials: Dia, para, Ferro, anti-ferro& Ferri magnetic materials - Domain concept for Ferromagnetism & Domain walls (Qualitative) - Hysteresis - soft and hard magnetic materials.

Dielectric Materials: Introduction - Dielectric polarization - Dielectric polarizability, Susceptibility, Dielectric constant and Displacement Vector –Relation between the electric vectors - Types of polarizations- Electronic (Quantitative), Ionic (Quantitative) and Orientation polarizations (Qualitative) - Lorentz internal field - Clausius-Mossotti equation–dielectric loss.

COs– CO3

Self-Learning Topics: Frequency dependence of polarization.

UNIT-IV Quantum Mechanics and Free electron theory**12 Hours**

Quantum Mechanics: Dual nature of matter – Heisenberg’s Uncertainty Principle – Significance and properties of wave function – Schrodinger’s time independent and dependent wave equations– Particle in a one-dimensional infinite potential well.

Free Electron Theory: Classical free electron theory (Qualitative with discussion of merits and demerits) – Quantum free electron theory –electrical conductivity based on quantum free electron theory - Fermi-Dirac distribution - Fermi energy.

COs– CO4

Self-Learning Topics: Density of states, Origin of energy bands in solids

UNIT-V Semiconductors**8 Hours**

Semiconductors: Formation of energy bands – classification of crystalline solids - Intrinsic semiconductors:– Fermi level – Extrinsic semiconductors- P-Type semiconductors- N-Type semiconductors- principle of operation and characteristics of P-N Junction diode - Drift and diffusion currents –Einstein’s equation - Hall effect and its applications.

COs – CO5

Self-Learning Topics: Zener diode, Solar cells

TEXT BOOKS:

1. “A Textbook of Engineering Physics” by M.N. Avadhanulu, P.G.Kshirsagar-S.Chand Publications, 2017.
2. “Engineering Physics” by D. K. Bhattacharya and Poonam Tandon, Oxford Press (2015).
3. “Engineering Physics” by R.K.Gaur and S.L.Gupta., -Dhanpat Rai publishers, 2012.

REFERENCE BOOKS:

1. Engineering Physics - B.K. Pandey and S. Chaturvedi, Cengage Learning.
2. The Principles of Quantum Mechanics, P. A. M. Dirac, fourth Edition (Oxford University Press, Oxford, 1958).
3. Physics-Resnick, Halliday, Krane, Fifth edition, Volume-1, Wiley student edition.
4. Engineering Physics - Dr.R. Swapna, Scientific International Publishing House.
5. Concepts of Modern Physics. Arthur Beiser, Tata McGraw-Hill, New Delhi (2010).
6. Engineering Physics” - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press.
7. Engineering Physics - M.R. Srinivasan, New Age international publishers (2009).

ONLINE RESOURCES:

Web References:

1. <https://www.ebooksdirectory.com/>
2. <http://www.sciencedirect.com/Science>
3. <https://onlinecourses.nptel.ac.in/>
4. <https://www.link.springer.com/physics/>
5. <https://www.loc.gov/rr/scitech/selected-internet/physics.html>

E-BOOKS:

1. <https://www.ebooksdirectory.com/>

BASIC CIVIL AND MECHANICAL ENGINEERING
IB.TECH-I SEMESTER (Common to CSE, CSE(AIML, CSE(DS))

Course Title: Basic Civil and Mechanical Engineering	Course Code: R24ES01
Teaching Scheme(L:T:P): 3:0:0	Credits:3
Type of Course:Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: To excel in a <i>Basic Civil and Mechanical Engineering</i> course, students should have foundational knowledge and skills in both mathematics and physics. These prerequisites help students understand the principles that underlie civil and mechanical engineering applications.	

COURSE OVERVIEW:

- Basic Civil and Mechanical Engineering course provides a broad foundation for all engineering disciplines, equipping students with a holistic understanding of the principles, design methods, and innovations shaping each branch, fostering interdisciplinary knowledge and skills.
- This introductory course covers fundamental concepts in Civil and Mechanical Engineering, emphasizing their roles in society and diverse applications. In Civil Engineering, students explore various disciplines, including structural, transportation, water resources, and environmental engineering, along with essential materials and construction techniques, surveying, and advancements in sustainable practices.
- The Mechanical Engineering segment introduces students to core sectors such as energy, manufacturing, and automotive, along with basic design principles and engineering materials. Key topics include thermal engineering, power cycles, IC engines, and power plant operations, as well as principles in manufacturing, CNC, 3D printing, and robotics. This course provides a foundational understanding of both fields, preparing students for more specialized study and practical applications in engineering.

COURSE OBJECTIVES:

The objectives of this course are to

1. **Understand the Role of Civil Engineers:** Familiarize students with the roles and responsibilities of civil engineers in society and the various sub-disciplines within civil engineering.
2. **Construction Materials:** Provide knowledge about different construction materials such as cement, aggregates, bricks, concrete, steel, soil, stones and their applications in building construction.
3. **Transportation Engineering:** Offer insights into the importance of transportation engineering for national economic development and the fundamentals of highway pavements, harbor, tunnel airport and railway engineering.
4. **Water Resources and Environmental Engineering:** Cover the basics of water sources, water quality specifications, hydrology, rainwater harvesting, and water storage structures, emphasizing their importance in environmental sustainability.
5. **Scope and Importance of Mechanical Engineering:** Familiarize students with the scope and significance of mechanical engineering in various sectors, including energy, manufacturing, automotive, aerospace and marine industries.
6. **Engineering Materials and Manufacturing Processes:** Explain different engineering

materials and various manufacturing processes and computational manufacturing.

7. **Thermal Engineering:** Provide an overview of thermal engineering principles, including the working of boilers, IC engines, and power plants, and introduce concepts related to electric and hybrid vehicles.
8. **Mechanical Power Transmission Systems:** Describe different mechanical power transmission systems such as belt drives, chain drives, gear drives, and their applications.
9. **Basics of Robotics:** Introduce the basics of robotics, including joints, links, configurations, and applications, along with advancements in robotics technology.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Understand the role of civil engineers in various disciplines, the scope of each discipline, and the materials used in building construction and principles of surveying.
CO2	Describe the fundamentals of transportation engineering, water resources, and environmental engineering, including highway pavements, water quality, hydrology, and water storage structures.
CO3	Understand and apply different manufacturing processes and engineering materials, including their applications, and basic mechanical design principles.
CO4	Explain the basics of thermal engineering, including working principles of engines, power plants, and related thermal cycles, along with their applications.
CO5	Describe the working of different mechanical power transmission systems and the basics of robotics and their applications.

COURSE CONTENT (SYLLABUS)

UNIT-I:

Role of Civil Engineers in Society, Various Disciplines of Civil Engineering, Structural Engineering, Geo-technical Engineering, Transportation Engineering, Hydraulics and Water Resources Engineering, Environmental Engineering, Scope of Each Discipline, Building Construction and Planning, Construction Materials Cement, Aggregate, Bricks, Cement Concrete-Steel, soils and stones. Introduction to Prefabricated construction Techniques.

Surveying: Objectives of Surveying, Horizontal Measurements, Angular Measurements, Introduction to Bearings Simple problems on bearings-Contour mapping.

Self-Learning Topic: Advancements in Prefabricated Construction Techniques

UNIT-II:

Transportation Engineering: Importance of Transportation in Nation's economic development, Types of Highway Pavements, Flexible Pavements and Rigid Pavements, Simple Differences. Basics of Harbor, Tunnel, Airport, and Railway Engineering

Water Resources and Environmental Engineering: Introduction, Sources of water, Quality of water, Specifications, Introduction to Hydrology, Rainwater Harvesting, Water Storage and Conveyance Structures (Simple introduction to Dams and Reservoirs).

Self-Learning Topic: Sustainable Transportation Engineering

UNIT-III:

Introduction to Mechanical Engineering: Role of Mechanical Engineering in Industries and Society Technologies in different sectors such as Energy, Manufacturing, Automotive, Aerospace, and Marine sectors.

Basic Mechanical Design Principles: Fundamentals of Mechanical Design- Introduction to the design process, understanding design requirements, and conceptual design, Design of Simple Machine Components - Design considerations for basic machine components like shafts, bearings, gears, and fasteners.

Engineering Materials – Metals - Ferrous and Non-ferrous, Ceramics, Composites, Smart materials.

Self-Learning Topics: Sustainable Engineering Practices, Advancements in Smart Materials.

UNIT-IV:

Thermal Engineering–Working principle of Boilers

Cycles-Ottocycle, Diesycle, Refrigeration and air-conditioning cycles,

Engines-IC Engines, 2-Stroke and 4-Stroke engines, SI/CI Engines,

Power plants – Working principle of Steam, Diesel, Hydro, Nuclear power plants, Introduction to Electric and Hybrid Vehicles.

Self-Learning Topics: Advanced Engine Technologies, Thermodynamics in Renewable Energy Systems.

UNIT-V:

Manufacturing Processes: Principles of Casting, Forming, joining processes,

Computational Manufacturing: Introduction to CNC machines, 3D printing and Smart manufacturing.

Machining– Conventional & Non-Conventional,

Mechanical Power Transmission-Belt Drives, Chain, Rope drives, Gear Drives and their applications.

Introduction to Robotics-Joints & links, configurations, and applications of robotics.

Self-Learning Topics: Additive Manufacturing Technologies, Innovations in Mechanical Power Transmission

TEXTBOOKS:

1. Basic Civil and Mechanical Engineering, by Ommi Srikanth, M. Sreenivasa Reddy S. Chand Publications
2. Internal Combustion Engines by V. Ganesan, By Tata McGraw Hill publications (India) Pvt. Ltd.
3. A Text book of Theory of Machines by S.S. Rattan, Tata McGraw Hill Publications, (India) Pvt. Ltd.
4. An introduction to Mechanical Engg by Jonathan Wicker and Kemper Lewis, cengage learning India pvt. Ltd.

REFERENCE BOOKS:

1. Appu Kuttan KK, Robotics, I.K. International Publishing House Pvt. Ltd. Volume-I
2. 3D printing & Additive Manufacturing Technology- L. Jyothish Kumar, Pulak M Pandey,

Springer publications

3. Thermal Engineering by Mahesh M Rathore Tata McGraw Hill publications (India) Pvt. Ltd.
4. G. Shanmugam and M. S. Palanisamy, Basic Civil and the Mechanical Engineering, Tata McGraw Hill publications (India) Pvt. Ltd.

ONLINE RESOURCES:

1. https://www.youtube.com/playlist?list=PLyqSpQzTE6M_SM0Lrnzk2dJFwElh0Ebhu
2. <https://nptel.ac.in/courses/105101087>
3. <https://archive.nptel.ac.in/courses/105/105/105105110/>
4. <https://archive.nptel.ac.in/courses/112/105/112105125/>
5. <https://www.youtube.com/watch?v=-cr5vfV4YAI>
6. <https://nptel.ac.in/courses/112105266>
7. <https://archive.nptel.ac.in/courses/112/104/112104301/>

Problem Solving and Programming with C
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: Problem Solving and Programming with C	Course Code: R24ES02
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: for learning C programming, a strong background in problem-solving skills and an understanding of data structures and algorithms.	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To impart adequate knowledge on the need of programming languages and problem-solving techniques and develop programming skills.
2. To express algorithms and draw flowcharts in a language independent manner.
3. To enable effective usage of Operators & Control Structures.
4. To learn about the design concept of Arrays, Strings and Functions.
5. To understand Structures and Unions and their usage.
6. To assimilate about Pointers, Dynamic Memory Allocation and know the significance of Pre-processors, perform operations on files.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Illustrate the fundamental concepts of computers and basic computer programming and problem-solving approach.
CO2	Understand the Control structures, Branching and Looping.
CO3	Make use of Arrays and Develop Programs on modular programming using functions and strings.
CO4	Demonstrate the ability to write programs using Structures and Unions.
CO5	Apply File handling operations.

COURSE CONTENT (SYLLABUS)

UNIT-I: Introduction to Programming and Algorithm for Problem Solving: 10 Hours
Introduction to Programming: The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation,

Algorithm for Problem Solving: Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, Reversing digits of an integer, GCD (Greatest Common Division) of two numbers, Test whether a number is prime, Organize numbers in ascending order, Find square root of a number, factorial computation, Fibonacci sequence, Evaluate 'sin x' as sum of a series, Reverse order of elements of an array, Find largest number in an array, Print elements of upper triangular matrix, multiplication of two matrices, Evaluate a Polynomial. **COs–CO1**

Self-Learning Topics: Compilation and Interpretation

UNIT- II: Introduction to the 'C' Programming

15 Hours

Introduction: Character set, Variables and Identifiers, Built-in Data Types, Input/output statements, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Type Casting and Type def Simple 'C' programs. Storage Classes: Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in multiple source files: extern and static.

Conditional Statements and Loops: Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, Break statement, Go to statement. **COs-CO2**

Self-Learning Topics: Escape Sequences

UNIT – III: Arrays

15 Hours

Arrays: One dimensional array: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; two dimensional arrays with examples.

Strings: Concepts, String Types, String Input / Output functions, String manipulation functions, Null terminated strings as array of characters, **COs–CO3**

Self-Learning Topics: String Pattern Matching

UNIT- IV: Functions&Pointers

15 Hours

Functions: Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, passing arguments to a Function: call by reference; call by value, Recursive Functions, arrays as function arguments, Standard library string functions..

Pointers: Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation. **COs– CO4**

Self-Learning Topics: How do you pass a structure to a function?

UNIT-V: Structures and Unions

10 Hours

Structures and Unions: Structure variables, initialization, structure assignment, nested structure, structures and functions, structures, and arrays: arrays of structures, structures containing arrays, unions, Enumeration.

File Processing: Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input/output functions (standard library input/output functions for files), file status functions (error handling), Positioning functions **Cos-CO5**

Self-Learning Topics: Binary Files and operations on Binary files

TEXT BOOKS:

1. Byron S Gottfried “Programming with C” Second edition, Tata McGrawhill, 2007 (Paperback)
2. R.G. Dromey, “How to solve it by Computer”, Pearson Education, 2008.
3. Kanetkar Y, “Let us C”, BPB Publications, 2007.
4. Hanly J R & Koffman E.B, “Problem Solving and Program design in C”, Pearson Education, 2009.

REFERENCE BOOKS:

1. E. Balaguruswamy, “Programming with ANSI-C”, Fourth Edition, 2008, Tata McGraw Hill.
2. Venugopal K. R and Prasad S. R, “Mastering ‘C’”, Third Edition, 2008, Tata McGraw Hill.
3. B.W. Kernighan & D. M. Ritchie, “The C Programming Language”, Second Edition, 2001, Pearson Education
4. ISRD Group, “Programming and Problem-solving Using C”, Tata McGraw Hill, 2008.
5. Pradip Dey, Manas Ghosh, “Programming in C”, Oxford University Press, 2007.

Web References:

1. <http://www.c4learn.com/>
2. <http://www.geeksforgeeks.org/c/>
3. <http://nptel.ac.in/courses/122104019/>
4. <http://www.learn-c.org/>
5. https://www.tutorialspoint.com/c_programming/

ENGINEERING PHYSICS LAB**I B.TECH- I SEMESTER (Common to CSE,CSE(AI&ML),CSE(DS))**

Course Title: ENGINEERING PHYSICS LAB	Course Code: R24BS03
Teaching Scheme (L:T:P): 0:0:2	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: The <i>Engineering Physics Lab</i> is designed to provide students with hands-on experience and practical knowledge of physics concepts applied in engineering. Through experiments and observations, students can bridge theoretical physics principles with real-world applications, enhancing their understanding of experimental physics and analytical skills.	

COURSE OVERVIEW:

To succeed in an *Engineering Physics Lab* course, certain foundational skills and knowledge are necessary for effective participation and understanding. Here are the key prerequisites:

1. Basic Physics Knowledge
2. Mathematics Skills
3. Measurement and Unit Conversions
4. Basic Laboratory Skills
5. Problem-Solving and Analytical Skills
6. Familiarity with Safety Practices
7. Basic Computing Skills

COURSE OBJECTIVES:

1. To study the concepts of optical phenomenon like interference, diffraction etc.,
2. To recognize the importance of energy gap in the study of conductivity and Hall effect in semiconductors
3. To study the parameters and applications of dielectric and magnetic materials by conducting experiments.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Demonstrate the modern engineering physics Techniques and tools in real times applications in engineering studies.
CO2	Develop the laboratory skills in handling of electrical and optical instruments.
CO3	Conduct experiment Independently and In team to record the measurements
CO4	Compare the experimental results with standard values and estimate errors

COURSE CONTENT (SYLLABUS)**List of Experiments**

1. Determination of radius of curvature of a given plano convex lens by Newton's ring's method.
2. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.

3. Determination of thickness of thin object by air wedge method
4. Determination of wavelength of Laser Source by diffraction grating.
5. Determination of rigidity modulus of the material of the given wire using Torsional pendulum.
6. Magnetic field along the axis of a current carrying circular coil by Stewart & Gee's Method.
7. Determination of dispersive power of the prism.
8. Determination of acceleration due to gravity and radius of Gyration by using Compound pendulum.
9. Determination of energy gap of a semiconductor using p-n junction diode.
10. Determination of dielectric constant using charging and discharging method.
11. Sonometer: Verification of laws of stretched string.
12. Estimation of Planck's constant using photoelectric effect.
13. Study the variation of B versus H by magnetization the magnetic material (B-H curve)
14. Determination of frequency of electrically maintained tuning fork by Melde's experiment.
15. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall effect.
16. Determination of the resistivity of semiconductor by four probe method.
17. Determination of young's modulus for the given material of wooden scale by non- uniform bending (or double cantilever) method .
18. Determination of magnetic susceptibility by Kundt's tube method

REFERENCE BOOKS:

1. S. Balasubramanian, M.N.Srinivasan "A Text Book of Practical Physics"-S ChandPublishers,2017.
2. J.Raja Gopalam Patnaik, "Physics Laboratory Manual for Undergraduate Students "Paramount Book Distributors 2023.

ONLINE RESOURCES:**Web References:**

1. <https://phet.colorado.edu/en/simulations/filter?subjects=physics&type=html,prototype>

COMMUNICATIVE ENGLISH LAB
I B.TECH- I SEMESTER (Common to CSE, CSE (AIML), CSE (DS))

Course Title: COMMUNICATIVE ENGLISH LAB	Course Code: R24HS02
Teaching Scheme (L:T:P): 0 0 2	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: To excel in a <i>Communicative English</i> course, certain foundational skills and prerequisites are helpful such as Basic Grammar Knowledge, Listening Skills , Basic Vocabulary, Reading Comprehension, Confidence in Speaking, Writing Skills.	

COURSE OVERVIEW:

A *Communicative English* course is designed to develop students' proficiency in spoken and written English through practical and interactive learning methods. The course focuses on improving students' ability to communicate effectively in real-life situations, emphasizing both fluency and accuracy.

COURSE OBJECTIVES:

The objectives of this course are to:

The main objective of introducing this course, Communicative English Laboratory, is to expose the students to a variety of self-instructional, learner friendly modes of language learning. Students undergo training in basic communication skills to make them into confident communicators in all situations.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Understand and recognize the various facets of English language ability with a focus on the four basic skills- namely -LSRW abilities.
CO2	Implement various activities for language learners to practise communication skills.
CO3	To enhance listening and speaking comprehension, analyze the sounds, stress, rhythm, intonation, and syllable division of English speech.
CO4	Assess the professionalism of students when taking part in group discussions, debates, JAM sessions, Presentations and Interviews.
CO5	Equipping oneself with Interview Skills and a range of Soft Skills for life and career.

COURSE CONTENT (SYLLABUS)**Week1:**

1. To explain and guide the students in decoding the sounds of English.
2. List all the consonant sounds and vowel sounds in English

Week2:

1. What is a syllable and describe the syllable structure.
2. Define stress, functional stress and various rules of stress.

3. What is connected speech?

Week3:

1. What is Intonation and mention the various pitch movements like rise, fall, fall-rise or rise-fall?
2. What is connected speech?

Week4:

1. To equip students to speak in English language confidently without any inhibitions.
2. Why are majority of the companies conducting JAM session as a preliminary interview?
3. What are the key skills tested in JAM round?

Week5:

1. To help students learn and understand different functions of language like greeting, asking
2. For information, giving information, meetings, requests, exchanging dialogues in formal and informal contexts.
3. Introduce yourself and others, give instructions and directions

Week 6:

1. To help the students understand and work on the digital age connector for personal correspondence, business communication, etc.
2. Write about email etiquette.
3. Draft an email to the HR Manager of Wipro Technologies requesting to consider your application for the post of Software Engineer.

Week 7:

1. To update students about the importance of Resume, the various types and the essentials of an effective resume
2. Draft a resume for a software post in reputed organization.

Week 8:

1. To educate students about the various styles of writing formal letters.
2. What is a cover letter? What are the different types of cover letters?
3. Write a job application letter for any post of your choice in a reputed company?

Week 9:

1. To help students know the importance of an SOP in their professional advancements?
2. What is an SOP and what are the different kinds and parts of an SOP?
3. Prepare an SOP to apply for a Master's Programme in any University of your choice.

Week 10:

1. To educate and guide the students about presentation skills and its importance in the technical evolving world.
2. To inform explain students about the importance of body language in various personal and professional forums
3. To help students to present papers, PPT's in seminars, workshops, conferences, research projects, interviews, etc.

Week 11:

1. To help students to give effective PPT's in various academic and professional platforms.
2. Describe various aspects that make PPT more effective.

3. Make a PPT on any topic of your choice and present it to the class.

Week 12:

1. To foster, creative, critical thinking skills, analytical skills and problem solving skills.
2. Suggest a few tips for preparing a poster.
3. Prepare posters from or outside your curriculum.

List of Activities:

- | | |
|--|--------------|
| 1. Sounds of English (Vowels and Consonants) | COs: CO1,CO2 |
| 2. Neutralization and Accent Rules | COs: CO1,CO2 |
| 3. Improving communication skills /JAM. | COs: CO3,CO4 |
| 4. Letter Writing and E-mail Writing | COs: CO1,CO2 |
| 5. Cover letters and Resume Writing | COs: CO1,CO2 |
| 6. Statement of Purpose. | COs: CO1,CO2 |
| 7. Debates | COs: CO4:CO5 |
| 8. Presentation skills- PPT and Poster | COs: CO4:CO5 |
| 9. Group Discussions , types and practice | COs: CO4,CO5 |
| 10. Interview skills – Mock interviews | COs: CO4,CO5 |

Reference Books:

1. Prof. M. Hari Prasad, Prof. Vijaya Babu, Prof. Padmaja Kalapala, Skill Craft – A Communicative English Laboratory Workbook, Maruthi Publications first Edition, 2023
2. Meenakshi Ramana, Sangeeta-Sharma, 4th Edition, Technical Communication, Oxford Press, 2022.
3. Grant Taylor: English Conversation Practice, 1st Edition, Tata ,Mc Graw-Hill Education India, 2001.
4. Hewing,s, Martin, Cambridge Academic English(B2), Cambridge University Press,2012.
5. T. Balasubramanyam, A Textbook of English Phonetics for Indian Students, 3rd Edition, Trinity, 2022.
6. Dr. ShaliniSharma’s Body Language Your Success Mantra, S. Chand publications 2010.
7. Sunitha Mishra and C.Murali Krishna’s Communication Skills for Engineers Pearson Education Edition 2009.

Suggested software:

- English Wordsworth –Language Lab- Wordsworth Software

Web References for:**Spoken English**

1. www.esl-lab.com
2. www.englishmedialab.com
3. www.englishinteractive.net
4. <https://www.britishcouncil.in/english/online>
5. <http://www.letstalkpodcast.com/>
6. . <https://www.youtube.com/c/ArnelsEverydayEnglish/featured>
7. <https://www.youtube.com/c/engvidAdam/featured>
8. . <https://www.youtube.com/c/EnglishClass101/featured>
9. <https://www.ted.com/watch/ted-ed>
10. <http://www.edest.org/>

Voice & Accent:

1. <https://www.youtube.com/user/letstalkaccent/videos>
2. <https://www.youtube.com/c/EngLanguageClub/featured>
3. https://www.youtube.com/channel/UC_OskgZBoS4dAnVUgJVexc
4. https://www.youtube.com/channel/UCNfm92h83W2i2ijc5Xwp_IA

Problem Solving and Programming with C Lab
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: Problem Solving and Programming with C Lab	Course Code: R24ES03
Teaching Scheme (L:T:P): 0 0 3	Credits: 1.5
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: Problem Solving & Programming with C lab include: Understanding programming fundamentals , Writing C programs, Applying programming techniques, Using algorithms, Using pseudocode and flowcharts	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

The course aims to give students hands – on experience and train them on the concepts of the C-programming language.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Read, understand, and trace the execution of programs written in C language.
CO2	Select the right control structure for solving the problems .and demonstrate the application of arrays functions and strings
CO3	Develop Debug and Execute programs to demonstrate the applications of Pointers, Structures& Unions, and Files.

COURSE CONTENT (SYLLABUS)**Developing the following programs:**

Week 1:

1. Write a C program using printf() and Scanf(). COs:CO1
2. Write a C program on swapping of two nos. COs:CO1
3. Write a C program using arithmetic Expressions. COs:CO1

Week 2:

4. Simple interest calculation COs:CO2
5. Finding compound interest COs:CO2
6. Area of a triangle using heron's formulae COs:CO2
7. Distance travelled by an object COs:CO2

Week 3:

8. Find the maximum of three numbers using conditional operator COs:CO2

9. Take marks of 5 subjects in integers, and find the total, average in float	COs:CO2
10. Write a C program to shift/rotate using bit fields.	COs:CO2
11. Finding the square root of a given number	COs:CO2
12. Write a C program using if-else statement.	COs:CO2
Week 4:	
13. Write a C program to find the max and min of four numbers using if-else.	COs:CO2
14. Write a C program to generate electricity bill.	COs:CO2
15. Find the roots of the quadratic equation.	COs:CO2
16. Write a C program to find the given year is a leap year or not.	COs:CO2
17. Write a C program to simulate a calculator using switch case.	COs:CO2
Week 5:	
18. Find the factorial of given number using any loop.	COs:CO2
19. Find the given number is a prime or not.	COs:CO2
20. Compute sine and cos series.	COs:CO2
21. Checking a number palindrome.	COs:CO2
22. Construct a pyramid of numbers.	COs:CO2
Week 6:	
23. Write a C program on Linear Search.	COs:CO3
24. Find the min and max of a 1-D integer array.	COs:CO3
25. Perform linear search on 1D array.	COs:CO3
26. The reverse of a 1D integer array.	COs:CO3
Week 7:	
27. Find 2's complement of the given binary number.	COs:CO3
28. Eliminate duplicate elements in an array.	COs:CO3
29. Sort array elements using bubble sort.	COs:CO3
30. Addition of two matrices.	COs:CO3
Week 8:	
31. Multiplication two matrices.	COs:CO3
32. Write a C program using call by reference.	COs:CO3
33. Write a C program to find factorial of n using recursion.	COs:CO3
34. Write a C function to calculate NCR value	COs:CO3
35. Concatenate two strings without built-in functions.	COs:CO3
Week 9:	
36. Write a C function to transpose of a matrix.	COs:CO3
37. Write a C function to find the length of a string.	COs:CO3
38. Reverse a string using built-in and without built-in string functions.	COs:CO3
39. Write a C program to find the sum of a 1D array using malloc ().	COs:CO3
Week 10:	
40. Write a recursive function to find the lcm of two numbers.	COs:CO3
41. Write a recursive function to find the sum of series.	COs:CO3
42. Write a C program to swap two numbers using call by reference.	COs:CO3
43. Write a C program using Pointers, Structures and Unions.	COs:CO4
44. Write a C program to find the total, average of n students using structures.	COs:CO4
Week 11:	

45. Enter n students data using calloc() and display failed students list. COs:CO4
 46. Read student name and marks from the command line and display the student details along with the total. COs:CO4
 47. Write a C program to implement realloc(). COs:CO4
 48. Write a C program to copy one structure variable to another structure of the same type. COs: CO4

Week 12:

49. Demonstrate Dangling pointer problem using a C program. COs: CO4
 50. Write a C program to copy one string into another using pointer. COs: CO4
 51. Write a C program to find no of lowercase, uppercase, digits and other characters using pointers. COs: CO4

Week 13:

52. Write a C program using Files operations. COs:CO5
 a. Sum and average of 3 numbers
 b. Conversion of Fahrenheit to Celsius and vice versa.
 53. Write a C program to write and read text into a file. COs:CO5

Week 14:

54. Write a C program to write and read text into a binary file using fread() and fwrite() COs:CO5
 55. Copy the contents of one file to another file. COs:CO5
 56. Write a C program to merge two files into the third file using command-line arguments. COs:CO5

Week 15:

57. Find no. of lines, words and characters in a file. COs:CO5
 58. Write a C program to print last n characters of a given file. COs:CO5

Scenario Based Case Study:

1. **Objective:** To develop a simple utility program to calculate the area of geometric shapes.

Target Audience: Beginner programmers and students.

A basic utility program in C that can:

1. Calculate the area of a circle.
2. Calculate the area of a rectangle.
3. Calculate the area of a triangle.

The program should be easy to understand and serve as an educational tool for new programmers.

1. Program Design:

The utility program will be designed with a simple menu-driven interface allowing the user to select the shape for which they want to calculate the area. The program will then prompt the user to input the necessary dimensions and display the result.

This basic C program demonstrates fundamental programming concepts such as variables, functions, and control structures in a practical context. It provides a clear introduction to C

programming for beginners by solving a simple problem using these core concepts.

This case study outlines a straightforward approach to teaching and implementing basic C programming concepts effectively.

2. Students Marks Sum HackerRank Solution

You are given an array of integers, marks, denoting the marks scored by students in a class.

- The alternating elements marks₀, marks₂, marks₄ and so on denote the marks of boys.
- Similarly, marks₁, marks₃, marks₅ and so on denote the marks of girls.

The array name, marks, works as a pointer which stores the base address of that array. In other words, marks contains the address where marks₀ is stored in the memory.

3. Sorting Array of Strings HackerRank Solution

To sort a given array of strings into lexicographically increasing order or into an order in which the string with the lowest length appears first, a sorting function with a flag indicating the type of comparison strategy can be written. The disadvantage with doing so has to rewrite the function for every new comparison strategy.

A better implementation would be to write a sorting function that accepts a pointer to the function that compares each pair of strings. Doing this will mean only passing a pointer to the sorting function with every new comparison strategy.

Textbooks:

1. Ajay Mittal, Programming in C: A practical approach, Pearson.
2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw Hill

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, PrenticeHall of India
2. C Programming, A Problem-Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE

IT Workshop

I B.TECH- II SEMESTER (Common to CSE,CSE(AIML),CSE(DS))

Course Title: IT Workshop	Course Code: R24ES04
Teaching Scheme (L:T:P): 0 0 2	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: an IT workshop for engineering students: Familiarity with hand tools, equipment, and machines, Computer skills.	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To assemble and disassemble a computer.
2. To solve hardware and software problems.
3. To learn about Networking of computers and use Internet facility for Browsing and Searching.
4. To develop project documentation using MS word
5. To work with various productivity tools including Excel, PowerPoint.
6. To work with different online repositories such as GITHUB, AI CHATBOT.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Perform Hardware troubleshooting and Perform Hardware troubleshooting
CO2	Apply different way of hooking the PC on to the internet from home and Workplace.
CO3	Design word documents by learning word processing and Create presentations by using different styles and using AI Tools-Chat GPT and GITHUB

COURSE CONTENT (SYLLABUS)

PC Hardware & Software Installation

9 Hours

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the Block diagram of the CPU along with the configuration of each peripheral and submit it to your Instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab Instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab Instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. Lab instructor should verify the installation and follow it up with a Viva. **COs-CO1**

Internet & World Wide Web

6 Hours

Task1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally, students Should demonstrate to the instructor, how to access the websites and email. If there is no internet Connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN Proxy settings, bookmarks, search toolbars and pop-up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: Search Engines & Netiquette: Students should know what search engines are and how to Use the search engines. A few topics would be given to the students for which they need to search On Google. This should be demonstrated to the instructors by the student. **COs-CO2**

MS WORD

6 Hours

Task 1: Creating project abstract Features to be covered: -Formatting Styles, Inserting table, Bullets And Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 2: Creating a Newsletter: Features to be covered: - Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

EXCEL

6 Hours

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool Excel as a Spreadsheet tool; give the details of the four tasks and features that would be covered in Each. Using Excel – Accessing, overview of toolbars, saving excel files, using help and resources. **Task 1:** Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, and auto Fill, Formatting Text.

Task 2: Calculating GPA -. Features to be covered: - Cell Referencing, Formulae in excel – Average, std. deviation, Charts, Renaming and Inserting worksheets, hyper linking, Count Function

POWER POINT

6 Hours

Task 1: Students will be working on basic power point utilities and tools which help them create basic power point presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Task 2: Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides. **Cos-CO3**

AI TOOLS – Chat GPT

6 Hours

Task 1: Prompt Engineering: Experiment with different types of prompts to see how the model Responds. Try asking questions, starting conversations, or even providing incomplete sentences to See how the model completes them. Ex: Prompt: "You are a knowledgeable AI. Please answer the following question: What is the capital of France?"

Task 2: Creative Writing: Use the model as a writing assistant. Provide the beginning of a story or a description of a scene, and let the model generate the rest of the content. This can be a fun way to

Brainstorm creative ideas Ex: Prompt: "In a world where gravity suddenly stopped working, people started floating upwards. Write a story about how society adapted to this new reality."

Explore – GITHUB

6 Hours

Task 1: Students should understand GITHUB and should possess accounts in it.

Task 2: Students should explore different repositories available in GITHUB and student should Create his/ her own simple repositories.

Task 3: Students should take simple experiments /presentations and upload them in their GITHUB Account.

Task 4: Students should understand how GITHUB Enterprise Cloud is used and also explore the GIT and GIT HUB resources.

COs-CO3

Reference Books:

1. Comdex Information Technology course tool kit, Vikas Gupta, WILEY Dream tech, 2003
2. The Complete Computer upgrade and repair book, Cheryl A Schmidt, WILEY Dream tech, 2013, 3rd edition
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education, 2012, 2nd edition
4. PC Hardware - A Handbook, Kate J. Chase, PHI (Microsoft)
5. IT Essentials PC Hardware and Software Companion Guide, David Anfins on and Ken Quamme. – CISCO Press, Pearson Education, 3rd edition
6. IT Essentials PC Hardware and Software Labs and Study Guide, Patrick Regan– CISCO Press, Pearson Education, 3rd edition
7. "Microsoft Word 2021: A Beginner's Guide"by Steve Lambert.
8. "Excel 2021: A Comprehensive Guide"by Chris Benham.
9. "Microsoft PowerPoint 2021: A Beginner's Guide" by Steve Lambert
10. GITHUB Quick Start Tutorials

HEALTH AND WELLNESS, YOGA AND SPORTS
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: HEALTH AND WELLNESS, YOGA AND SPORTS	Course Code: R24MC01
Teaching Scheme (L:T:P): 0:0:1	Credits: 0.5
Type of Course: Practical	
Continuous Internal Evaluation: 100 Marks	Semester End Exam: 0 Marks
Pre requisites: Prerequisites for courses in Health and Wellness, Yoga, and Sports can vary by institution and program. However, here's a general outline of common prerequisites and recommended qualifications for these types of courses.	

COURSE OVERVIEW:

Here's a general course overview for programs in **Health and Wellness, Yoga, and Sports**. Each area may have specific courses and focuses depending on the institution, but this will provide a foundational understanding of what to expect.

Health and Wellness

- **Overview:** This program focuses on promoting overall health, wellness strategies, and preventive health measures. It often includes the study of physical, mental, and social well-being.
- **Core Courses:**
 - **Introduction to Health and Wellness:** Basics of health concepts, wellness promotion, and lifestyle choices.
 - **Nutrition and Health:** Understanding dietary needs, nutritional guidelines, and the role of nutrition in health.
 - **Mental Health and Wellness:** Exploring psychological well-being, stress management, and mental health issues.
 - **Exercise Physiology:** The study of how physical activity affects the body and mind.
 - **Health Education and Promotion:** Strategies for promoting health within communities and organizations.
- **Practical Experience:** Some programs may include internships or fieldwork in health settings, community organizations, or wellness programs.

2. Yoga

- **Overview:** Yoga programs typically cover the physical, mental, and spiritual aspects of yoga practice. They can be geared toward practitioners or those looking to teach.
- **Core Courses:**
 - **Yoga Philosophy:** Study of the history and philosophy of yoga, including key texts and principles.
 - **Asana Practice:** Detailed exploration of yoga postures (asanas), including alignment, modifications, and variations.
 - **Meditation and Pranayama:** Techniques for breath control and meditation practices.
 - **Anatomy for Yoga:** Understanding the human body in relation to yoga practice, focusing on anatomy and physiology.
 - **Teaching Methodology:** Instruction on how to teach yoga classes, including class planning, communication skills, and sequencing.
- **Practical Experience:** Teaching practice sessions, observation of experienced teachers, and peer teaching.

3. Sports

- **Overview:** Sports programs often encompass a broad understanding of physical education, sports science, coaching, and athletic training.
- **Core Courses:**
 - **Introduction to Sports Science:** Overview of the key principles in sports science, including biomechanics and exercise physiology.
 - **Coaching Principles:** Theories and techniques related to effective coaching and athlete development.
 - **Sport Psychology:** Understanding the mental aspects of sports performance and strategies for enhancing motivation and focus.
 - **Exercise and Sport Nutrition:** Nutrition principles specifically tailored for athletes and active individuals.
 - **Sport Management:** Insights into the business side of sports, including marketing, finance, and event management.

COURSE OBJECTIVES:

The objectives of this course are to:

The main objective of introducing this course is to make the students maintain their mental and physical wellness by balancing emotions in their life. It mainly enhances the essential traits required for the development of the personality.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Understand the importance of yoga and sports for Physical fitness and sound health.
CO2	Demonstrate an understanding of health-related fitness components.
CO3	Compare and contrast various activities that help enhance their health.
CO4	Assess current personal fitness levels.
CO5	Develop Positive Personality

COURSE CONTENT (SYLLABUS)

UNIT I

Concept of health and fitness, Nutrition and Balanced diet, basic concept of immunity Relationship between diet and fitness, Globalization and its impact on health, Body Mass Index (BMI) of all age groups.

Activities:

- i) Organizing health awareness programmes in community
- ii) Preparation of health profile
- iii) Preparation of chart for balance diet for all age groups

UNIT II

Concept of yoga, need for and importance of yoga, origin and history of yoga in Indian context, classification of yoga, Physiological effects of Asanas- Pranayama and meditation, stress management and yoga, Mental health and yoga practice.

Activities:

Yoga practices – Asana, Kriya, Mudra, Bandha, Dhyana, Surya Namaskar

UNIT III

Concept of Sports and fitness, importance, fitness components, history of sports, Ancient and Modern Olympics, Asian games and Commonwealth games.

Activities:

- i) Participation in one major game and one individual sport viz., Athletics, Volleyball, Basketball, Handball, Football, Badminton, Kabaddi, Kho-kho, Table tennis, Cricket etc.
- ii) Practicing general and specific warm up, aerobics
- iii) Practicing cardiorespiratory fitness, treadmill, run test, 9 min walk, skipping and running.

Reference Books:

1. Gordon Edlin, Eric Golanty. Health and Wellness, 14th Edn. Jones & Bartlett Learning, 2022
2. T.K.V.Desikachar. The Heart of Yoga: Developing a Personal Practice
3. Archie J.Bahm. Yoga Sutras of Patanjali, Jain Publishing Company, 1993
4. Wiseman, John Lofty, SAS Survival Handbook: The Ultimate Guide to Surviving Anywhere Third Edition, William Morrow Paperbacks, 2014
5. The Sports Rules Book/ Human Kinetics with Thomas Hanlon. -- 3rd ed. Human Kinetics, Inc.2014

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities of Health/Sports/Yoga.
2. Institutes must provide field/facility and offer the minimum of five choices of as many as Games/Sports.
3. Institutes are required to provide sports instructor / yoga teacher to mentor the students.

Differential Equations and Vector Calculus
IB.TECH-II SEMESTER (Common to all Branches)

Course Title: Differential Equations and Vector Calculus	Course Code: R24BS04
Teaching Scheme(L:T:P): 3:0:0	Credits:3
Type of Course: Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester EndExam: 70Marks
Pre requisites: To succeed in Differential Equations and Vector Calculus, you'll need a strong foundation in several key areas of mathematics. Here are the typical prerequisites: Calculus I (Single-variable Calculus), Calculus II (Single-variable Calculus, continuation), Calculus III (Multivariable Calculus), Linear Algebra.	

COURSE OVERVIEW:

This course is often taken after completing Calculus I, II, and III, and Linear Algebra. It combines methods and applications of differential equations with essential topics in vector calculus, as used in fields like physics, engineering, and applied mathematics.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To enlighten the learners in the concept of differential equations and multivariable calculus.
2. To furnish the learners with basic concept and techniques at plus two level to lead them in to advanced level by handling various real-world applications.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Solve the first order differential equations related to various engineering fields.
CO2	Model engineering problems as higher order differential equations and solve analytically.
CO3	Identify solution methods for partial differential equations that model physical processes.
CO4	Interpret the physical meaning of different operators such as gradient, curl and divergence.
CO5	Estimate the work done against a field, circulation and flux using vector calculus.

COURSE CONTENT (SYLLABUS)**UNIT- I: Differential equations of first order and first degree****10 Hours**

Formation of differential equations, order, degree, separation of variables (only Review). Linear differential equations-Bernoulli's equations-Exact equations and equations reducible to exact form.

Applications: Newton's Law of cooling – Law of natural growth and decay, Electrical circuits (RL and LC).

COs-CO1

Self-Learning Topic: Mixed tank problems

UNIT- II: Higher order Linear differential equations with Constant Coefficients **10 Hours**

Definitions, homogenous and non-homogenous, complimentary function, particular integral (e^{ax} , $\sin ax$, $\cos ax$, Polynomial in x , $e^{ax}V(x)$, $xV(x)$), general solution, Wronskian, method of variation of parameters.

Applications: L-C-R Circuit problems **COs-CO2** Self-Learning Topic: Simple Harmonic motion

UNIT-III: Partial Differential Equations **10 Hours**

Introduction and formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions, solution of first order linear equations using Lagrange's method. Homogenous Linear Partial differential equations with constant coefficients. **COs-CO3**

Self-Learning Topic: Method of Separation of Variables

UNIT- IV: Vector differentiation **10 Hours**

Vector, Scalar, dot product, cross product, unit vector, equation of a line passing through two points (Review only)

Scalar and vector point functions, vector operator del, del applies to scalar point function-Gradient, del applied to vector point function – Divergence and Curl, Vector Identities

Application: Scalar Potential **COs-CO4**

Self-Learning Topic: Equation of tangent plane and Normal plane.

UNIT-V: Vector integration **10 Hour**

Line integral – circulation – work done, surface integral-flux, Green's theorem in the plane (without proof), Stoke's theorem (without proof), volume integral, Divergence theorem (without proof).

COs-CO5

Self-Learning Topic: Application of above theorems.

Text Books:

1. B.S.Grewal, Higher Engineering Mathematics, 44/e, Khanna publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.

Reference Books:

1. Dennis G. Zill and Warren S. Wright, Advanced Engineering Mathematics, Jones and Bartlett, 2018.
2. Michael Green berg, Advanced Engineering Mathematics, 9th edition, Pearson edn
3. George B. Thomas, Maurice D. Weir and Joel Hass, Thomas Calculus, 14/e, Pearson Publishers, 2018.
4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5/e, Alpha Science
5. International Ltd., 2021 (9th reprint).
6. B.V. Ramana, Higher Engineering Mathematics, McGraw Hill Education, 2017.

Web References:

1. <http://onlinecourses.nptel.ac.in>
2. <https://nptel.ac.in/courses/111105121>
3. https://onlinecourses.nptel.ac.in/noc24_ma86/course

APPLIED CHEMISTRY
IB.TECH- II SEMESTER(Common to EEE, ECE, CSE, CSE(AIML), CSE(DS))

Course Title: APPLIED CHEMISTRY	Course Code: R24BS05
Teaching Scheme(L:T:P): 3 0 0	Credits:3
Type of Course :Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites:	

COURSEOVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSEOBJECTIVES:

The objectives of this course are to:

1. To familiarize Applied Chemistry and its application.
2. To train the students on the principles and applications of electrochemistry and polymers
3. To elucidate the Structure and bonding of molecules
4. To impart Basic concepts of Semiconductors
5. To introduce modern engineering materials
6. To introduce instrumental methods, chromatographic technique

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Describe Molecular orbital diagrams
CO2	Importance of Graphenes. Apply the principle of band diagrams in the application of Super conductors and semiconductors.
CO3	Compare the Materials of Construction for Battery and Electro Chemical Sensors.
CO4	Explain the Preparation, Properties and applications of thermos plastics and thermos setting plastics, Elastomers and conducting polymers.
CO5	Summarize the concepts of instrumental methods

COURSECONTENT(SYLLABUS)**UNIT- I:Structure and Bonding models****10 hours**

Fundamentals of Quantum mechanics, Schrodinger Wave equation, significance of Ψ and Ψ^2 , particle in one dimensional box, molecular orbital theory–bonding in homo- and hetero nuclear diatomic molecules – energy level diagrams of O₂ and CO, etc. π -molecular orbitals of butadiene and Benzene, calculation of bond order.

COs-CO1

Self-Learning Topics: Energy Level Diagrams of N₂ & CN Molecule.

UNIT- II: Series Modern Engineering materials**10 hours**

Semiconductors – Introduction, Classification semiconductor devices P-N junction diode as a rectifier and transistor, applications. Super conductors-Introduction basic concept, Classification,

applications.

Super capacitors: Introduction, Basic Concept-Classification-Applications.

Nanomaterial: Introduction, classification, properties and applications of Fullerenes, carbon nano tubes and Graphenes

COs-CO2

Self Learning Topics: Band Theory of Solids, Sol-Gel method.

TEXT BOOKS:

1. 'Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.
2. 'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.
3. 'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

Web References:

1. <http://www.hackerrank.com/domains/datastructures>
2. http://www.github.com/topics/data_structures_c
3. <http://nptel.ac.in/courses>
4. <http://www.cslibrary.stanford.edu>

Basic Electrical and Electronics Engineering
I B.TECH- II SEMESTER (Common to CSE, CSE(AIML), CSE(DS))

Course Title: Basic Electrical and Electronics Engineering	Course Code: R24ES05
Teaching Scheme(L:T:P): 3:0:0	Credits:3
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: Solid state physics, Linear algebra, calculus.	

COURSEOVERVIEW:

- This course introduces to the concepts and definitions of Ohms law, KCL, KVL, power and energy. By applying Kirchoff's current and voltage laws to circuits in order to determine voltage, current and power in branches of any circuits excited by DC voltages and current sources. Apply simplifying techniques to solve DC circuit problems using basic circuit theorems and structured methods like node voltage and mesh current analysis. This course also introduces the construction and operating principle of AC machines, DC machines, Generators and Transformers.
- This course explores the evolution of electronics, characteristics of PN junction and Zener diodes, and bipolar junction transistors in various configurations. It includes rectifiers, power supplies, and amplifiers, focusing on circuit diagrams and frequency responses and covers number systems, Boolean algebra, and logic gates, along with simple combinational circuits like adders. It also introduces sequential circuits, including flip-flops and counters, and concludes with a block diagram of an electronic instrumentation system.

COURSEOBJECTIVES:

The objectives of this course are to

1. To expose to the field of electrical & electronics engineering.
2. To understand the importance of electrical safety.
3. To teach the fundamentals of semiconductor devices and its applications.
4. To teach the working process and analysis of different rectifying and Amplifying Circuits.
5. To teach the fundamental principles and rules of digital electronic circuits like gates, Sequential and Combinational Circuits.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Understand the problem-solving concepts associated to AC and DC circuits
CO2	Remember the fundamental laws, construction and operation of AC and DC machines, instruments.
CO3	Understand different power generation mechanisms, Electricity billing concept and important safety measures related to electrical operations.
CO4	Understand the fundamental principles of electronic devices, analyzing the different rectifying and Amplifying Circuits.
CO5	Analyze and design different digital electronic circuits like gates, Sequential and Combinational Circuits and Understand the basic Electronic instrumentation system.

COURSE CONTENT (SYLLABUS)**Part A-BASIC ELECTRICAL ENGINEERING****UNIT-I: DC & AC Circuits**

DC Circuits: Electrical circuit elements (R, L and C), Ohm's Law and its limitations, KVL & KCL, series, parallel, series-parallel circuits, Super Position theorem, Simple numerical problems.

AC Circuits: A.C Fundamentals: Equation of AC Voltage and current, waveform, time period, frequency, amplitude, phase, phase difference, average value, RMS value, form factor, peak factor, Voltage and current relationship with phasor diagrams in R, L, and C circuits, Concept of Impedance, Active power, reactive power and apparent power, Concept of power factor (Simple Numerical problems).

COs – CO1

Self-Learning Topics: Source Transformation

UNIT-II: Machines and Measuring Instruments

Machines: Construction, principle and operation of (i) DC Motor, (ii) DC Generator, (iii) Single Phase Transformer, (iv) Three Phase Induction Motor and (v) Alternator, Applications of electrical machines.

Measuring Instruments: Construction and working principle of Permanent Magnet Moving Coil (PMMC), Moving Iron (MI) Instruments and Wheat Stone bridge.

COs – CO2

Self-Learning Topics: Magnetic materials.

UNIT-III: Energy Resources, Electricity Bill & Safety Measures

Energy Resources: Conventional and non-conventional energy resources; Layout and operation of various Power Generation systems: Hydel, Nuclear, Solar & Wind power generation.

Electricity bill: Power rating of house hold appliances including air conditioners, PCs, Laptops, Printers, etc. Definition of "unit" used for consumption of electrical energy, two-part electricity tariff, calculation of electricity bill for domestic consumers.

Equipment Safety Measures: Working principle of Fuse and Miniature circuit breaker (MCB), merits and demerits. Personal safety measures: Electric Shock, Earthing and its types, Safety Precautions to avoid shock.

COs– CO3

Self-Learning Topics: Different types of electrical tools.

Part B: ELECTRONICS ENGINEERING**UNIT-IV: Semiconductor Devices and Basic Electronic Circuits**

Introduction - Evolution of electronics – Vacuum tubes to nano electronics - Characteristics of PN Junction Diode — Breakdown Effects in diodes — Zener Diode and its Characteristics. Bipolar Junction Transistor — CB, CE, CC Configurations and Characteristics — Introduction to Small Signal CE configuration.

Rectifiers and power supplies: Block diagram description of a dc power supply, Half-Wave Rectifiers, Full-Wave Rectifiers, capacitor filter (no analysis). Amplifiers: Block diagram of Public Address system, Circuit diagram and working of common emitter (RC coupled) amplifier with its frequency response.

COs–CO4

Self-Learning Topics: Electronic components and characteristics,

Design Amplifier circuit at different R, C Values

UNIT-V: DIGITAL ELECTRONICS and INSTRUMENTATION

Overview of Number Systems, BCD codes, Excess-3 code, Gray code, Hamming code. Boolean Algebra, Basic Theorems and properties of Boolean Algebra, Truth Tables and Functionality of Logic Gates – NOT, OR, AND, NOR, NAND, XOR and XNOR. Simple combinational circuits – Half and Full Adders. Introduction to sequential circuits, Flipflops, Registers and counters (Elementary Treatment only), Electronic Instrumentation: Block diagram of an electronic instrumentation system

COs – CO5

Self-Learning Topics: Develop digital circuits using minimum no. of gates, design principles of electronic instruments.

TEXTBOOKS:

1. Basic Electrical Engineering, D.C. Kulshreshtha, Tata McGraw Hill, 2019, First Edition.
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A. Chakrabarti, Dhanpat Rai & Co, 2013.
3. Fundamentals of Electrical Engineering, Rajendra Prasad, PHI publishers, 2014, Third Edition
4. R.L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
5. R.P. Jain, Modern Digital Electronics, 4th Edition, Tata McGraw Hill, 2009

REFERENCE BOOKS :(Basic Electrical Engineering)

1. Basic Electrical Engineering, D.P. Kothari and I.J. Nagrath, McGraw Hill, 2019, Fourth Edition.
2. Principles of Power Systems, V.K. Mehtha, S. Chand Technical Publishers, 2020.
3. Basic Electrical Engineering, T. K. Nagsarkar and M. S. Sukhija, Oxford University Press, 2017.
4. Basic Electrical and Electronics Engineering, S. K. Bhattacharya, Pearson Publications, 2018, Second Edition.

REFERENCE BOOKS: Electronics Engineering

1. R.S. Sedha, A Textbook of Electronic Devices and Circuits, S. Chand & Co, 2010.
2. Santiram Kal, Basic Electronics - Devices, Circuits and IT Fundamentals, Prentice Hall, India, 2002.
3. R.T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson

ONLINE RESOURCES:

Web References: (Basic Electrical Engineering)

1. <https://nptel.ac.in/courses/108105053>
2. <https://nptel.ac.in/courses/108108076>

Web References: (Electronics Engineering)

1. <https://archive.nptel.ac.in/courses/108/101/108101091/>
2. https://www.tutorialspoint.com/basic_electronics/index.htm
3. https://www.tutorialspoint.com/digital_circuits/index.htm

E-BOOKS:

<https://www.pdfdrive.com/basic-electronics-for-scientists-and-engineers-e28939124.html>

Engineering Graphics

IB.TECH-IISEMESTER(Common to CSE, CSE(AIEM), CSE(DS))

Course Title : Engineering Graphics	Course Code : R24ES06
Teaching Scheme(L:T:P): 1:0:4	Credits:3
Type of Course: Lecture +Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Prerequisites:	

COURSEOVERVIEW:

- The Engineering Graphics and Drawing course provides essential skills in visualizing and representing three-dimensional objects on two-dimensional media. Through structured units, students learn fundamentals such as line work, lettering, dimensioning, and geometric construction. The syllabus includes constructing curves, understanding scales, and mastering orthographic and isometric projections, essential for accurately depicting objects in engineering design.
- Additionally, students gain experience with computer-aided drafting using AutoCAD, learning to create 2D and 3D drawings and perform basic transformations. This course provides a strong foundation for technical drawing, crucial for design, manufacturing, and communication in engineering.

COURSEOBJECTIVES:

The objectives of this course are to

1. Understand the fundamentals of engineering drawing, including lines, lettering, and dimensioning.
2. Develop skills in geometrical constructions, including regular polygons and curves.
3. Learn orthographic projection techniques, including projections of points, lines, and planes.
4. Understand how to project solids in simple positions and create sectional views.
5. Develop skills in converting isometric views to orthographic views and vice versa.
6. Apply computer-aided design (CAD) techniques using AutoCAD to create 2D and 3D drawings.
7. Understand the importance of reference planes and reference lines in orthographic projection.
8. Develop problem-solving skills in engineering drawing, including creating and interpreting drawings.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Understand the basics of Engineering Graphics to construct the polygon, curves, and scales.
CO2	Draw the orthographic projections of points and straight lines inclined to both the planes.
CO3	Draw the projections of planes in various conditions.
CO4	Draw the projections of regular solids, with its axis inclined to one plane and sections of solids.
CO5	Visualize the 3D isometric views from 2D orthographic views and vice versa along with basic introduction to CAD.

COURSE CONTENT (SYLLABUS)

UNIT-I:	12 Hours
Introduction: Lines, Lettering and Dimensioning, Geometrical Constructions and Constructing regular polygons by general method.	
Curves: construction of ellipse, parabola and hyperbola by general method, Normal and tangent to Curves.	
Scales: Plain scales, diagonal scales and vernier scales.	
	COs-C01
 UNIT-II:	 16 Hours
Orthographic Projections: Reference plane, importance of reference lines or Plane, Projections of a point situated in any one of the four quadrants.	
Projections of Straight Lines: Projections of straight lines parallel to both reference planes, perpendicular to one reference plane and parallel to other reference plane, inclined to one reference plane and parallel to the other reference plane. Projections of Straight Line Inclined to both the reference planes.	
	COs-C02
 UNIT-III:	 10 Hours
Projections of Planes: Regular planes Perpendicular to both reference planes, parallel to one reference plane and inclined to the other reference plane; plane inclined to both the reference planes.	
	COs-C03
 UNIT-IV:	 16 Hours
Projections of Solids: Types of solids: Polyhedra and Solids of revolution. Projections of solids in simple positions: Axis perpendicular to horizontal plane, Axis perpendicular to vertical plane and Axis parallel to both the reference planes, Projection of Solids with axis inclined to one reference plane and parallel to another plane.	
Sections of Solids: Perpendicular and inclined section planes, Sectional views and True shape of sections for simple position only.	
	COs-C04
 UNIT-V:	 12 Hours
Conversion of Views: Conversion of isometric views to orthographic views and Conversion of orthographic views to isometric views for simple objects only.	
Computer graphics: Creating 2D&3D drawings of objects including PCB and Transformations using Auto CAD (Not for end examination).	
	COs-C05

TEXT BOOKS:

1.N. D.Bhatt, Engineering Drawing, Charotar Publishing House.

REFERENCE BOOKS:

1. Engineering Drawing, K.L.Narayana and P.Kannaiah, Tata McGrawHill.
2. Engineering Drawing, M.B.Shah and B.C. Rana, Pearson Education Inc.
3. Engineering Drawing with an Introduction to AutoCAD, Dhananjay Jolhe, Tata McGrawHill.

ONLINE RESOURCES:

1. <https://www.iitg.ac.in/rkbc/me111.htm>
2. <https://archive.nptel.ac.in/courses/112/105/112105294/>

E-BOOKS: <https://www.pdfdrive.com/textbook-of-engineering-drawing-e28918244.html>

Data Structures through C**IB.TECH- IISEMESTER(Common to CSE, CSE (DS) & CSE (AI&ML))**

Course Title: Data Structures through C	Course Code: R24CSPC01
Teaching Scheme(L:T:P): 3 0 0	Credits:3
Type of Course: Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: a basic understanding of programming concepts and some familiarity with C: Programming concepts, C knowledge, Code editors. and Mathematics.	

COURSEOVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSEOBJECTIVES:

The objectives of this course are to:

1. Understanding of fundamental data structures and algorithms.
2. To understand importance of data structure in context of writing efficient program.
3. Create and manage linked lists to efficiently organize and manipulate data, emphasizing memory efficiency.
4. Implement and apply stacks to manage program flow and solve problems involving expression evaluation and backtracking.
5. Explore basic concepts of hashing and apply it to solve problems requiring fast data retrieval and management.
6. To solve problems using data structures such as binary trees, binary search trees, and graphs.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Explain the role of linear data structures in organizing and accessing data efficiently in algorithms
CO2	Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems.
CO3	Apply queue-based algorithms for distinguish between dequeues and priority queues, and apply them appropriately to solve data management challenges, and apply Hash based solutions for specific problems.
CO4	Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation.
CO5	Implement operations on Binary tree, Demonstrate the representation and traversal techniques of graphs and their applications.

COURSECONTENT(SYLLABUS)**UNIT-I: Introduction to Linear Data Structures, Searching, and Sorting** **10 Hours**

Introduction to Linear Data Structures: Definition and importance of linear data structures, Abstract data types (ADTs) and their implementation, Overview of time and space complexity analysis for linear data structures.

Searching Techniques: Linear & Binary Search.

Sorting Techniques: Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort. **COs–CO1**

Self-Learning Topics: Analyse Time complexity of searching and sorting techniques

UNIT–II: Stacks and its Operations **10 Hours**

Stacks: Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, Applications of stacks in expression evaluation (Prefix, Postfix, Infix), backtracking, reversing list etc. **COs–CO2**

Self-Learning Topics: Types of Stacks

UNIT-III: Queues and its Operations **12 Hours**

Queues: Introduction to queues: properties and operations, implementing queues using arrays and linked lists, Applications of queues in breadth-first search, scheduling, etc.

Deque: Introduction to deque (double-ended queues), Operations on deque and their applications, Circular Queues, Priority Queues, Multiple Queues.

Hash Tables: Hash table implementation, Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing and rehashing, extensible hashing. **COs–CO3**

Self-Learning Topics: Different types of Queues

UNIT- IV Linked List **14 Hours**

Linked Lists: Singly linked lists, representation and operations, Applications on Single Linked List- Polynomial Expressions, Sparse Matrix Expressions, Advantages and disadvantages of Single Linked list, doubly linked lists and circular linked lists, Comparing arrays and linked lists, Applications of linked lists. **COs–CO4**

Self-Learning Topics: Swap nodes in a single linked list (without swapping data).

UNIT-V: Introduction to Graphs and Trees **14 Hours**

Graphs: Basic Concepts, Representations of Graphs-Adjacency Matrix and using Linked list, Graph Traversals (DFS&BFS), Graph Applications.

Trees: Introduction to Trees, Properties of Binary Tree, Representation of Binary Tree using Arrays and Linked List, Binary Search Tree – Insertion, Deletion & Traversals, B-Trees, B+ Trees, AVL Trees.

COs–CO5

Self-Learning Topics: Red-Black Trees

TEXT BOOKS:

1. 'Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.
2. 'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.
3. 'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

Web References:

1. <http://www.hackerrank.com/domains/datastructures>
2. http://www.github.com/topics/data_structures_c
3. <http://nptel.ac.in/courses>
4. <http://www.cslibrary.stanford.edu>

APPLIED CHEMISTRY LAB**IB.TECH-II SEMESTER** (Common to CSE, CSE (DS) & CSE (AI&ML))

Course Title: APPLIED CHEMISTRY LAB	Course Code: R24BS06
Teaching Scheme(L:T:P): 0:0:2	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Prerequisites:	

COURSE OVERVIEW:

To succeed in an *APPLIED CHEMISTRY Lab* course, certain foundational skills and knowledge are necessary for effective participation and understanding. Here are the key prerequisites:

1. Basic Chemistry Knowledge
2. Measurement
3. Basic Laboratory Skills
4. Problem-Solving and Analytical Skills
5. Familiarity with Safety Practices

COURSE OBJECTIVES:

1. Verify the fundamental concepts with experiments.
2. Learn and carry out some of the important experiments related to batteries and their properties.
3. Learn the preparation of engineering polymer materials like Bakelite
4. Know the fundamental principles of chemistry lab experiments which include volumetric analysis, dichrometry, conductometry and potentiometer

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Determine the cell constant and conductance of solutions. Determine redox potentials
CO2	Prepare advanced polymer Bakelite materials. Calculate strength of acid in Pb-Acid battery and Ferrous Iron by Dichrometry
CO3	Measure the strength of an acid present in secondary batteries. Moisture content in a coal sample.

COURSE CONTENT (SYLLABUS)**List of Experiments**

1. Conductometric titration of strong acid vs. strong base
2. Conductometric titration of weak acid vs. strong base
3. Determination of cell constant and conductance of solutions
4. Potentiometry - determination of redox potentials and emfs
5. PH metric titration – determination of Strength of Strong acid vs Strong base
6. Determination of Strength of an acid in Pb-Acid battery
7. Determination of Hardness of Water
8. Determination of KMnO₄ Using Standard Oxalic Acid Solution
9. Adsorption of acetic acid by charcoal
10. Estimation of Ferrous Iron by Dichrometry
11. Preparation of a Bakelite
12. Preparation of nanomaterials by precipitation method.

REFERENCE BOOKS:

Vogel's Quantitative Chemical Analysis 6th Edition 6th Edition" Pearson Publications by J. Mendham, R.C.Denney, J.D.Barnes and B. Sivasankar

Basic Electrical and Electronics Engineering Lab
IB.TECH-II SEMESTER(Common toCSE, CSE(AIML), CSE(DS))

Course Title: Basic Electrical and Electronics Engineering Lab	Course Code: R24ES07
Teaching Scheme(L:T:P): 0:0:3	Credits:1.5
Type of Course: Practical	
ContinuousInternalEvaluation:30Marks	Semester End Exam: 70Marks
Pre requisites:Understanding of Circuit Components, Breadboard connections.	

COURSEOVERVIEW:

In this lab, the students of all engineering streams are trained on basic concepts of electrical engineering, such as DC circuits, AC circuits, Resonance for series RLC and Parallel RLC circuit, AC to DC conversion, measurement, Efficiency and voltage regulation of transformer, electrical machines, verification of basic laws and theorems.

COURSEOBJECTIVES:

The objectives of this course are to

impart knowledge on the fundamental laws & theorems of electrical circuits, functions of electrical machines and energy calculations.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Apply the theoretical concepts and operating principles to derive mathematical models for circuits, Electrical machines and measuring instruments; calculations for the measurement of resistance, power and power factor.
CO2	Apply the theoretical concepts to obtain calculations for the measurement of resistance, power and power factor.
CO3	Plot and discuss the characteristics of various electron devices/instruments.
CO4	Design suitable circuits and methodologies for the measurement of various electrical parameters; Household and commercial wiring.
CO5	Understand the usage of electronic measuring instruments.
CO6	Plot and discuss the characteristics of various electron devices.

List of Experiments:**Part A-Basic Electrical Engineering**

- | | |
|--|----------------|
| 1. Verification of KCL and KVL | COs:CO1 |
| 2. Verification of Super position theorem | COs:CO1 |
| 3. Measurement of Resistance using Wheat stone bridge | COs:CO1 |
| 4. Magnetization Characteristics of DC shunt Generator | COs:CO1 |
| 5. Measurement of Power and Power factor using Single-phase watt meter | COs:CO2 |
| 6. Verification of ohms law | COs:CO1 |
| 7. Calculation of Electrical Energy for Domestic Premises | COs:CO3 |

Part B: Basic Electronics Engineering

- | | |
|---|-----------------|
| 1. Plot V-I characteristics of PN Junction diode A) Forward bias B) Reverse bias | COs:CO3 |
| 2. Plot V –I characteristics of Zener Diode and its application as voltage Regulator. | COs:CO3 |
| 3. Implementation of half wave and full wave rectifiers | COs: CO3 |

4. Plot Input & Output characteristics of BJT in CE and CB configurations **COs: CO3**
5. Frequency response of CE amplifier. **COs: CO3**
6. Simulation of RC coupled amplifier with the design supplied **COs: CO3**
7. Verification of Truth Table of AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates
Using ICs. **COs:CO2**
8. Verification of Truth Tables of S-R, J-K& D flip flops using respective ICs. **COs: CO2**

REFERENCEBOOKS:

1. BasicElectricalEngineering,D.C.Kulshreshtha,TataMcGrawHill,2019,FirstEdition
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A.Chakrabarti,Dhanpat Rai & Co, 2013
3. FundamentalsofElectricalEngineering,RajendraPrasad,PHIpublishers,2014,ThirdEdition
4. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
5. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009
6. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education, 2009.

Data Structures through C Lab
IB.TECH- ISEMESTER(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Title: Data Structures through C Lab	Course Code: R24CSPC02
Teaching Scheme(L:T:P): 0 0 3	Credits: 1.5
Type of Course: Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Pre requisites: Data Structures through C Lab, you should have a good understanding of the following topics: Software development,Programming languages,Time and space complexity analysis.	

COURSEOVERVIEW:

1. Basics of data structures including their fundamentals building blocks: arrays and linked list.
2. To solve problems using linear data structures such as linear lists, stacks, queues.
3. To solve problems using searching and sorting techniques.
4. To be familiar with non-linear data structures such as trees.

COURSEOBJECTIVES:

The objectives of this course are to:

The course aims to give students hands – on experience and train them on the concepts of the C-programming language.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Explain the role of linear data structures in organizing and accessing data efficiently in algorithms.
CO2	Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems & Queues.
CO3	Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation, Graphs & Trees.

COURSECONTENT(SYLLABUS)**Developing the following programs:****Week 1:**

1. Implement an array and perform operations such as insertion, deletion, and traversal.
2. Given an M*N integer matrix, if an element is 0, set its entire row and column to 0's.
3. Write a program that uses both recursive and non-recursive functions to perform linear search operations for a key value in the given list of integers.

Week 2:

4. Develop a program to perform quick sort, merge sort.
5. Create a program to perform bubble sort and selection sort.
6. Given an unsorted array of integers, find the length of the longest consecutive elements sequence.

Week 3:

7. Implement a stack using arrays.
8. Delete middle element of a stack
9. Develop a program to evaluate a postfix expression using stack.
10. Convert the prefix expression to postfix expression using the Stacks

Week 4:

11. Implement Queue operations using Arrays..
12. Create a circular queue and perform operations like insertion and deletion.

Week 5:

13. Develop a scenario-based program to simulate a ticket booking system using queues.
14. Implement a linked list and perform operations such as insertion, deletion, and traversal.
15. Detect the cycle in the given linked list.

Week 6:

16. Given two singly linked lists, determine if they intersect and return the intersecting node
17. Implement operations on Circular Linked List.
18. Remove Nth Node from end of List

Week 7:

19. Write a program that uses functions to perform the operations on Doubly Linked List.
20. Write a program that reverse the given linked list without using new list or array.
21. Delete last occurrence of an item from linked list.

Week 8:

22. Implement a queue using arrays and linked lists.

Week 9:

23. Create a scenario-based program to manage a library system using linked lists.

Week 10:

24. Create a program to find the height of a binary tree.
25. Create a program to find sub tree of another tree.

Week 11:

26. Develop a program to check if two binary trees are identical.

Week 12:

27. Develop a binary tree and perform in-order, pre-order, and post-order traversal.

Week 13:

28. Implement a binary search tree (BST) and perform operations such as insertion, deletion, and searching.

Week 14:

29. Implement graph data structure and perform depth-first search (DFS) and breadth-first search (BFS).

Week 15:

30. Implement a scenario-based program for a social network using graph data structures.

1. Delete duplicate-value nodes from a sorted Linked list| Linked List

You are given the pointer to the head node of a sorted linked list, where the data in the

nodes is in ascending order. Delete nodes and return a sorted list with each distinct value in the original list. The given head pointer may be null indicating that the list is empty.

Example:

head refers to the first node in the list.

1->2->2->3->3->3->3->NULL

Remove 1 of the 2 data values and return head pointing to the revised list.

Function Description

Complete the *removeDuplicates* function in the editor below.

removeDuplicates has the following parameter:

- *SinglyLinkedListNode pointer head*: a reference to the head of the list

Returns

- *SinglyLinkedListNode pointer*: a reference to the head of the revised list

2. Sparse Arrays| Array

There is a collection of input strings and a collection of query strings. For each query string, determine how many times it occurs in the list of input strings. Return an array of the results.

Example

stringList=['ab','ab','abc']

queries=['ab','abc','bc']

There are 2 instances of 'ab', 1 of 'abc' and 0 of 'bc'. For each query, add an element to the return array, results=[2,1,0] .

Function Description

Complete the function *matchingStrings* in the editor below. The function must return an array of integers representing the frequency of occurrence of each query string in *stringList*.

matchingStrings has the following parameters:

- *string stringList[n]* - an array of strings to search
- *string queries[q]* - an array of query strings

Returns

- *int[q]*: an array of results for each query

3. Mini-Max Sum

Given five positive integers, find the minimum and maximum values that can be calculated by summing exactly four of the five integers. Then print the respective minimum and maximum values as a single line of two space-separated long integers.

Example

arr=[1,3,5,7,9]

The minimum sum is $1+3+5+7=16$ and the maximum sum is $3+5+7+9=24$.

The function prints:16 and 24

Textbooks:

Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.

'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.

'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

ENGINEERING WORKSHOP**IB.TECH-II SEMESTER(Common to CSE, CSE(AIML), CSE(DS))**

Course Title : ENGINEERING WORKSHOP	Course Code: R24ES08
Teaching Scheme(L:T:P): 0:0:3	Credits:1.5
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSEOVERVIEW:

- The Engineering Workshop Lab introduces students to essential hands-on skills across multiple trades, fundamental for understanding material manipulation, joining techniques, and assembly processes. Through eight core experiments, students learn woodworking, sheet metal fabrication, fitting, foundry molding, welding, electrical wiring, plumbing, and blacksmithing. These exercises build practical knowledge in creating structures, forming metal parts, achieving precision fits, casting molds, and assembling electrical and plumbing systems. This workshop lays a solid foundation for understanding and applying basic engineering processes, crucial for practical problem-solving and project execution in various engineering fields.

COURSEOBJECTIVES:

The objectives of this course are to

- Describe how different tools are used in home wiring, tin smiting, blacksmithing, carpentry, and fitting.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Identify workshop tools and their operational capabilities. Practice on manufacturing of components using workshop trades including carpentry, fitting, sheet metal
CO2	Practice on manufacturing of components using workshop trades including foundry and welding.
CO3	Apply fitting operations in various applications and engineering knowledge for Plumbing, House Wiring Practice, and Making square rod and L-bend from the round rod in black smithy

COURSECONTENT(SYLLABUS)

- Wood Working**
 - Half Lap joint
 - Mortise and Tenon joint
 - Corner Dovetail joint or Bridle joint
- Sheet Metal Working**
 - Tapered tray
 - Conical funnel
 - Elbow pipe
 - Brazing
- Fitting**

- a) V-fit
- b) Dovetail fit
- c) Semi-circular fit
- d) Bicycle tire puncture and change of two-wheeler tyre

4. Foundry Trade: Preparation of Green Sand Moulds

- a) Single piece pattern
- b) Double piece pattern

5. Welding Shop: Arc welding Practice

- a) Lap joint
- b) Butt joint

6. Electrical Wiring

- a) Parallel and series connection
- b) Two-way switch connection
- c) Tube light connection
- d) Soldering of wires

7. Plumbing

- a) Prepare Pipe joint with coupling for 1 inch diameter
- b) Prepare Pipe joint with coupling for 1.5inch diameter

8. Black smithy

- a) Round rod to Square
- b) Round rod to S-Hook

TEXTBOOKS:

1. Basic Workshop Technology: Manufacturing Process, Felix W.; Independently Published, 2019. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5th Edn. 2015.
2. A Course in Workshop Technology Vol I. & II, B.S. Raghuwanshi, Dhanpath Rai & Co., 2015 & 2017

REFERENCEBOOKS:

1. Elements of Workshop Technology, Vol. I by S. K. Hajra Choudhury & Others, Media Promoters and Publishers, Mumbai. 2007, 14th edition
2. Workshop Practice by H.S. Bawa, Tata-McGrawHill, 2004.
3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan 2021-22

ONLINE RESOURCES:

1. https://youtube.com/playlist?list=PLzkMouYverALpuDJ4g4TiICc6_vLcS1Ny&si=YGrVJY8uB0tHy_iQ

E-BOOKS:

1. <https://www.pdfdrive.com/workshop-processes-practices-and-materials-third-edition-d158706794.html>
2. <https://www.pdfdrive.com/introduction-to-basic-manufacturing-processes-and-workshop-e217530.html>
3. <https://www.pdfdrive.com/workshop-technology-e55714020.html>

NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE
IB.TECH-II SEMESTER (Common to all Branches)

Course Title: NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE	Course Code: R24MC02
Teaching Scheme(L:T:P): 0:0:1	Credits:0.5
Type of Course: Practical	
Continuous Internal Evaluation:100Marks	Semester End Exam: 0Marks
Pre requisites: Prerequisites for programs related to NSS (National Service Scheme) , NCC (National Cadet Corps) , Scouts and Guides , and Community Service can vary by institution and specific program. However, here are general guidelines and common prerequisites for each: National Service Scheme (NSS), National Cadet Corps (NCC), Scouts and Guides, Community Service Programs.	

COURSE OVERVIEW:

Here's a general course overview for programs related to NSS (National Service Scheme), NCC (National Cadet Corps), Scouts and Guides, and Community Service. Each program focuses on different aspects of personal development, leadership, and community engagement.

1. National Service Scheme (NSS)

- **Overview:** NSS is a voluntary program aimed at fostering a sense of social responsibility and community service among students. It emphasizes the importance of personal and community development.
- **Core Components:**
 - **Community Service Projects:** Participation in various community development activities, such as health camps, environmental awareness programs, and literacy initiatives.
 - **Workshops and Seminars:** Educational sessions on social issues, health, hygiene, and community development strategies.
 - **Leadership Development:** Training sessions focused on leadership skills, teamwork, and effective communication.
 - **Camps and Activities:** Organizing and participating in camps, rallies, and other events that promote social awareness and civic responsibility.

2. National Cadet Corps (NCC)

- **Overview:** NCC is a youth development movement that aims to develop character, discipline, leadership, and a spirit of adventure among young people.
- **Core Components:**
 - **Military Training:** Basic training in drill, weapons handling, and military tactics, combined with emphasis on discipline and teamwork.
 - **Adventure Activities:** Participation in activities such as trekking, mountaineering, and camping to foster adventure skills and resilience.
 - **Community Service:** Involvement in social service initiatives and community development projects.
 - **Leadership and Management Skills:** Training sessions focused on leadership, communication, and management, preparing cadets for future responsibilities.

3. Scouts and Guides

- **Overview:** Scouts and Guides programs promote personal development, leadership skills, and community service among young people through various outdoor and indoor activities.

- **Core Components:**
 - **Skill Development:** Learning practical skills such as first aid, navigation, and survival skills.
 - **Community Projects:** Engaging in community service projects and environmental conservation efforts.
 - **Outdoor Activities:** Camping, hiking, and other outdoor adventures that promote teamwork, resilience, and a love for nature.
 - **Values and Ethics:** Education on values such as integrity, respect, and service, aligning with the principles of scouting.

4. Community Service Programs

- **Overview:** Community service programs are designed to engage individuals in volunteer work that benefits their communities, fostering civic responsibility and social awareness.
- **Core Components:**
 - **Volunteer Projects:** Participation in various service projects, such as assisting in local shelters, food banks, environmental clean-ups, and educational initiatives.
 - **Skill-Building Workshops:** Workshops on leadership, teamwork, and project management to enhance volunteers' capabilities.
 - **Awareness Campaigns:** Engaging in campaigns to raise awareness about social issues, such as health, education, and the environment.
 - **Reflection and Evaluation:** Opportunities to reflect on experiences, discuss challenges, and evaluate the impact of their service.

COURSEOBJECTIVES:

The objectives of this course are to:

The objective of introducing this course is to impart discipline, character, fraternity, teamwork, social consciousness among the students and engaging them in selfless service.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Understand the importance of discipline, character and service motto.
CO2	Solve some societal issues by applying acquired knowledge, facts, and techniques.
CO3	Explore human relationships by analyzing social problems.
CO4	Determine to extend their help for the fellow beings and downtrodden people
CO5	Develop leadership skills and civic responsibilities.

COURSECONTENT(SYLLABUS)

UNIT I Orientation

General Orientation on NSS/NCC/ Scouts & Guides/Community Service activities, career guidance.

Activities:

- i) Conducting –ice breaking sessions-expectations from the course-knowing personal talents and skills
- ii) Conducting orientations programs for the students –future plans-activities-releasing road map etc.
- iii) Displaying success stories-motivational biopics- award winning movies on societal issues etc.

iv) Conducting talent show in singing patriotic songs-paintings- any other contribution.

UNIT II Nature & Care

Activities:

- i) Best out of waste competition.
- ii) Poster and signs making competition to spread environmental awareness.
- iii) Recycling and environmental pollution article writing competition.
- iv) Organising Zero-waste day.
- v) Digital Environmental awareness activity via various social media platforms.
- vi) Virtual demonstration of different eco-friendly approaches for sustainable living.
- vii) Write a summary on any book related to environmental issues.

UNIT III Community Service

Activities:

- i) Conducting One Day Special Camp in a village contacting village-area leaders- Survey in the village, identification of problems- helping them to solve via media- authorities experts-etc.
- ii) Conducting awareness programs on Health-related issues such as General Health, Mental health, Spiritual Health, HIV/AIDS,
- iii) Conducting consumer Awareness. Explaining various legal provisions etc.
- iv) Women Empowerment Programmes- Sexual Abuse, Adolescent Health and Population Education.
- v) Any other programmes in collaboration with local charities, NGOs etc

Reference Books:

1. Nirmalya Kumar Sinha & Surajit Majumder, *A Text Book of National Service Scheme Vol: I*, Vidya Kutir Publication, 2021 (ISBN 978-81-952368-8-6)
2. *Red Book - National Cadet Corps* – Standing Instructions Vol I & II, Directorate General of NCC, Ministry of Defence, New Delhi
3. Davis M. L. and Cornwell D. A., —Introduction to Environmental Engineering, McGraw Hill, New York 4/e 2008
4. Masters G. M., Joseph K. and Nagendran R. —Introduction to Environmental Engineering and Science, Pearson Education, New Delhi. 2/e 2007
5. Ram Ahuja. *Social Problems in India*, Rawat Publications, New Delhi.



AVANTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113. www.avanthienggcollege.ac.in,
mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Program: **B.Tech Computer Science and Engineering**

Regulation: R24

II Year I Semester-Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	ES	R24CSES09	Mathematical Foundations of Computer Science	3	0	0	3
2	PC	R24CSPC03	Design and Analysis of Algorithms	3	0	0	3
3	PC	R24CSPC04	Object Oriented Programming using JAVA	3	0	0	3
4	PC	R24CSPC05	Digital Logic and Computer Organization	3	0	0	3
5	PC	R24CSPC06	Software Engineering and Project Management	3	0	0	3
6	PC	R24CSPC07	Object Oriented Programming using JAVA Lab	0	0	3	1.5
7	PC	R24CSPC08	Algorithms For Efficient coding Lab	0	0	3	1.5
8	SC	R24CSSC01	Full Stack Development	0	0	3	1.5
9	HS	R24CSHS05	Soft skills& Verbal Ability	0	0	2	1
10	MC	R24MC04	Indian Traditional Knowledge	2	0	0	-
Total				17	00	11	20.5

Category	Courses	Credits
ES- Engineering Science Course	1	03
PC-Proficitional Core Course	6	15
HS- Humanity Science and Management Course	1	01
SC- Skill Oriented Course	1	1.5
MC-Mandatory Course	1	00
Total	10	20.5

R24CSES09 Mathematical Foundations of Computer Science 3 0 0 3

(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

The course is designed with the objective to:

- Apply the knowledge in equivalence formulas, tautological implications in finding normal forms, and theory of inference in Statement Calculus and predicates, and explain Mathematical Induction principle and apply the same
- Develop the skill in equivalences and inference theory in Predicate Calculus
- Learn the properties of relations, Posets, Lattices, sets and to apply in problem solving skills
- Acquire the knowledge and skill to draw graphs and related structures and to apply the knowledge skill in solving problem
- Enhance skill and knowledge to draw of trees and solve minimum cost spanning tree problems and solve and formulate, generating functions and recurrence relations.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs										
		PO1	PO2	PO3	PO4	PO5	PO6	PO10	PO11	PSO1	PSO2	BT
R24ES09.1	Apply equivalence formulas, tautological implications in finding normal forms, and theory of Inference in Statement Calculus and predicates	3	3	2	2	3	2	3	2	3	2	L2
R24ES09.2	Apply skill in equivalences and inference theory in Predicate Calculus.	3	3	3	3	3	3	3	2	3	2	L3
R24ES09.3	Students gain the knowledge of the properties of relations, Posets, Lattices, Sets and apply the same in solving the problems.	3	3	3	3	3	2	3	2	3	3	L3
R24ES09.4	solve Generating functions and recurrence relations	3	3	3	2	3	3	3	3	3	2	L3

R24ES09.5	Identify the properties of graphs and related structures and solve the related problems.											
	Identify the properties of Trees and solve minimum cost spanning tree problems	3	3	3	2	3	3	-	-	-	-	L3

SYLLABUS

UNIT-I

14 hours

Mathematical logic: Fundamentals (statements and notations, connectives, Truth tables), Tautologies, Equivalence of formulas, Tautological implications, Normal forms, Theory of Inference for statement calculus. **CO's-CO1**

Self-Learning Topics:

- Build and analyze truth tables manually.
- Practice rewriting expressions into CNF/DNF.
- Memorize common logical equivalences.

UNIT-II

14 hours

Predicate Calculus: Predicate logic, statement functions, variables and quantifiers, free and bound variables. Inference Theory of the Predicate Calculus: Logical implication involving quantifiers, Statements with more than one variable. **CO's-CO2**

Self-Learning Topics:

- Memorize Inference Rules.
- Practice symbolic representation of logical statements with quantifiers

UNIT-III

15 hours

Set Theory: Operations on Sets, Principle of Inclusion-Exclusion.

Relations: Relations, Properties of Relations, Equivalence relations, Operations on relations, Representation of relations, partial orders, Hasse Diagram, Lattices, properties of Lattices, Special types of Lattices (Proofs not required). **CO's-CO3**

Self-Learning Topics:

- Practice solving problems on inclusion-exclusion
- Memorize the properties of relations

UNIT-IV

14 hours

Recurrence relations: Generating Function of Sequences, calculating coefficient of Generating function, Partial Fractions. First order Linear Homogeneous and Non- Homogeneous recurrence relations, Method of generating functions, Method of characteristic root. **COs-CO4**

Self-Learning Topics:

- Solve recurrence problems using both methods

UNIT-V

13 hours

Graph Theory: Basic Concepts of Graphs, Matrix representation of graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian & Hamiltonian graphs, Planar Graphs, Graph coloring.

Trees: Introduction, Spanning Trees, BFS, DFS, Minimum cost spanning trees (Prim's & Kruskal's)

Self-Learning Topics:

- Memorize all types of graphs and their properties
- Practice the problems on Isomorphic, Hamiltonian and Eulerian graphs
- Practice BFS, DFS, Prim's, Kruskal

Board of Studies : Computer Science and Engineering

Approved in BoS No : 02, --, April 2025

Approved in ACM No: 02

Text Books:

1. Trembly J.P. and Manohar. P, "Discrete Mathematical Structures with applications to computer science," Tata McGraw Hill, New Delhi, 2017
2. Kolman B, Busoy R.C, Ross S.C, Discrete Mathematical Structures, 5th Edition, Prentice Hall, 2004.
3. D.S. Chandra sekharaiyah, "Mathematical Foundation of Computer Science" Prism Publications 2009.

Reference Books:

1. J.L. Mott, A. Kandel, T. P Baker, "Discrete Mathematics for Computer Scientists and Mathematicians," Prentice Hall India, 2nd Edition 2015
2. Kenneth H. Rosen, "Discrete Mathematics and its Applications," Tata McGraw Hill, New Delhi, 7th edition, 2017
3. V. Krishnamurthy, "Combinatory: Theory and Applications", East-West Press. Seymour Lipschutz, M. Lipson, "Discrete Mathematics" Tata McGraw Hill, 2005.

Reference Links:

1. <https://nptel.ac.in/courses/106106094>
2. <https://nptel.ac.in/courses/106103205>
3. <https://nptel.ac.in/courses/106106183>

**Chairperson
Board of Studies (CSE)**

R24CSPC03| Design and Analysis of Algorithms 3 0 0 3
 (Common to CSE, CSE (DS), CSE (AI&ML))

Course Objectives:

1. To understand and apply the algorithm analysis techniques
2. To critically analyze the efficiency of alternative algorithmic solutions for the same problem
3. To understand different algorithm design techniques
4. To understand the limitations of Algorithmic power

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

Course Code	Course Outcomes	Mapping with POs and PSOs							Dok
		PO1	PO2	PO3	PO4	PS01	PS02	PS03	
R24CSPC03.1	Review the fundamentals of algorithmic problem solving and analyzing efficiency of algorithms	3	3	1	0	2	3	3	L1,L2,L4
R24CSPC03.2	Apply mathematical formulation, complexity analysis and methodologies	3	3	1	0	2	3	3	L2,L3
R24CSPC03.3	Compare the time complexities of various algorithms	2	3	0	1	3	2	3	L3,L4
R24CSPC03.4	Critically analyze the different algorithm design techniques for a given problem	2	3	0	2	2	3	3	L3,L4
R24CSPC03.5	Illustrate NP class problems and formulate solutions using standard approach	1	2	0	1	2	3	3	L4,L5

SYLLABUS

UNIT-I: Introduction of an Algorithm **15 Hours**

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving –Important Problem Types – Fundamentals of the Analysis of Algorithmic Efficiency –Asymptotic Notations and their properties - Analysis Framework –Empirical analysis - Mathematical analysis for Recursive and Non-recursive algorithms –Visualization.

Self-Learning Topics: Amortized analysis **COs–CO1**

UNIT–II: Brute Force and Divide-and-Conquer **13 Hours**

Brute Force – Computing an – String Matching - Closest-Pair and Convex-Hull Problems - Exhaustive Search - Travelling Salesman Problem – Knapsack Problem - Assignment Problem.

Divide and Conquer Methodology – Binary Search – Merge sort – Quick sort –Heap Sort - Multiplication of Large Integers – Closest-Pair and Convex – Hull Problems.

Self-Learning Topics: Knuth-Morris-Pratt Algorithm **COs–CO2**

UNIT-III: Dynamic Programming and Greedy Techniques **14 Hours**

Dynamic programming – Principle of optimality - Coin Changing Problem -Computing a Binomial Coefficient – Floyd's Algorithm – Multi Stage Graph -Optimal Binary Search Trees – Knapsack Problem and Memory functions

Greedy Technique – Container Loading Problem - Prim's Algorithm and Kruskal's Algorithm – 0/1 Knapsack Problem - Optimal Merge pattern -Huffman Trees.

Self-Learning Topics: Travelling Sales Person Problem **COs–CO3**

UNIT- IV Iterative Improvement **14 Hours**

The Simplex Method - The Maximum-Flow Problem – Maximum Matching in Bipartite Graphs - Stable Marriage problem

Self-Learning Topics: Hill Climbing **COs–CO4**

UNIT-V: Coping with the Limitations of Algorithm Power **14 Hours**

Lower - Bound Arguments - P, NP, NP - Complete and NP-Hard Problems-Backtracking – n-Queen Problem - Hamiltonian Circuit Problem – Subset Sum Problem. Branch and Bound – LIFO Search and FIFO Search –Assignment Problem – Knapsack Problem – Travelling Salesman Problem.

Self-Learning Topics: Approximation Algorithms for NP-Hard Problems – Travelling Salesman Problem – Knapsack Problem **COs–CO5**

Board of Studies : Computer Science and Engineering

Approved in BOS No: 02, --, May, 2025

Approved in ACM No: 02

Expert Talk (To be delivered by SMEs from Industries)

	COs	POs / PSOs
1. Branch and Bound Algorithmic Approach	CO5	PO1,PO2,PO4,PSO1,PSO2,PSO3

Text Books:

1. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, “Computer Algorithms/ C++”, Second Edition, Universities Press, 2007

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, PHI Learning Private Limited, 2012.
2. Alfred V.Aho, John E. Hopcroft and Jeffrey D. Ullman, “Data Structures and Algorithms”, Pearson Education, Reprint 2006.
3. Harsh Bhasin, “Algorithms Design and Analysis”, Oxford university press,2016.
4. S. Sridhar, “Design and Analysis of Algorithms”, Oxford university press,2014

Web References:

1. <https://nptel.ac.in/courses/106101060>
2. https://www.cse.iitm.ac.in/course_details.php?arg=OTI
3. https://swayam.gov.in/nd1_noc19_cs47/previ

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
-----------------	---------------------------	---------------------------

L1	35	-
L2	30	-
L3	20	30
L4	15	40
L5	-	30
TOTAL (%)	100	100

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is an algorithm, and why is it important in computer science?
2. Name two examples of important problem types in algorithm design.
3. What does asymptotic notation analyze, and what are three common types of asymptotic notations?
4. Define dynamic programming and provide a short description of its principle of optimality.
5. What is the difference between NP and NP-Hard problems?

L2: Understand

1. Explain the purpose of empirical analysis versus mathematical analysis when evaluating algorithm efficiency.
2. Describe how the divide-and-conquer methodology works and provide an example of an algorithm that uses this approach.
3. Discuss the advantages and disadvantages of using a brute force approach versus a more efficient algorithm like quick sort.
4. What role does visualization play in understanding and analyzing algorithms?
5. How does the Simplex Method solve linear programming problems?

L3: Apply

1. Given an algorithm to sort a list of numbers, analyze its time complexity using asymptotic notation to demonstrate its efficiency.
2. Implement a brute-force solution for the string matching problem, and analyze the time complexity of your approach.
3. Use dynamic programming to solve the Coin Changing Problem and explain your solution step-by-step.
4. Apply the concept of the maximum-flow problem to a specific network example and demonstrate how to calculate the maximum flow.
5. Create a greedy algorithm to tackle the 0/1 Knapsack Problem and explain the logic behind your selection at each step.

L4: Analyze

1. Compare the performance of the naive recursive solution versus a dynamic programming solution for the Fibonacci sequence in terms of time and space.
2. Analyze why the Travelling Salesman Problem is categorized as NP-Complete and discuss

its implications for algorithm design.

3. Evaluate the efficiency of various algorithms (like Merge Sort and Quick Sort) for sorting a large dataset under different conditions.
4. Investigate the effectiveness of the branch-and-bound method compared to backtracking for solving combinatorial optimization problems.
5. Examine the potential trade-offs in solving an optimization problem using iterative improvement methods versus exact algorithms.

L5: Evaluating

1. Critically assess how well amortized analysis can predict the performance of data structures over time.
2. Evaluate the practical applications of approximation algorithms for NP-Hard problems, citing specific examples.
3. Discuss the implications of classifying problems into different categories (P, NP, NP-Complete, and NP-Hard) in the context of computer science and real-world applications.
4. Assess the effectiveness of dynamic programming techniques in practical scenarios compared to greedy algorithms.
5. Evaluate the significance of understanding algorithmic limitations in practical computing, particularly regarding the P vs NP debate.

**Chairperson
Board of Studies (CSE)**

R24CSPC04**Object Oriented Programming using JAVA****3 0 0 3**

(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

The main objectives of the course is to

- Understand the basic ideas of Object-Oriented Programming (OOP) and implement them using Java through classes, objects, inheritance, and polymorphism.
- Understand and implement basic Java programming using data types, operators, control statements, and arrays to solve simple problems.
- Implement programs using Java features like constructors, method overloading, exception handling, and multithreading for better program design.
- Understand and use Java's built-in libraries like collections, file handling, and interfaces to create useful and organized programs.
- Implement graphical programs using JavaFX and understand how to create simple network-based applications using Java networking.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs						BT	
		PO 1	PO 2	PO 3	PO 5	PO 11	PS0 1		PS0 2
R24CSPC04.01	Implement object orientated programming strategies and Contrast classes and objects.	3	2	2	3	2	3	3	L2, L3
R24CSPC04.02	Analyze Inheritance and Dynamic Method Dispatch	3	3	2	2	2	2	3	L2, L3, L4
R24CSPC04.03	Demonstrate various classes in different packages and can design own packages	3	2	3	3	2	2	3	L2, L3
R24CSPC04.04	Manage Exceptions and Apply Threads	3	3	3	3	2	2	3	L3, L4
R24CSPC04.05	Create GUI screens along with event handling and write network programs	3	2	3	3	3	2	3	L5, L6

SYLLABUS**UNIT-I:****15 Hours**

INTRODUCTION TO OBJECTS & CLASSES: What is Object Oriented Programming? Object Orientation as a New Paradigm, An Overview of Java: Process Oriented Vs Object Oriented

Programming, OOP Principles, Java BuzzWords, The Byte Code, A First Simple Program. Class Fundamentals with Variables and Methods, Declaring objects for accessing variables and methods. Data Types and Variables, Operators and Expressions, Control Statements, Type Conversion and Casting, Lexical Issues in Java, command line arguments,

ARRAYS: Introduction, Declaration and Initialization of Arrays, Single Dimension, Operations on Array Elements, Sorting of Arrays, Search for Values in Arrays, Multi Dimension. **CO's-CO1**

Self-learning topics:

- Difference between JDK, JRE, and JVM
- History and evolution of Java
- Best practices for writing Java code (naming conventions, comments, code formatting)
- Enhanced for-loop and foreach in Java

UNIT-II:

13 Hours

CONSTRUCTORS: Default and Parameterized, this keyword and Garbage Collection, Final and Static Keywords, Overloading Methods, Overloading Constructors, Using objects as Parameters, Returning objects, Strings, String methods, String Buffer and String Builder.

INHERITANCE: Inheritance Basics, Types of Inheritance, Using Super keyword for constructors, Super to call variables and methods, Method Overriding, Dynamic Method Dispatch. **CO's-CO2**

Self-learning topics:

- Constructor chaining and private constructors
- Immutable classes and the final keyword usage in depth
- StringBuilder vs StringBuffer: performance comparison

UNIT-III:

15 Hours

PACKAGES AND INTERFACES: Defining a Package, importing a package, Package Example, Access Protection, An Access Example, Abstract classes, Interfaces: Defining and Implementing Interfaces

EXPLORING java.lang: Wrapper classes, Object, Math, Runtime

EXPLORING java.util: The collection framework: ArrayList, HashSet HashMap, String Tokenizer, Calendar, Random, Scanner

EXPLORING java.io: File class, Byte Streams, Character Streams, File Input Stream, FileOutputStream, FileReader and FileWriter classes. **CO's-CO3**

Self-learning topics:

- Creating custom packages and JAR files
- Interface vs Abstract class: When to use what?
- Iterators and ListIterator in Java Collections

- Serialization and Deserialization using java.io

UNIT-IV:

14 Hours

EXCEPTION HANDLING: Exception Handling Fundamentals, Exception Types, throw, throws and finally, Creating your own exceptions, Chained Exceptions.

MULTITHREADED PROGRAMMING: Java Thread Model, The Main thread, Two ways of Creating a Thread, Creating Multiple Threads, isAlive(), join(), Thread Priority, Synchronization, Inter Thread Communication.

COs-CO4

Self-learning topics:

- Try-with-resources statement in exception handling
- Uncaught exceptions and the default handler
- Daemon threads and thread groups

UNIT-V:

12 Hours

JAVA FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events

EXPLORING JAVA.NET: Socket, ServerSocket, InetAddress, DatagramSocket, URL, Client Server Program using Sockets.

COs-CO5

Self-learning topics:

- Creating custom controls in JavaFX
- Building forms and handling validation in JavaFX
- Introduction to Java NIO (New I/O)
- Understanding protocols (TCP vs UDP) in Java networking

Board of Studies : Computer Science and Engineering

Approved in BoS No : -- May 2025

Approved in ACM No: 02

Text Books:

1. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019
2. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson, 2009.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

Reference Books:

1. Cay S. Horstmann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition,

Pearson, 2015.

3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006

ReferenceLinks:

1. http://en.wikibooks.org/wiki/Java_Programming-JavaLearningWikiBook
2. <http://www.javabeginner.com>-JavaBeginnerTutorial

**Chairperson
Board of Studies (CSE)**

R24CSPC05**Digital Logic and Computer Organization****3 0 0 3**

(Computer Science and Engineering)

Course Objectives:

The main objectives of the course is to

- Provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals
- Apply Boolean algebras and simplify logic expressions and can design sequential circuits.
- Design combinational circuits and perform arithmetic operations using binary address, subtractors, multipliers and code converter.
- Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices
- Explore secondary storage mechanism and their role in memory hierarchy.

Course Outcomes

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

Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	BT
R24CSPC 10.01	Understand Significance of Number Systems, Conversions, and Binary Codes.	3	2	2	1	2	-	-	1	1	1	-	3	2	L2, L3
R24CSPC 10.02	Apply Different Simplification Methods For Minimizing Boolean Functions, The Operation Of Flip-Flops, Registers, And Counters	3	3	2	2	2	-	-	1	1	1	-	3	2	L1, L4
R24CSPC 10.03	Illustrate Knowledge On Design Of Various Combinational Circuits.	3	3	3	2	2	-	-	1	1	2	1	3	2	L1, L3
R24CSPC 10.04	Understand the Basic Structure of Computer and Memory Organization.	3	3	2	2	3	1	1	1	2	2	2	2	1	L2, L4
R24CSPC 10.05	Discuss The Basic Structure And Organization Of Computers.	3	3	2	2	3	1	2	1	1	2	2	2	1	L5, L6

SYLLABUS**UNIT I:****15 Hours**

Number Systems and Codes: Decimal, Binary, Octal, and Hexa-decimal number systems and their conversions, ASCII code, Excess-3 code, Gray code, Complement representation of negative numbers: Signed Magnitude, One's complement method, Two's complement method, Binary

Arithmetic.

Digital Logic Circuits-I: Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers

Self Learning Concepts:

CO's-CO1

Basic Gates: AND, OR, NOT, XOR Learn 2, 3, 4-variable K-Maps, Multiplexers (MUX): Select one input among many

UNIT II:

15 Hours

Boolean algebra: Boolean operations, Boolean functions, algebraic manipulations, min-terms and max-terms, sum-of-products and product-of-sum representations, two-input logic gates, NAND/NOR implementations, Minimization of Boolean functions using Karnaugh map, don't-care conditions, prime implicants, Tabular Method

CO's-CO2

Digital Logic Circuits-II: Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters

Self Learning Concepts: Shift Registers: Serial-In Serial-Out (SISO), Serial-In Parallel-Out (SIPO), etc

UNIT III:

15 Hours

Combinational Logic Design: Analysis of combinational circuits, Design Procedure – Binary Adder, Subtractor, BCD Adder, multiplier, comparator, decoders, encoders, multiplexers, demultiplexers, Code Converters

Computer Arithmetic : Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations

CO's-CO3

Self Learning Concepts: Use Half Adder (1-bit) and Full Adder (with carry), simulate arithmetic circuits and MUX/DEMUX.

UNIT IV:

15 Hours

Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von- Neumann Architecture.

Computer Organization

Organization and Architecture, Structure and Function, Computer Components, Bus Interconnection, Processor Organization, Register Organization. Instruction codes, Computer instructions, Memory reference instructions, Instruction Cycle. Stack organization, instruction formats, addressing modes, program control.

CO's-CO4Self

Learning Concepts: Memorize types of registers: Program Counter (PC), Instruction Register (IR), Accumulator, MAR, MDR

UNIT V:

10 Hours

The Memory Organization: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage. **CO'S-CO5**

Self Learning Concepts: Learn the difference between volatile (e.g., RAM) and non-volatile (e.g., ROM, SSD) memory,

Board of Studies : Computer Science and Engineering

Approved in BoS No: 02, -- April, 2025

Approved in ACM No: 02

Text Books:

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 6th edition, McGraw Hill
2. M. Moris Mano, Computer Systems Architecture, 3rd Edition, Pearson Education, 2007.
3. William Stallings, Computer Organization and Architecture: Designing for Performance, 10th Edition, Pearson Education, 2016

Reference Books:

1. Computer Systems Architecture, M. Moris Mano, 3rd Edition, Pearson
2. Computer Organization and Design, David A. Paterson, John L. Hennessy, Elsevier
3. Fundamentals of Logic Design, Roth, 5th Edition, Thomson M. Moris Mano, Computer Systems Architecture, 3rd Edition, Pearson Education, 2007.
4. John P. Hayes, Computer Architecture and Organization, 3rd Edition, WCB/McGraw-Hill.
5. M. Morris Mano and Michael D. Ciletti, Digital Design, 4th Edition, Pearson Education, 2013.
6. Vincent P. Heuring and Harry F. Jordan, Computer System Design and Architecture, 2nd Edition, Pearson Education, 2004.
7. Carl Hamacher, Computer Organization and Embedded Systems, 6th Edition, McGraw Hill Higher Education, 2002.

Online Learning Resources:

1. <https://archive.nptel.ac.in/courses/106/105/106105163/>
2. https://onlinecourses.nptel.ac.in/noc21_ee39/preview

Web References:

1. <http://www.c4learn.com/>
2. <http://www.geeksforgeeks.org/c/>
3. <http://nptel.ac.in/courses/106105163/>
4. <http://www.learn-c.org/>
5. <https://www.tutorialspoint.com/c-programming/>

**Chairperson
Board of Studies (CSE)**

R24CSPC06 Software Engineering and Project Management 3 0 0 3
(Computer Science and Engineering)

Course Objectives:

1. To comprehend software development life cycle.
2. To gain knowledge of requirement engineering and SRS documents.
3. To understand software architecture styles.
4. To learn various software testing techniques and their applicability.
5. To apply and analyze project management life cycle.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs							DoK	
		PO1	PO2	PO3	PO4	PO5	PO11	PSO1		PSO2
R24CSPC05-01	Describe the principles of software engineering, life cycle models	3	2	2	-	1	1	2	-	L1,L2
R24CSPC05-02	Analyze the computing requirements to solve a given problem	2	3	2	2	2	1	3	-	L1,L2,L3
R24CSPC05-03	Demonstrate the importance of software modeling and modeling languages	2	2	3	2	3	1	2	-	L2,L3,L4
R24CSPC05-04	Illustrate the necessity of software testing and design test cases for a software	2	3	2	3	3	1	3	-	L4,L5,L6
R24CSPC05-05	Interpret Software maintenance and state the concepts of project Management.	2	2	2	1	2	2	-	2	L4,L5,L6

SYLLABUS

UNIT-I:

10 Hours

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Software Product and Process Characteristics, Software Process customization and improvement, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

CO's-CO1

Self-Learning Topics: Advanced Software Development Paradigms

UNIT II:

15

HoursRequirements Analysis and Specification: Requirements gathering and analysis, Software

Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL, Petri Nets- Data Dictionary.**CO's–CO2**
Self-Learning Topics:Modern Requirements Engineering Approaches

UNIT – III: 15 Hours

Design process and Design Concepts, Design Model– Design Heuristic, Architectural Design - Architectural styles, Architectural Design, Architectural Mapping using Data Flow- User Interface, Design: Interface analysis, Interface Design, Component level Design: Designing Class based components, traditional Components) **CO's–CO3**

Self-Learning Topics:Design Patterns (Gang of Four - GoF)

UNIT- IV: 10 Hours

Coding And Testing: Coding, Code review, Software documentation, Software testing fundamentals, Internal and external views of Testing, Black-box testing, White-Box testing, , Debugging, Software **Implementation Techniques:** Coding practices- Refactoring, Maintenance and Reengineering-BPR model , Reengineering process model-Reverse and Forward Engineering, Program analysis tools, Unit Testing – Integration Testing – Validation Testing Q, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing. **.CO's–CO4**

Self-Learning Topics:Test-Driven Development (TDD) and Behavior-Driven Development (BDD)

UNIT-V: 10Hours

Software Maintenance & Software Project Measurement: Software Configuration Management (SCM), Software Change Management, Version Control, Change control and Reporting,Re-engineering, Reverse Engineering, Project Management Concepts, Project Scheduling and Tracking, Software Quality Assurance (SQA)

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, a reuse approach, and Reuse at organization level. **CO's-CO5**

Self-Learning Topics: Adaptive and Preventive Maintenance Strategies

Board of Studies : Computer Science and Engineering

Approved in BoS No : 02, --, April, 2025

Approved in ACM No: 02

Expert Talk (To be Delivered by SMEs from Industries) CO's/ PO's / PSO's

1. Agile Software Development in the Real World: Challenges and Best Practices" Industry adoption of Agile, Scrum, and DevOps with practical case studies.CO1–CO5, PO1,PO2,PO3,PO12,PSO1,PSO2
2. Software Architecture and Design Patterns for Scalable Applications" Focus on modern design principles used in scalable and maintainable software CO1–CO5, PO1, PO2,PO3,PO12,PSO1,PSO2

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering aPractitioner's Approach, Roger S. Pressman, 9th Edition, Mc Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
3. Software Project Management” by Bob Hughes, Mike Cotterell, and Rajib Mall.
4. Managing Software Projects” by Gopaldaswamy Ramesh.

Web References:

1. <https://www.computer.org/education/bodies-of-knowledge/software-engineering>
2. <https://www.geeksforgeeks.org/software-engineering/>
3. <https://www.pmi.org/pmbok-guide-standards>
4. https://www.mindtools.com/pages/main/newMN_PPM.htm

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35%	--
L2	40%	--
L3	25%	25%
L4	--	35%
L5	--	25%
L6	--	15%
Total (%)	100%	100%

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is the purpose of software engineering?
2. Define the term "software life cycle."
3. List the characteristics of a software product.
4. Name any four software life cycle models.
5. What is the waterfall model?
6. What is the purpose of a Software Requirements Specification (SRS)?
7. Define formal system specification.
8. What is a data dictionary?
9. List the types of formal specification techniques.
10. What are Petri Nets used for in system modeling?

L2: Understand

1. Explain the difference between software product and software process.
2. Describe how exploratory software development differs from planned development.

3. Compare traditional software development with agile development.
4. Explain why customization of the software process is important.
5. Describe the evolution of software engineering.
6. Explain the role of requirements gathering in software development.
7. Describe the characteristics of a good SRS document.
8. Differentiate between axiomatic and algebraic specifications.
9. Explain how Petri Nets represent system behavior.
10. Describe the use of 4GLs in executable specifications.
11. Explain the importance of architectural design in software engineering.
12. Describe the purpose of design heuristics in the design process.
13. Discuss the role of data flow diagrams in architectural mapping.
14. Differentiate between class-based and traditional components.
15. Explain the concept of interface analysis in UI design.

L3: Apply

1. Write a simple example of an algebraic specification for a stack ADT.
2. Apply Petri Nets to model a simple producer-consumer system.
3. Develop a basic data dictionary for a student registration system.
4. Create an SRS for a library management system.
5. Use axiomatic specification to define a simple banking operation.
6. Apply design heuristics to improve an existing system design.
7. Use a data flow diagram to map the architecture of a student management system.
8. Develop a simple interface design for an online bookstore.
9. Design class-based components for a ride-sharing application.
10. Implement a basic architectural model using a layered style.

L4: Analysing

1. Analyze the relationship between software architecture and system performance.
2. Compare and contrast different architectural styles (e.g., layered vs. client-server).
3. Identify the strengths and weaknesses of traditional components in modern software systems.
4. Break down the steps involved in interface design for a mobile application.
5. Examine the impact of poor design heuristics on software maintainability.
6. Analyze the impact of poor documentation on software maintenance.
7. Compare white-box testing and black-box testing with respect to coverage.
8. Differentiate between reverse engineering and reengineering.
9. Examine how object-oriented features affect testing strategies.

10. Identify and explain common issues associated with testing large-scale systems.
11. Analyze the relationship between software maintenance and configuration management.
12. Compare re-engineering and reverse engineering processes.
13. Differentiate between corrective, adaptive, and perfective maintenance.
14. Identify the barriers to software reuse in large organizations.
15. Examine the role of SQA in reducing post-deployment defects.

L5: Evaluating

1. Evaluate the effectiveness of smoke testing in agile environments.
2. Justify the use of program analysis tools in improving code quality.
3. Assess the benefits of incorporating refactoring in continuous integration.
4. Critically evaluate various testing strategies for object-oriented software.
5. Review and recommend practices for effective debugging in real-time applications.
6. Evaluate the effectiveness of SCM tools in large-scale software projects.
7. Justify the need for strict change control in mission-critical applications.
8. Assess the impact of poor version control on project success.
9. Critique the current industry practices on software reuse.
10. Evaluate a project tracking system in terms of its ability to meet deadlines and budget.

L6: Creating

1. Design a comprehensive test plan for an e-commerce application.
2. Create a code review checklist tailored for a distributed development team.
3. Develop a test suite for validating a banking application using black-box techniques.
4. Construct a model that integrates reverse engineering and forward engineering in legacy system migration.
5. Propose a set of guidelines for testing object-oriented programs effectively.
6. Design a version control and change management strategy for a distributed team.
7. Develop an SQA plan for a medical software application.
8. Create a reuse strategy for an educational software company.
9. Construct a project management schedule using available software tools for a mid-sized web application.
10. Propose an organization-wide reuse program, addressing its key challenges and success factors.

**Chairperson
Board of Studies (CSE)**

R24CSPC07 Object Oriented Programming using JAVA Lab 0 0 3 1.5
 (Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

1. To develop skills to design and analyze the applications with respect to java programming.
2. To strengthen the ability to identify and apply the suitable object oriented concept for the given real world problem.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs						BT
		PO1	PO2	PO3	PO5	PS01	PS02	
R24CSPC06.1	Write basic Java programs using arrays, control structures, and demonstrate the use of classes and objects.	3	2	2	1	3	2	L2, L3
R24CSPC06.2	Apply object-oriented concepts such as inheritance and packages, and implement applications using built-in and user-defined packages.	2	2	3	3	2	3	L3, L4
R24CSPC06.3	Develop Java applications with exception handling, multithreading, graphical user interfaces, and basic networking.	2	2	3	3	1	3	L3, L4, L6

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:

1. a) Implement the following programs using command line arguments and Scanner class
 - i) Accept two strings from the user and print it on console with concatenation of “and” in the middle of the strings. **CO’s-CO1**
 - ii) To find the perimeter and area of a circle given a value of radius. **CO’s-CO1**
- b) Write a program using classes and objects in java? **CO’s-CO1**
2. a) Write a program to call default constructor first and then any other constructor in the class? **CO’s-CO1**
- b) Write a program that accepts an array of integers and print those which are both odd and prime. If no such element in that array print “Not found”. **CO’s-CO1**

- c) Write a program to accept contents into an Integer Array and print the frequency of each number in the order of their number of occurrences. **CO's-CO1**
- d) Write a program that accepts an 'm x n' double dimension array, where 'm' represents financial years and 'n' represents Ids of the items sold. Each element in the array represents number of items sold in a particular year. Identify the year and id of the item which has more demand. **CO's-CO2**
3. a) Create a class Box that uses a parameterized constructor to initialize the dimensions of a box. The dimensions of the Box are width, height, depth. The class should have a method that can return the volume of the box. Create an object of the Box class and test the functionalities. **CO's-CO2**
- b) Create a new class called Calculator with the following methods:
A static method called power Int(int num1,int num2) This method should return num1 to the power num2.
A static method called power Double(double num1,double num2). This method should return num1 to the power num2.
Invoke both the methods and test the functionality. Also count the number of objects created. **CO's-CO2**
4. a) Accept a String and a number 'n' from user. Divide the given string into substrings each of size 'n' and sort them lexicographically. **CO's-CO1**
- b) Accept an array of strings and display the number of vowels and consonants occurred in each string. **CO's-CO1**
- c) Accept two strings from the user and determine if the strings are anagrams or not. **CO's-CO1**
5. a) Create a multilevel inheritance for classes vehicle, brand and cost. The vehicle class determines the type of vehicle which is inherited by the class brand which determines the brand of the vehicle. Brand class is inherited by cost class, which tells about the cost of the vehicle. Create another class which calls the constructor of cost class and method that displays the total vehicle information from the attributes available in the super classes. **CO's-CO2**
- b) Create an inheritance hierarchy of Figure_3D, Cylinder, Cone, Sphere etc. In the base class provides methods that are common to all Figure_3Ds and override these in the derived classes to perform different behaviors, depending on the specific type of Figure_3D. Create an array of Figure_3D, fill it with different specific types of Figure_3Ds and call your base class methods. **CO's-CO2**
6. a) Design a package to contain the class Student that contains data members such as name,

roll number and another package contains the interface Sports which contains some sports information. Import these two packages in a package called Report which process both Student and Sport and give the report. **CO's-CO2**

b) Write a program that accepts values of different data types and convert them to corresponding wrapper classes and display using the vector. **CO's-CO1**

7. a) Write a program to generate a set of random numbers between two numbers x1 and x2, and $x1 > 0$. **CO's-CO1**

b) Write a program to implement a new Array List class. It should contain add(), get(), remove(), size() methods. Use dynamic array logic. **CO's-CO2**

c) Create an employee class containing at least 3 details along with Id, setters, and getters. Insert the employee objects dynamically key as employee id and value as its corresponding object into a HashMap. Perform Id based search operation on the HashMap. **CO's-CO2**

8. a) Write a program that reads file name from the user then displays information about that file, also read the contents from the file in byte stream to count the number of alphabets, numeric values, and special symbols. Write these statistics into another file using byte streams **CO's-CO3**

b) Write a program that reads a CSV file containing a super market data containing product ID, Name, Cost and Quantity of sales and calculate the total revenue of the supermarket also sort the products in the order of their demand. **CO's-CO3**

c) Write a program that reads a text file containing some technical content and identify the technical terms and sort them alphabetically. **CO's-CO3**

Note: use a file containing stop words (general English and Grammar terms as many as possible)

9. a) Write a program that reads two numbers from the user to perform integer division into Num1 and Num2 variables. The division of Num1 and Num2 is displayed if they are integers. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception.

CO's-CO3

b) Create a user defined exception. **CO's-CO3**

10. a) Write a program that creates 3 threads by extending the Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2seconds and the third displays "Welcome" every 3 seconds. (Repeat the same by implementing Runnable). **CO's-CO3**

b) Write a program to illustrate Thread synchronization. **CO's-CO3**

11. a) Create a JApplet that displays a message which is scrolling from left to right. **CO's-CO3**

- b) Write a program that displays a sample registration page using Swing controls use appropriate layout managers. **CO's-CO3**
- c) Write a program for handling mouse events with adapter classes. **CO's-CO3**
12. a) Create an interface containing 3 radio buttons named line, rectangle, and oval. Based on the radio button selected, allow user to draw lines, rectangles, or ovals as per the locations selected by the user. **CO's-CO3**
- b) Write a program to create a Table inside a JFrame. **CO's-CO3**
- c) Create an interface that illustrates JFile Chooser class and read CSV file containing employee data of various departments and display the records department wise on the interface. **CO's-CO3**
13. a) Check all the fields filled or not, display success dialogue if all fields are filled with the help of ActionListener for program **CO's-CO3**
- b) Display respective error dialogue if a field is empty. **CO's-CO3**
14. Write a program to create three JSliders where each represents colors RED, GREEN and BLUE. Each slider has a value from 0 to 255. The background color of the applet is set based on the values retrieved from each slider to form a color using the color class constructor. On sliding any slider, the background color of the applet changes. **CO's-CO3**
15. Complete the code to develop an ADVANCED CALCULATOR that emulates all the functions of the GUI Calculator as shown in the image. **CO's-CO3**
16. Write a program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console.
- For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. **CO's-CO3**

Case Studies:

1. Grading a Multiple-choice Test for students **CO's-CO1**
2. Create a Person class containing basic details like Name, Gender, Mobile and Email. Based on that create and manage the objects that are related to student and employee Classes **CO's-CO2**
3. Create a package called Banking containing classes and interfaces related to various banking operations such as withdrawal, deposits, loans and insurance etc. Create two classes related to any two specific banks that uses this package. **CO's-CO2**
4. Write a program that implements simple chat application using GUI. **CO's-CO3**

Textbooks:

1. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019
2. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson Education, 2009.

Reference Books:

1. Cay S. Horstmann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition, Pearson, 2015.
3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006

Web References:

1. https://onlinecourses.nptel.ac.in/noc19_cs84/preview

**Chairperson
Board of Studies(CSE)**

Course Objectives:

The course aims to give students hands – on experience and train them to learn Design and Analysis Algorithm

1. To learn Prolog Program.
2. To Implement in prolog and its working environment
3. To Implement N-Queen problem and puzzle problem using Prolog
4. To Analyze the problem using BFS and DFS algorithm

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

Course Code	Course Outcomes	Mapping with POs and PSOs							Dok
		PO 1	PO 2	PO 3	PO 5	PS O1	PS O2	PS O3	
R24CSPC07.1	Understand of algorithm design techniques applicable across various tasks and ensuring foundational knowledge in problem-solving.	3	3	3	3	3	3	3	L3
R24CSPC07.2	Implement of the algorithms in both sequential and parallel paradigms, relevant for weeks involving sorting and graph traversals.	3	3	3	1	3	3	3	L4
R24CSPC07.3	Handle real-world computational problems.	3	3	3	1	3	3	3	L5

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, May, 2025

Approved in ACM No: 02

Developing the following programs**Week 1:**

1. Sort a given set of elements using the Quick sort method and analyze time. **CO's-CO1**

Week 2:

2. Implement a parallelized Merge Sort algorithm using Open MPI **CO's-CO1**

Week 3:

3. Implement binary tree traversal techniques using recursion and non-recursion. **CO's-CO1**

Week 4:

4. Print reachable nodes in a digraph using BFS; check connectivity using DFS. **CO's-CO2**

Week 5:

5. Determine articulation points and biconnected components in a graph. **CO's-CO2**

Week 6:

6. Implement algorithms for minimum cost spanning trees (Prim's and Kruskal's). **CO's-CO2**

Week 7:

7. Find shortest paths in a graph using Dijkstra's algorithm; study of Prolog.CO's-CO2

Week 8:

8. Write simple fact for the statements using prologCO's-CO2

Week 9:

9. Write predicates one convert's centigrade temperature to Fahrenheit,other checks if a temperature is below freezing.CO's-CO2

Week 10:

10. Write a program to solve 4-Queen problem.CO's-CO3

Week 11:

11. Write a program to solve 8-puzzle problem.CO's-CO3

Week 12:

12. Write a program to solve any problem using Breadth First Search.CO's-CO3

Week 13:

13. Write a program to solve any problem Depth First SearchCO's-CO3

Week 14:

14. Write a program to solve Travelling salesman ProblemCO's-CO3

Week 15 and Week 16:

15. Develop a tool to analyze social networks using graph algorithms (e.g., finding the shortest path, clustering, and connectivity). Input social network data and visualize the connections.

CO's-CO3

Text Books:

1. AnanyLevitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Computer Algorithms/ C++", Second Edition, Universities Press, 2007

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and CliffordStein, "Introduction to Algorithms", Third Edition, PHI Learning PrivateLimited, 2012.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press,2014

Web References:

1. <https://nptel.ac.in/courses/106101060>
2. https://www.cse.iitm.ac.in/course_details.php?arg=OTI
3. https://swayam.gov.in/nd1_noc19_cs47/previ

**Chairperson
Board of Studies (CSE)**

R24CSSC01**Full Stack Development****0 0 3 1.5****Course Objectives:**

1. To implement the client side of the web application using java-script.
2. To understand Java script on the desktop using Node JS.
3. To develop a web application using Node JS and Express.
4. To implement a SPA using React.
5. To develop a full stack single page application using React ,Node JS, and a Database (MongoDB or SQL).

Course Outcomes

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

Mapping with POs and PSOs																BT
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
24SC02.1	Implement the client side of the web application.	2	1	2	1	3	1	0	2	3	2	1	-	2	1	L1, L2
24SC02.2	Deploy server side applications using NodeJS. Use express framework in web development	2	2	3	2	3	1	0	2	2	2	3	-	3	2	L2, L3
24SC02.3	Implement and architect database systems. Develop a full stack single page applications	3	2	3	2	3	2	1	3	3	3	3	-	3	2	L2, L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Experiments covering the Topics:

- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML5 and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript-internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events

Developing the following programs:

Week-1:

1. Write a HTML program, to explain the working of lists. **CO's:CO1**
2. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes. **CO's:CO1**
3. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles. **CO's:CO1**
4. To create Form validation using JavaScript. **CO's:CO1**

Week-2:

1. Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan) **CO's:CO2**
2. Write a HTML program, to explain the working of forms by designing Registration form. **CO's:CO2**
3. Get data using Fetch API from an open-source end point and display the contents in the form of a card. **CO's:CO2**

Week-3:

1. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, <div>, tags. **CO's:CO2**
2. Write a HTML program, to embed audio and video into HTML webpage. **CO's:CO2**
3. Write a program to apply different types (or levels of styles or style specification formats) inline, internal, external styles to HTML elements. (identify selector, property and value). **CO's:CO2**

Week-4:

1. Write a program to apply different types of selector forms
 - i. Simple selector (element, id, class, group, universal)
 - ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
 - iii. Pseudo-class selector
 - iv. Pseudo-element selector
 - v. Attribute selector **CO's:CO2**

Week-5:

1. Write a program to demonstrate the various ways you can reference a color in CSS. **CO's:CO2**
2. Write a CSS rule that places a background image half way down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.

CO's:CO2

3. Write a program using the following terms related to CSS font and text:
 - i. font-size ii. font-weight iii. font-style
 - iv. text-decoration v. text-transformation vi. text-alignment**CO's: CO2**
4. Write a program, to explain the importance of CSS Box model using
 - i. Content ii. Border iii. Margin iv. Padding**CO's: CO2**

Week-6:

1. Write a program to embed internal and external Java Script in a webpage. **CO's: CO3**
2. Write a program to explain the different ways for taking input & displaying output..
CO's: CO3
3. Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not
CO's: CO3

Week-7:

1. Write a program using document object properties and methods.
2. Write a program using window object properties and methods.
3. Write a program using array object properties and methods.
4. Write a program using math object properties and methods.
5. Write a program using string object properties and methods. **CO's: CO3**

Week-8:

1. Write a program to display week days using switch case.
2. Write a program to print 1 to 10 numbers using for, while and do-while loops.
3. Write a program to print data in object using for-in, for-each and for-of loops. Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not.
4. Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1-10's, 1-2's & 1-1's) **CO's: CO3**

Week-9:

1. Design a appropriate function should be called to display
 - I. Factorial of that number
 - II. Fibonacci series up to that number
 - III. Prime numbers up to that number
 - IV. Is it palindrome or not**CO's: CO3**

Week-10:

1. Design a HTML having a textbox and four buttons named Factorial, Fibonacci, Prime, and

Palindrome. When a button is pressed an appropriate function should be called to display

I. Factorial of that number

II. Fibonacci series up to that number

III. Prime numbers up to that number

IV. Is it palindrome or not

CO's:CO3

Week-11:

1. Write a program to validate the following fields in a registration page

I. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)

II. Mobile(onlynumbersandlength10digits)

III. E-mail(shouldcontainformatlike xxxxxxx@xxxxxx.xxx).

CO's:CO3

Week-12:

1. Create a form and validate the contents of the form using JavaScript. CO's: CO3

2. Get data using Fetch API from an open-source end point and display the contents in the form of a card.. CO's: CO3

3. Create a Node JS server that serves static HTML and CSS files to the user without using Express. CO's: CO3

Week-13:

1. Create a Node JS server using Express that stores data from a form as a JSON file and displays it in another page. The redirect page should be prepared using Handlebars.

CO's:CO3

2. Create a Node JS server using Express that creates, reads, updates and deletes students' details and stores them in MongoDB database. The information about the user should be obtained from a HTML form. Conversion of Fahrenheit to Celsius and vice versa.

CO's:CO3

Week-14:

1. Create a Node JS server that creates, reads, updates and deletes event details and stores them in a MySQL database. The information about the user should be obtained from a HTML form.

CO's:CO3

2. Create a counter using React JS

CO's:CO3

3. Create a To do application using ReactJS. Store the data to a JSON file using a simple Node JSserver and retrieve the information from the same during page reloads. CO's:CO3

Week-15:

1. Create a simple Signup and Login mechanism and authenticate the user using cookies. The user information can be stored in either MongoDB or MySQL and the server should be built using Node JS and Express Framework. **CO's:CO3**

2. Create and deploy a virtual machine using a virtual box that can be accessed from the host computer using SSH. Write a C program to print last n characters of a given file. **CO's:CO3**

3. Create a docker container that will deploy a Node JSping server using the Node JS image. **CO's:CO3**

Textbooks:

1. JohnDean, WebProgrammingwithHTML5, CSSandJavaScript,Jones&Bartlett Learning, 2019.

Reference Books:

1. Programming the World WideWeb, 7thEdition, RobertWSebesta, Pearson, 2013.
2. ProMERNStack: FullStackWebAppDevelopmentwithMongo, Express, React, and Node, Vasam Subramanian, 2nd edition, APress, O'Reilly.

Online Learning Resources:

1. <https://www.w3schools.com/html>
2. <https://www.w3schools.com/css>
3. <https://www.w3schools.com/js/>
4. <https://www.w3schools.com/nodejs>
5. <https://www.w3schools.com/typescript>

**Chairperson
Board of Studies (CSE)**

Course Title: SOFT SKILLS & VERBAL ABILITY	CourseCode: R24CSHS05
Teaching Scheme(L:T:P):0:1:2	Credits: 02
Type of Course: Tutorials + practical's	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Prerequisites: To succeed in Soft Skills and Verbal ability, students should have a basic understanding of effective communication principles, including verbal and non-verbal communication. Familiarity with teamwork and collaboration techniques is essential for group activities and presentations. A positive attitude and willingness to receive feedback help in personal development. Time management and problem-solving abilities support efficient task handling. Basic proficiency in English and presentation tools also enhances overall performance in soft skills training	

Course objective:

1. Enhance proficiency in English grammar, vocabulary, and reasoning skills for recruitment exams.
2. Develop effective communication skills for group discussions, resume building, and interviews.
3. Equip students with techniques for reading comprehension, logical reasoning, and professional presentation.
4. Prepare students for successful career placements through improved language and soft skills.

Course Outcomes:

At the end of this course the student will be able to

Course Outcomes	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 8	PO 9	PO1 0	PO1 1	PSO 1	BT LEVEL
CO1: Demonstrate strong command over English grammar, vocabulary, and reasoning skills	3	3	2	2	1	1	1	1	2	1		L3
CO2: Effectively communicate in group discussions, create impactful resumes, and excel in interviews	2	1	1	1	2	2	2	3	3	2		L4
CO3: Develop critical thinking	3	3	2	2	1	1	1	1	2	1		L3

and problem-solving abilities for recruitment exams												
CO4: Be well-prepared for career placements with enhanced professional communication and soft skills	2	2	1	1	2	2	2	3	3	2		L3

Unit I – English Grammar and Usage (10 Hours)

This unit focuses on core grammar concepts frequently tested in company recruitment exams. Topics include:

Parts of Speech, Tenses and Subject-Verb Agreement, Articles and Prepositions, Sentence Correction and Spotting Errors, Active and Passive Voice, Direct and Indirect Speech

Learning Outcome: Students will demonstrate accurate grammar usage and error detection skills in various sentence structures.

Vocabulary Development and Application (10 Hours)

This unit enhances vocabulary required for business communication and aptitude tests. Topics include:

Synonyms and Antonyms, One-word Substitution, Idioms and Phrases, Confusing Word Pairs, Phrasal Verbs and Collocations

Learning Outcome: Students will improve their vocabulary strength and apply words appropriately in verbal and written contexts.

Unit II – Reading Comprehension Skills (5 Hours)

Students will learn techniques to understand, interpret, and analyze passages. Focus areas:

Main Idea and Supporting Details, Inference-Based Questions, Vocabulary in Context, Tone and Author’s Perspective

Learning Outcome: Students will effectively comprehend and answer questions based on unseen passages within time constraints.

Verbal Reasoning and Logic-Based Language Skills (5 Hours)

This unit covers logical verbal questions commonly seen in recruitment exams:

Sentence Completion, Cloze Tests, Para Jumbles / Sentence Rearrangement, Statement and Conclusion / Assumptions.

Learning Outcome: Students will develop reasoning skills to solve pattern-based language puzzles.

Unit III – Group Discussion Skills (10 Hours):

This unit develops students' ability to communicate effectively in a group setting. It includes understanding the GD format, evaluation criteria, and participation strategies. Sessions will train students on body language, tone modulation, handling abstract and controversial topics, and presenting logical arguments. Multiple GD simulations will be conducted with personalized feedback to improve spontaneity and structure in speaking.

Learning Outcome: students will be able to communicate their ideas clearly, listen actively, contribute effectively to discussions, and demonstrate leadership and teamwork while maintaining professionalism and respect for diverse opinions.

Unit IV – Resume Preparation and Personal Branding (10 Hours):

This unit guides students in preparing an impactful, professional resume suited for technology and consulting sectors. Key areas include formatting, project and internship presentation, using effective language, and highlighting strengths and certifications. Students will also learn to optimize their LinkedIn profiles and online presence to reflect a professional digital identity.

Learning Outcome: Students will be able to create a professional, well-structured resume that highlights their skills and experiences, and build a strong personal brand to effectively present them in the job market

Unit V – Interview Preparation (10 Hours):

This unit addresses all aspects of interview readiness. It covers commonly asked HR and technical questions, behavioral questions using the STAR (Situation, Task, Action, Result) method, and communication strategies during online interviews. Students will receive training in grooming, attire, voice modulation, and confidence building

Learning Outcome: students will be able to confidently handle both technical and HR interviews, presenting themselves professionally and effectively communicating their skills and experiences

Text Books:

1. **Wren, P. C., and H. Martin,***High School English Grammar and Composition*, S. Chand Publishing, 1990.
2. **Lewis, N.,***Word Power Made Easy*, Goyal Publishers, 1993.

3. **Aggarwal, R. S.**, *A Modern Approach to Verbal & Non-Verbal Reasoning*, S. Chand Publishing, 2017.
4. **Bakshi, S. P.**, *Objective General English*, Arihant Publications, 2018.

E-Resources :

1. **Grammarly**, *AI-powered writing assistant*, Grammarly, <https://www.grammarly.com/>
2. **IndiaBIX**, *Online Aptitude & Reasoning Practice*, IndiaBIX, <https://www.indiabix.com/>.
3. **AmbitionBox**, *Interview Experiences and Reviews*, AmbitionBox, <https://www.ambitionbox.com/>.
4. **Canva**, *Online Resume Builder and Templates*, Canva, <https://www.canva.com/resumes/templates>
5. **Testbook**, *Testbook: Online Mock Tests and Practice Papers*, <https://testbook.com/>.

**Chairperson
Board of Studies (CSE)**

Course Title: Indian Traditional Knowledge	CourseCode:R24MC04
Teaching Scheme(L:T:P): 2:0:0	Credits: -
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: To succeed in Indian Traditional Knowledge, students should have a basic understanding of cultural heritage, environmental sustainability, and indigenous practices. Familiarity with legal and constitutional frameworks, especially related to biodiversity and forest rights, is essential.	

Course Objectives:

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

- The course aim of the importing basic principle of third process reasoning and inference sustainability is at the course of Indian traditional knowledge system.
- To understand the legal framework and traditional knowledge and biological diversity act 2002 and geographical indication act 2003.
- The courses focus on traditional knowledge and intellectual property mechanism of traditional knowledge and protection.
- To know the student traditional knowledge in different sector.

Course Outcomes:

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

Course Outcomes	PO1	PO2	PO6	PO7	PO8	PSO1	BT LEVEL
CO1: Understand the concept of Traditional knowledge and its importance	2	0	2	2	2		L2
CO2: Know the need and importance of protecting traditional knowledge	2	2	3	2	3		L2
CO3: Know the various enactments related to the protection of traditional knowledge	2	2	3	2	3		L1
CO4: Understand the concepts of Intellectual property to protect traditional knowledge	3	2	2	2	2		L2

UNIT-I

Introduction to traditional knowledge: Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge

systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge.

COs- CO1

UNIT-II

Protection of traditional knowledge: the need for protecting traditional knowledge Significance of TK Protection, value of TK in global economy, Role of Government to harness TK.

COs- CO1, CO2

UNIT-III

Legal framework and TK: A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmers Rights Act, 2001 (PPVFR Act);B:The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledge bill, 2016. Geographical indications act 2003.

COs- CO2, CO3

UNIT-IV

Traditional knowledge and intellectual property: Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge.

COs- CO3, CO4

UNIT-V

Traditional knowledge in different sectors: Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of TK.

COs- CO3, CO4

Reference Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and

Vipin Kumar Singh, Pratibha Prakashan 2012.

3. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002
4. "Knowledge Traditions and Practices of India" Kapil Kapoor, Michel Danino

e-Resources:

1. <https://www.youtube.com/watch?v=LZP1StpYEPM>
2. <http://nptel.ac.in/courses/121106003/>

**Chairperson
Board of Studies (CSE)**



AVANTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113. www.avanthienggcollege.ac.in,
mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

II Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	BS	R24CSBS07	Probability & Statistics	3	0	0	3
2	PC	R24CSPC09	Formal Languages and Automata Theory	3	0	0	3
3	PC	R24CSPC10	Database Management Systems	3	0	0	3
4	HS	R24HS03	Universal Human Values-Understanding Harmony & Human Ethical Conduct	2	0	0	2
5	PC	R24CSPC11	Operating Systems	3	0	0	3
6	PC	R24CSPC12	Database Management Systems Lab	0	0	3	1.5
7	PC	R24CSPC13	Operating Systems Lab	0	0	3	1.5
8	PC	R24CSPC14	Advanced Coding Lab-I	0	0	3	1.5
9	SC	R24CSSC02	Python Programming	0	1	2	2
10	HS	R24HS04	Quantitative aptitude & Logical Reasoning	0	0	2	1
11	MC	R24MC03	Environmental Science	2	0	0	-
Total				16	01	13	21.5
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation							

Category	Courses	Credits
BS- Basic Sciences Course	1	03
PC-Proficitional Core Course	6	13.5
SC- Skill Oriented Course	1	2
HS- Humanity Science and Management Course	2	03
MC-Mandatory Course	1	--
Total	11	21.5

**Chairperson
Board of Studies (CSE)**

R24CSPC09

Formal Languages and Automata Theory

3 0 0 3

(Common to CSE,CSE (DS),CSE (AI&ML))

Course Objectives:

The course is designed with the objective to:

- Introduce languages, grammars, and computational models
- Understand the foundational principles of Automata Theory & Analyze and design Finite Automata (FA)
- Formulate Regular Expressions and Grammars and demonstrate their equivalence to Finite Automata
- Discussing regular expressions and regular languages
- Construct Context Free Grammars (CFG) to describe context-free languages, and simplify and normalize these grammars into CNF and GNF.
- Design and analyze Pushdown Automata (PDA) to recognize CFL'S and understand their relationship with CFG
- Explain the capabilities and limitations of Turing Machines & Demonstrate decidability and undecidability for NP Hard problems

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs						Dok
		PO 1	PO 2	PO 3	PO 4	PS0 1	PS0 2	
R24CSPC08.1	Comprehensible to understand the concept of abstract machine and their ability to recognize languages based on their computational power and the types of languages they can process	3	3	2	--	3	2	L1, L2
R24CSPC08.2	Understand regular expressions and finite automata	3	3	3	--	3	2	L1, L2 L3
R24CSPC08.3	Develop context free grammars for formal languages.	3	3	3	--	2	3	L1, L2, L3
R24CSPC08.4	Design pushdown automata for context free grammars	3	3	3	--	3	2	L4
R24CSPC08.5	Design Turing machine and Formulate decidability and un-decidability problems and P&NP problems	3	3	3	2	3	3	L4, L5,L6

SYLLABUS

UNIT-I:Finite Automata:

14 hours

Need of Automata theory, Central Concepts of Automata Theory, Automation, FiniteAutomation, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA,Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with ϵ -Transitions,Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applicationsand Limitation of Finite Automata.

Self Learning Concepts: Research how DFAs/NFAs are used in lexical analyzer, Network protocol Design & Control system. **CO's-CO1**

UNIT-II:Regular Expressions

12 hours

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Inter Conversion, Equivalence between FA and RE, Pumping Lemmaof Regular Sets, Grammars, Classification of Grammars, ChomskyHierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, InterConversion. **CO's-CO2**

Self Learning Concepts: Closure properties of Regular set

UNIT-III:Context Free Grammar

10 hours

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, AmbiguousGrammars, Simplification of Context Free Grammars, Normal Forms-CNF and GNF, Pumping Lemma CFG, Applications of Context Free Grammars. **COs-CO3**

Self Learning Concepts: Elimination of left recursion & left factoring

UNIT-IV:Pushdown Automata

12 hours

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, LanguageAcceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non -Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars,Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata. **CO's-CO4**

Self Learning Concepts: Investigate practical applications of PDAs in parsing such as in compiler & syntax analysis.

UNIT-V: Turning Machine

14 hours

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, TransitionTables and Transition Diagrams, Language Acceptanceof a TM, Design of TMs, Types of TMs,UniversalTM,Decidable and Un-decidable Problems, Halting Problem of TMs, Post'sCorrespondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems. **CO's-CO5**

Self Learning Concepts: Church's Thesis for Turning machine.

Board of Studies:Computer Science and Engineering

Approved in BoS No: 02,--April,2025

Approved in ACM No: 02

Text Books:

- 1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rdEdition, Pearson, 2008.
- 2) Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N.Chandrasekharan, 3rdEdition, PHI, 2007.

ReferenceBooks:

- 1) Elements of Theory of Computation, Lewis H.P. &Papadimition C.H., Pearson /PHI
- 2) Theory of Computation, V. Kulkarni, Oxford University Press, 2013.
- 3) Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014

Web References:

- 1) <https://nptel.ac.in/courses/106103070>
- 2) <https://nptel.ac.in/courses/106106049>

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35%	--
L2	40%	--
L3	25%	15%
L4	--	35%
L5	--	35%
L6	--	15%
Total (%)	100%	100%

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

- 1) What is Automata Theory and why is it needed?
- 2) What are Mealy and Moore Machines?
- 3) Define Regular Expressions and Regular Sets.
- 4) List the Identity Rules for Regular Expressions.
- 5) Define Chomsky Normal Form (CNF) and Greibach Normal Form (GNF).
- 6) State the Pumping Lemma for CFG.
- 7) What is an Instantaneous Description of a PDA?
- 8) Differentiate between Deterministic and Non-deterministic PDA.
- 9) Define a Turing Machine..

10) Define Decidable and Undecidable Problems

L2: Understand

- 1) Describe the differences between a DFA and an NFA.
- 2) Explain the significance of minimizing Finite Automata.
- 3) Explain the equivalence of two Regular Expressions.
- 4) Describe the inter-conversion between Finite Automata and Regular Expressions.
- 5) Explain how Parse Trees represent the derivation of a string.
- 6) Describe the steps involved in simplifying Context Free Grammars.
- 7) Describe the relationship between PDA and Context Free Grammars.
- 8) Explain the process of converting a CFG to a PDA and vice versa.
- 9) Describe the different types of Turing Machines.
- 10) Describe the classes of P and NP.

L3: Apply

- 1) Design a DFA to accept strings with a specific pattern (e.g., strings containing "ab").
- 2) Construct a Mealy Machine to perform a specific sequence transformation.
- 3) Convert a given Finite Automaton to a Regular Expression.
- 4) Use the Pumping Lemma to prove whether a given set is regular or not.
- 5) Convert a given CFG to CNF. And Convert a given CFG to GNF.
- 6) Use the Pumping Lemma to prove whether a language is Context Free or no
- 7) Design a PDA to accept a given Context Free Language.
- 8) Convert a given CFG to an equivalent PDA.
- 9) Design a Turing Machine to perform a specific computation (e.g., addition, subtraction).
- 10) Represent a Turing Machine using Instantaneous Descriptions.

L4: Analysing

- 1) Analyze the time and space complexity of DFA vs. NFA.
- 2) Differentiate between the capabilities and limitations of Finite Automata
- 3) Analyze the equivalence of different Regular Expressions.
- 4) Analyze the given grammar and classify it according to Chomsky Hierarchy.
- 5) Analyze a given grammar and determine if it is ambiguous.
- 6) Compare the advantages and disadvantages of CNF and GNF
- 7) Analyze the limitations of Pushdown Automata.
- 8) Compare the complexity of Deterministic and Non-deterministic PDA.
- 9) Design a Turing Machine to perform a specific computation (e.g., addition, subtraction).
- 10) Represent a Turing Machine using Instantaneous Descriptions.
- 11) Show that a given problem is decidable or undecidable.

L5: Evaluating

- 1) Evaluate whether a given language can be recognized by a Finite Automaton.
- 2) Compare different designs of Finite Automata for the same language in terms of efficiency.
- 3) Justify the need for NFA when DFA can recognize the same language.
- 4) Evaluate the limitations of Regular Expressions in expressing certain languages.
- 5) Determine whether a given language can be described by a Regular Expression.
- 6) Evaluate whether a given language can be generated by a Context Free Grammar.
- 7) Assess the impact of ambiguity in a grammar.
- 8) Evaluate whether a given language can be generated by a Context Free Grammar.
- 9) Assess the impact of ambiguity in a grammar.
- 10) Evaluate the impact of the Halting Problem on computation.
- 11) Assess the theoretical limits of computation.
- 12) Justify the importance of P vs. NP problem.

L6: Creating

- 1) Design a Context Free Grammar for a given language.
- 2) Develop a parser for a simple Context Free Grammar.
- 3) Design a Pushdown Automaton with additional features.
- 4) Develop an application using PDA for tasks like expression evaluation or syntax checking

**Chairperson
Board of Studies (CSE)**

R24CSPC10**DATABASE MANAGEMENT SYSTEMS****3 0 0 3****(Common to CSE,CSE (DS),CSE (AI&ML))****Course Objectives:**

- Train in the fundamental concepts of database management systems, database modelling and design, SQL, PL/SQL, and System implementation techniques.
- Enable students to model ER diagram for any customized applications.
- Provide knowledge on concurrency techniques.
- Understand normalization theory and apply such knowledge to the normalization of a database.
- To learn the principles of systematically designing and using large scale Database Management Systems for various applications.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PO11	PSO1	PSO2	
R24CSPC0 9.1	Understand basic concepts of database systems and data models.	3	2	-	1	2	2	3	-	L 1, L2
R24CSPC0 9.2	Design ER diagrams and apply conceptual modeling for real-world problems like Railway & Hospital systems.	3	3	3	1	2	2	3	1	L1, L2, L3
R24CSPC0 9.3	Apply relational model concepts and SQL operations to implement simple databases.	3	2	2	1	3	2	3	1	L 2, L3
R24CSPC0 9.4	Perform schema refinement using normalization techniques.	3	2	2	2	3	2	3	-	L3, L4
R24CSPC0 9.5	Explain transaction management concepts and implement concurrency control and recovery methods. Understand indexing techniques like B+ Trees and Hash-based indexing for efficient data retrieval.	3	3	2	3	3	2	3	-	L4, L5,L 6

SYLLABUS**UNIT I: 15 Hours**

Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; three tier schema architecture

for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams **CO's–CO1**

Self Learning Topics: Database Applications(Real-world applications: Banking, E-commerce, Healthcare, Education,and Social Media)

UNIT II:15 Hours

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus.

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).**CO's–CO2**

Self Learning Topics: Structure of a Relational Database

UNIT III:15 Hours

SQL: Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.**CO's–CO3**

Self-Learning Topics: Advanced Grouping and Aggregation (GROUP BY with ROLLUP, CUBE, and GROUPING SETS, Filtering groups using HAVING)

UNIT IV:10 Hours

Schema Refinement (Normalization):Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce Codd-normal form(BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).

CO's–CO4

Self-Learning Topics: Decomposition Deep Dive (Comparing Lossless vs. Lossy Decomposition,

Dependency-Preserving Decomposition: When to compromise)

UNIT V:10 Hours

Transaction Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Introduction to Indexing Techniques: B+ Trees, operations on B+ Trees, Hash Based Indexing

CO's–CO5

Self-Learning Topics: Locking & Deadlock Handling – Strategies for handling concurrent transactions.

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Text Books:

1. Introduction to Database Systems, CJ Date, Pearson.
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA Mc Graw Hill3rd Edition.

Reference Books:

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel7thEdition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education.
3. Database Systems - The Complete Book, H G Molina, J D Ullman, J Widom Pearson.
4. Data base System Concepts,5/e, Silberschatz, Korth, TMH.

Web References:

1. <http://www.c4learn.com/>
2. <http://www.geeksforgeeks.org/c/>
3. <http://nptel.ac.in/courses/122104019/>
4. <http://www.learn-c.org/>
5. <https://www.tutorialspoint.com/c-programming/>

**Chairperson
Board of Studies (CSE)**

Course Objectives:

The main objectives of the course is to

- To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
- To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.
- To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO 1	PO 2	PO 6	PO 7	PO 8	PS 09	PS 11	PSO1	
R24HS03.01	Define the terms like Natural Acceptance, Happiness and Prosperity.	1	2	2	3	--	2	3		L1, L2
R24HS03.02	Analyze the coexistence of the self and the body, distinguish between their needs, and apply practices that promote self regulation and overall well-being.	1	2	3	3	2	2	3		L4
R24HS03.03	Relate human values with human relationship and human society.	1	2	2	3	2	2	3		L4
R24HS03.04	Justify the need for universal human values and harmonious existence.	1	2	3	3	3	2	3		L5
R24HS03.05	Develop as socially and ecologically responsible engineers.	2	2	3	3	3	2	3		L3, L6

SYLLABUS**UNIT-I: Introduction to Value Education (6 lectures and 3 tutorials for practice session)**

Lecture 1: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Lecture 2: Understanding Value Education Tutorial 1: Practice Session PS1 Sharing about Oneself Lecture 3: self-exploration as the Process for Value Education Lecture4: Continuous Happiness and Prosperity – the Basic Human Aspirations Tutorial 2: Practice Session PS2 Exploring Human Consciousness

Lecture 5: Happiness and Prosperity – Current Scenario Lecture 6: Method to Fulfill the Basic Human Aspirations Tutorial 3: Practice Session PS3 Exploring Natural Acceptance **COs- CO1, CO2, CO3**

UNIT-II: Harmony in the Human Being (6 lectures and 3 tutorials for practice session)

Lecture 7: Understanding Human being as the Co-existence of the self and the body. Lecture 8: Distinguishing between the Needs of the self and the body Tutorial 4: Practice Session PS4 Exploring the difference of Needs of self and body. Lecture 9: The body as an Instrument of the self Lecture 10: Understanding Harmony in the self Tutorial 5: Practice Session PS5 Exploring Sources of Imagination in the self-Lecture 11: Harmony of the self with the body Lecture 12: Programme to ensure self-regulation and Health Tutorial 6: Practice Session PS6 Exploring Harmony of self with the body **COs- CO2, CO3**

UNIT-III: Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)

Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction Lecture 14: 'Trust' – the Foundational Value in Relationship Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust Lecture 15: 'Respect' – as the Right Evaluation Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect Lecture 16: Other Feelings, Justice in Human-to-Human Relationship Lecture 17: Understanding Harmony in the Society Lecture 18: Vision for the Universal Human Order Tutorial 9: Practice Session PS9 Exploring Systems to fulfil Human Goal **COs- CO2, CO4, CO5**

UNIT-IV: Harmony in the Nature/Existence (4 lectures and 2 tutorials for practice session)

Lecture 19: Understanding Harmony in the Nature Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature Lecture 21: Realizing Existence as Co-existence at All Levels Lecture 22: The Holistic Perception of Harmony in Existence Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence. **COs- CO2, CO5**

UNIT-V: Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)

Lecture 23: Natural Acceptance of Human Values Lecture 24: Definitiveness of (Ethical) Human Conduct Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order Lecture 26: Competence in Professional Ethics Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies Lecture 28: Strategies for Transition towards Value-based Life and Profession Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order. **COs- CO3, CO5**

Practice Sessions for UNIT I – Introduction to Value Education

PS1 Sharing about Oneself

PS2 Exploring Human Consciousness

PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being

PS4 Exploring the difference of Needs of self and body

PS5 Exploring Sources of Imagination in the self

PS6 Exploring Harmony of self with the body

Practice Sessions for UNIT III – Harmony in the Family and Society

PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect

PS9 Exploring Systems to fulfil Human Goal

Practice Sessions for UNIT IV – Harmony in the Nature (Existence)

PS10 Exploring the Four Orders of Nature

PS11 Exploring Co-existence in Existence

Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at Professional Ethics

PS12 Exploring Ethical Human Conduct

PS13 Exploring Humanistic Models in Education

PS14 Exploring Steps of Transition towards Universal Human Order

Readings:

Textbook and Teachers Manual

a. The Textbook

- R R Gaur, R Asthana, G P Bagaria, A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93 87034-47-1

b. The Teacher's Manual

- R R Gaur, R Asthana, G P Bagaria, Teachers' Manual for A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-53-2

Reference Books

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.

**Chairperson
Board of Studies (CSE)**

R24CSPC11**Operating Systems****3 0 0 3**

(Computer Science and Engineering)

Course Objectives:

1. To understand operating system overview, functions and services.
2. To analyze process concept, inter process communication and synchronization.
3. Illustrate different conditions for deadlock and their possible solutions.
4. To understand secondary storage management.
5. To understand protection and security.

Course Outcomes

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

Mapping with POs and PSOs															Dok
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSP C11.1	Describe operating system overview, functions, and services	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2
R24CSP C11.2	Analyze process concept, inter-process communication, and synchronization	3	3	-	2	3	-	-	2	2	-	2	3	2	L2, L3
R24CSP C11.3	Analyze process synchronization mechanisms and dead lock handling techniques.	3	3	-	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSP C11.4	Describe methods for mapping secondary storage management.	3	-	-	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSP C11.5	Evaluate protection mechanism and security	3	2	3	2	3	2	3	-	-	2	3	-	-	L5, L6

SYLLABUS**UNIT I:****15 Hours**

Operating System Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source.

System Structures: Operating System Services, system calls, Types of System Calls, system programs, Operating system structure, Operating system Design and Implementation, Building and Booting an Operating System, Operating system debugging.

CO's-CO1

Self-Learning Topics: Evaluation of Operating systems.

UNIT II:**15 Hours**

Processes: Process Concepts, Process states, Process scheduling, Operations on processes, Inter-process communication,

Threads and Concurrency: Multithreading models, Threads, Thread libraries, threading issues.

CPU Scheduling: Basic concepts, Scheduling-Criteria, CPU Scheduling Algorithms, Algorithms Evaluation, Thread Scheduling, Multiple processor scheduling. **CO's-CO2**

Self-Learning Topics: Multicore and Multithreading.

UNIT III: 15 Hours

Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Monitors, Classic Problems of Synchronization, Synchronization examples

Deadlock: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock. **CO's-CO3**

Self-Learning Topics: Mutual Exclusion: Hardware Support

UNIT IV: 15 Hours

Memory-Management Strategies: Introduction, Contiguous Memory Allocation, Paging, Page-Table Structure, Swapping, Segmentation.

Virtual Memory Management: Virtual Memory, Demand Paging, Copy-on-write, Page-Replacement Algorithms, Frames Allocation, Thrashing.

File System Interface: File Concepts, Access Methods, Directory Structure, File system Implementation: File-system structure, File-system Operations, Directory implementation, File System Mounting, File Sharing and Protection, Allocation Methods, Free-Space Management. **COs-CO4**

Self-Learning Topics: Memory Partitioning and Hardware Control Structures.

UNIT V: 10 Hours

Mass Storage: Overview Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, RAID Structure

Protection: Goals of Protection, Principles of Protection, Domain of Protection Access Matrix, Implementation of Access Matrix.

Security: Security Problem, Program Threats, System and Network Threats **CO's-CO5**
Self-Learning Topics: Disk Cache and I/O Buffering

Board of Studies : Computer Science and Engineering

Approved in BoS No: 02, --, April, 2025

Approved in ACM No: 02

Expert Talk (To be Delivered by SMEs from Industries) CO's/ PO's / PSO's

1. Operating Systems in Cloud Computing (Virtualization Techniques in Industry Cloud Platforms (AWS, Azure, GCP), Containerization (Docker, Kubernetes) vs Virtual Machines, Operating System Abstractions in Cloud Architecture) CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Real-Time Operating Systems (RTOS) in Industry Applications (RTOS in Automotive Systems (ADAS, AUTOSAR), RTOS for Industrial Automation and Robotics vs General Purpose OS in Embedded Systems) CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

3. Operating Systems in Cybersecurity (OS Hardening Techniques in Enterprises, Secure Boot and Trusted Execution Environments, Role of OS in Endpoint Security Solutions) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 10th Edition, John Wiley
2. Operating Systems- A Concept based Approach, D. M. Dhamdhere, 2nd Edition, Tata McGraw-Hill
3. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley

Reference Books:

1. Operating Systems-Internal and Design Principles, Stallings, 5th Edition, Pearson Education/PHI, 2005
2. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
3. UNIX: Concepts and Applications, Sumit Abha Das, 4th Edition, Tata McGraw-Hill
4. Operating System: A Design-Oriented Approach, Charles Patrick Crowley

Web References:

1. <https://archive.nptel.ac.in/courses/106/102/106102132/>
2. https://edurev.in/courses/23057_Operating-System
3. https://www.tutorialspoint.com/operating_system

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is an operating system (OS)?
2. What is meant by an operating system's "kernel"?
3. Define "System Programs" in the context of operating systems.
4. What are system calls in an operating system?
5. Define multithreading.
6. What are the basic criteria for CPU scheduling?
7. What is the critical-section problem in process synchronization?
8. What are mutex locks, and how do they function in process synchronization?
9. Define deadlock in a system.
10. What is a page table, and what role does it play in memory management?
11. What is segmentation in memory management?
12. What are the key components of a mass-storage structure?
13. Define the disk structure and its main components.

L2: Understand

1. Explain the role of the operating system in managing hardware resources.
2. Compare the difference between free and open-source operating systems and proprietary operating systems.
3. Describe the process of booting an operating system from start to finish.
4. Explain the different states that a process can be in and how processes transition between these states.
5. How does inter-process communication (IPC) help processes communicate and synchronize?
6. Explain the difference between a process and a thread.
7. Describe the benefits and challenges of multithreading in modern operating systems.
8. Describe how Peterson's solution works to ensure mutual exclusion.
9. How does deadlock occur in a system? Explain the deadlock characterization in terms of the four necessary conditions.
10. Describe the structure of a page table and its role in translating virtual addresses to physical addresses.
11. Explain how the file system structure is organized to store data efficiently.
12. Describe how the directory structure in a file system helps in managing files.
13. Describe how disk scheduling algorithms manage read/write operations and improve efficiency.

L3: Apply

1. In what way does an operating system structure affect its performance?
2. Demonstrate how system services manage the hardware components of a computer system.
3. How can an operating system be debugged to ensure it works correctly?
4. Given a scenario with multiple processes, how would you apply a CPU scheduling algorithm (e.g., FCFS, SJF) to allocate CPU time?
5. In a system with multiple threads, demonstrate how semaphores can be used to solve the Producer-Consumer problem.
6. Given a scenario of resource allocation, explain how you would detect deadlock using a resource allocation graph.
7. Given a scenario with file system fragmentation, how would you apply free-space management techniques to handle the issue?
8. Apply demand paging in a system with limited physical memory. How would you ensure that the page-replacement algorithm minimizes page faults?
9. Given a system with limited physical memory, demonstrate how paging can be used to manage the memory.
10. Apply a disk scheduling algorithm (e.g., FCFS, SSTF) to a series of disk I/O requests and explain the resulting order of operations.
11. Given a security policy, how would you implement the access matrix model to control access to resources?
12. Given a security threat (e.g., buffer overflow), apply appropriate measures to secure the system against this type of attack.

L4: Analysing

1. Analyze how the structure of an operating system (monolithic vs. microkernel) impacts system performance and security.
2. Analyze how computing environments (such as cloud-based and embedded systems) influence the design and functionality of an operating system.
3. Examine how the operating system handles process synchronization using inter-process communication (IPC) mechanisms like message passing and shared memory.
4. Compare and contrast the different CPU scheduling algorithms (e.g., Round Robin, Shortest Job First, and Priority Scheduling). What are their strengths and weaknesses?
5. Analyze a system where deadlock has occurred. Based on the resource allocation graph, identify the processes involved in the deadlock.
6. Compare and contrast the effectiveness of Peterson's solution and hardware-based synchronization mechanisms in solving the critical-section problem.
7. Analyze how segmentation can affect the allocation of memory for processes with different

memory requirements.

8. Compare the different file access methods (sequential, direct, and indexed) and their suitability for various types of applications.
9. Compare the efficiency of different disk scheduling algorithms (e.g., FCFS, SSTF, SCAN, C-SCAN) in terms of average seek time and throughput.
10. Analyze the advantages and disadvantages of using different RAID levels (e.g., RAID 0, RAID 1, RAID 5) in terms of performance, redundancy, and cost.
11. Break down how the principles of protection are applied in a multi-user system to ensure fairness and security.
12. Analyze the security vulnerabilities that can arise from program threats versus network threats and recommend strategies to mitigate them.

L5: Evaluating

1. Evaluate the impact of free and open-source software on the development of modern operating systems.
2. Based on a particular computing environment (e.g., mobile vs. desktop OS), assess the suitability of specific operating systems for that environment.
3. Evaluate the efficiency of CPU scheduling algorithms in a real-time operating system.
4. Critically evaluate the fairness of the Round Robin scheduling algorithm in a system with diverse process priorities & evaluate the efficiency of the Shortest Job First (SJF) scheduling algorithm in a system with mixed process types (short and long processes).
5. Evaluate the advantages and disadvantages of using semaphores over mutex locks in process synchronization.
6. Assess the effectiveness of deadlock avoidance techniques in real-time systems. Which strategy would you recommend for a highly dynamic environment?
7. Evaluate the use of different page-replacement algorithms (e.g., FIFO, LRU, Optimal) in terms of efficiency and system performance.
8. Evaluate the effectiveness of demand paging versus preloading all necessary pages into memory at the beginning of a program's execution.
9. Evaluate how file protection mechanisms (such as access control lists) can be integrated into a file system to prevent unauthorized access.
10. Evaluate a system's vulnerability to program, system, and network threats and propose a security enhancement plan.
11. Evaluate the security strengths and weaknesses of the access matrix model in terms of scalability and flexibility.

L6: Create

1. Design a new microkernel-based operating system for a smart IoT device. Describe the architecture, essential system services, and justify your design choices with respect to security and performance.
2. Create a CPU scheduling simulator that allows the user to switch between different algorithms (FCFS, SJF, RR, Priority). Include multithreading support and compare the performance for different workloads.
3. Develop a solution to a new synchronization problem in a multi-robot warehouse environment using semaphores or monitors. Justify how your solution avoids deadlocks and ensures resource fairness.
4. Design a hybrid memory management scheme for a custom OS that combines paging and segmentation to optimize memory usage in embedded systems. Provide diagrams and justify your approach.
5. Develop a security framework for a cloud-based file storage system that integrates protection principles, access control, and encryption. Illustrate how your framework defends against both system and network threats.

**Chairperson
Board of Studies (CSE)**

R24CSPC12 DATABASE MANAGEMENT SYSTEMS LAB 0 0 3 1.5

(Common to CSE, CSE (DS), CSE (AI &ML))

Course Objectives:

1. Populate and query a database using SQL DDL/DML Commands
2. Declare and enforce integrity constraints on a database
3. Writing Queries using advanced concepts of SQL
4. Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs													
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	BT
R24CSPC 12.1	Utilizing Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL) commands effectively within a database environment	3	2	2	1	3	1	1	1	2	1	2	3	1	L1, L2
R24CSPC 12.2	Constructing and execute queries to manipulate and retrieve data from databases	3	3	2	2	3	1	1	1	2	1	2	3	1	L2, L3
R24CSPC 12.3	Develop application programs using PL/SQL. Establish database connectivity through JDBC (Java Database Connectivity)	2	2	3	2	3	1	1	2	2	2	2	2	3	L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:**Week 1:****CO's:CO1**

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

Week 2:**CO's:CO1**

2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION,

INTERSECT, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.

Week 3: **CO's:CO1**

3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUPBY, HAVING and Creation and dropping of Views.

Week 4: **CO's:CO1**

4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

Week 5: **CO's:CO1**

5.
 - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
 - ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.

Week 6: **CO's:CO2**

6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.

Week 7: **CO's:CO2**

7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exception RAISEAPPLICATION ERROR.

Week 8: **CO's:CO2**

8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

Week 9: **CO's:CO2**

9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.

Week 10: **CO's:CO2**

10. Develop programs using features parameters in a CURSOR, FOR UPDATECURSOR, WHERE CURRENT of clause and CURSOR variables.

Week 11: **CO's:CO3**

11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and

INSTEAD OF Triggers

Week 12:

CO's:CO3

12. Create a table and perform the search operation on table using indexing and no indexing techniques.

Week 13:

CO's:CO3

13. Write a Java program that connects to a database using JDBC

Week 14:

CO's:CO3

14. Write a Java program to connect to a database using JDBC and insert values into it

Week 15:

CO's:CO3

15. Write a Java program to connect to a database using JDBC and delete values from it

Reference Books:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007
4. RamezElmasri, Shamkant, B. Navathe, "Database Systems", Pearson Education, 6th Edition, 2013.
6. Database Principles Fundamentals of Design Implementation and Management, 10th edition, Carlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022
7. edition, Carlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Online Learning Resources:

1. <http://www.scoopworld.in>
2. <http://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/index.php>

**Chairperson
Board of Studies(CSE)**

R24CSPC12**Operating Systems Lab**
(Computer Science and Engineering)**0 0 3 1.5****Course Objectives:**

1. To understand the basic system calls related to process
2. To implement various CPU scheduling algorithms
3. To implement memory management and virtual memory techniques
4. To implement file management and disk scheduling algorithms

Course Outcomes

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

Mapping with POs and PSOs					DoK
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	
R24CSPC12.1	Know the basic system calls related to process	3	2	1	L2
R24CSPC12.2	Simulate CPU scheduling algorithms, disk scheduling algorithms	3	3	2	L3
R24CSPC12.3	Understand memory management and virtual memory techniques, file management techniques	3	3	3	L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:**Week 1:**

1. Practicing of Basic UNIX Commands. CO's:CO1

Week 2:

2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir CO's:CO1

Week 3:

3. Simulate the following CPU scheduling algorithms a) FCFS b) SJF c) Priority d) Round Robin CO's: CO2

Week 4:

4. Control the number of ports opened by the operating system with a) Semaphore b) Monitors. CO's: CO2

Week 5:

5. Write a program to solve producer-consumer problem using Semaphores. CO's: CO2

Week 6:

6. Implement the following memory allocation methods for fixed partition a) First fit b) Worst fit c) Best fit **CO's:CO3**

Week 7:

7. Simulate the following page replacement algorithms a) FIFO b) LRU c) LFU **CO's:CO3**

Week 8:

8. Simulate Paging Technique of memory management. **CO's:CO3**

Week 9:

9. Implement Bankers Algorithm for Dead Lock avoidance and prevention **CO's: CO2**

Week 10:

10. Simulate the following file allocation strategies a) Sequential b) Indexed c) Linked **CO's:CO5**

Week 11:

11. Simulate the following Disk scheduling algorithms. a) FCFS b) SSTF c) SCAN **CO's:CO4**

Week 12:

12. Simulate the following Disk scheduling algorithms. a) C-SCAN b) LOOK c) C-LOOK **CO's:CO4**

Textbooks:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley
2. Operating Systems- A Concept based Approach, D. M. Dhamdhare, 2nd Edition, Tata McGraw-Hill

Reference Books:

1. Operating Systems-Internal and Design Principles, Stallings, 5th Edition, Pearson Education/PHI, 2005
2. UNIX: Concepts and Applications, Sumitabha Das, 4th Edition, Tata McGraw-Hill
3. Operating System: A Design-Oriented Approach, Charles Patrick Crowley
4. Operating Systems: A Modern Perspective, Gary J. Nutt
5. Design of the Unix Operating Systems, Maurice J. Bach

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. The objective of this course is to provide students with hands-on experience and advanced knowledge in programming through the use of multiple languages, including C (Data Structures (DS)), and JAVA.
2. The course aims to develop problem-solving skills, enhance algorithmic thinking, and train students in the practical implementation of core programming concepts across different paradigms and environments.

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

Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	Dok
R24CSPC 13.1	Apply logical techniques such as pattern recognition, sequencing, and classification to solve problems.	3	3	2	2	2	-	-	-	1	1	-	3	2	L3,L4
R24CSPC 13.2	Implement and manage linear and nonlinear data structures (like stacks, queues, linked lists, trees, and graphs).	3	3	3	2	3	-	-	-	2	1	1	3	2	L4,L6
R24CSPC 13.3	Solve complex real-world problems using object-oriented programming concepts in Java.	3	3	3	3	3	2	1	1	3	3	3	2	3	L5,L6

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:**C AND DATA STRUCTURES****WEEK-1****CO's: CO1**

1. The given function has a string (str) and two characters, ch1 and ch2. Execute the function in such a way that str returns to its original string, and all the events in ch1 are replaced by ch2, and vice versa.

Replace character (Char str1, Char ch1, Int 1, Char ch2)

Assumption

This string has only alphabets that are in lower case.

Example

Input:

str: apples

ch1: a

ch2: p

Output:

paales

Explanation: All the 'a' in the string is replaced with 'p'. And all the 'p's are replaced with 'a'.

WEEK-2

COs: CO1

2. In an enchanted garden, flowers bloom in various colors, and each color has a specific point value. Your task is to calculate the total points based on the flowers picked by a gardener.

Input Format

A list of integers representing the points for each flower.

Output Format

An integer representing the total points earned from the flowers.

Example

Input: [5, 3, 8, 2]

Output: 18

WEEK-3

CO's: CO2

3. Count nodes in circular linked list

Algorithm for count nodes in circular linked list

A-Accept the head pointer in the function

B-Declare the count variable which maintains the count of nodes and initialize it to zero.

C-Traverse the link till next node is the head node

D- Define a Node class which represents a node in the list. It has two properties data and next which will point to the next node.

E-Define another class for creating the circular linked list and it has two nodes: head and tail. It has two methods: add () and display ().

WEEK-4

CO's: CO2

4. Implement a binary search tree (BST) and perform operations such as insertion, deletion, and searching.

WEEK-5

CO's: CO2

5. Implement a scenario-based program for a social network using graph data structures.

WEEK-6

CO's: CO1

6. Develop a scenario-based program to simulate a ticket booking system using queues.

WEEK-7**CO's: CO2**

7. Implement graph data structure and perform depth-first search (DFS) and breadth-first search (BFS).

WEEK-8**CO's : CO1**

8. The function accepts string str of size n as an argument. Implement the function which returns 1 if given string str is valid password else 0. str is a valid password if it satisfies the below conditions.

- – At least 4 characters
- – At least one numeric digit
- – At Least one Capital Letter
- – Must not have space or slash (/)
- – Starting character must not be a number

Assumption:

Input string will not be empty.

Example:**Input 1:**

aA1_67

Input 2:

a987 abC012

Output 1:

1

Output 2:

0

WEEK-9**CO's: CO1**

9. The function accepts an integer array 'arr', its length and two integer variables 'num' and 'diff'. Implement this function to find and return the number of elements of 'arr' having an absolute difference of less than or equal to 'diff' with 'num'. Note: In case there is no element in 'arr' whose absolute difference with 'num' is less than or equal to 'diff', return -1.

Example:**Input:**

- arr: 12 3 14 56 77 13
- num: 13
- diff: 2

Output: 3

WEEK-10**CO's: CO1**

10. Write a function to validate if the provided two strings are anagrams or not. If the two strings are anagrams, then return 'yes'. Otherwise, return 'no'.

Input:

Input 1: 1st string

Input 2: 2nd string

Output:

(If they are anagrams, the function will return 'yes'. Otherwise, it will return 'no'.)

Example

Input 1: Listen

Input 2: Silent

Output:

Yes

Programming through JAVA

WEEK-11**CO's: CO3****11. Problem Statement:**

In a classroom some of the seats are already occupied by students and only a few seats are available in the classroom. The available seats are assumed as **r** and **n** number of students is looking for the seat. We need to find in how many different permutations n number of students can sit on r number of chairs.

Algorithm

- Input number of students in n
- Input number of seats in r
- Use permutation formula { $\text{factorial}(n) / \text{factorial}(n-r)$ }
- Print Output

WEEK-12**CO's: CO3**

12. Find all possible Palindromic Partitions of the given String in Java.

WEEK-13**CO's: CO3**

13. Write a Java program to remove a specific element from an array.

WEEK-14**CO's: CO3**

14. Develop a Java application that demonstrates comprehensive file handling operations. The application should perform the following tasks:

- Create a new file
- Read the contents of the file and display it on the console.

- Append additional text to the existing content of the file.

Delete the original file

WEEK-15

CO's: CO3

15. Can a number be expressed as a sum of two prime numbers using java

Here, we will discuss program to check whether a number be expressed as a sum of two prime numbers using java. we will ask the user to enter a positive integer and check whether that number can be expressed as the sum of two prime numbers. If that number can be expressed as sum of two prime numbers then print the numbers in the output and else print the statement that the number cannot be expressed as a sum of two prime numbers.

WEEK-16

CO's: CO3

16. Write a Java program to compare two files lexicographically.

According to Wikipedia:

In mathematics, the lexicographic or lexicographical order (also known as lexical order, dictionary order, alphabetical order or lexicographic (al) product) is a generalization of the way the alphabetical order of words is based on the alphabetical order of their component letters. This generalization consists primarily in defining a total order over the sequences (often called words in computer science) of elements of a finite totally ordered set, often called alphabet.

Textbooks:

1. Data structures Using C Second Edition, Reema Thareja.
2. Programming Problem Solving, Reema Thareja.
3. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019.
4. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson, 2009.

Reference Books:

1. Lets Us C, Yeswanth Kanethkar.
2. Data Structures, S.Chand And Company Limited, Dr.KV Sambasivarao.
3. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition, Pearson, 2015.
4. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006.

**Chairperson
Board of Studies (CSE)**

R24CSSC01**Python Programming**
(Computer Science and Engineering)**0 1 2 2****Course Objectives:**

- Introduce core-programming concepts of Python programming language.
- Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
- Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs			
		PO1	PO2	PO3	Dok
R24ES03.1	Introduce core-programming concepts of Python programming language.	3	3	3	L1
R24ES03.2	Develop, run and manipulate Python programs using Functions, Core data structures like Lists, Dictionaries, and use of Strings Handling methods	3	3	3	L2
R24ES03.3	Demonstrate about Python data structures like Tuples, Sets and dictionaries	3	3	3	L2
R24ES03.4	Develop, run and manipulate Python programs using File Operations and concepts of object-oriented programming	3	3	3	L3
R24ES03.5	Understand Data Science, Numpy, Pandas and working with XML, JSON and other file formats.	3	3	3	L3

Board of Studies : Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:**UNIT-I:****CO's:CO1**

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook, Features, Limitations, advantages, and applications of python.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly

Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif...else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

Sample Experiments:

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
 - i. Arithmetic Operators
 - ii. Relational Operators
 - iii. Assignment Operators
 - iv. Logical Operators
 - v. Bit wise Operators
 - vi. Ternary Operator
 - vii. Membership Operators
 - viii. Identity Operators
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.
7. Write how to Handle specific exceptions – like division of a number by zero

UNIT-II:

CO's:CO2

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments, Lambda Functions: Syntax and usage of lambda functions..

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings, Regular Expressions (Regex)

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

Sample Experiments:

1. Write a Python program to square each element in a list using a lambda function.
2. Write a program to define a function with multiple return values.
3. Write a program to define a function using default arguments.
4. Write a program to find the length of the string without using any library functions.

5. Write a program to check if the substring is present in a given string or not.
6. Write a program to perform the given operations on a list:
 - i. Addition
 - ii. Insertion
 - iii. Slicing
7. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III:**CO's:CO3**

Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement. Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset, Defaultdict and OrderedDict (from collections).

Sample Experiments:

1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed).
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.
6. Write a program using defaultdict to count the frequency of each fruit in the given list: ['apple', 'banana', 'apple', 'orange', 'banana', 'apple']

UNIT-IV:**CO's:CO4**

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

Sample Experiments:

1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.
4. Write a program to create, display, append, insert and reverse the order of the items in the

array.

5. Write a program to add, transpose and multiply two matrices.
6. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V:

CO's:CO5

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

Sample Experiments:

1. Python program to check whether a JSON string contains complex object or not.
2. Python Program to demonstrate NumPy arrays creation using array () function.
3. Python program to demonstrate use of ndim, shape, size, dtype.
4. Python program to demonstrate basic slicing, integer and Boolean indexing.
5. Python program to find min, max, sum, cumulative sum of array
6. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
 - a) Apply head () function to the pandas data frame
 - b) Perform various data selection operations on Data Frame
7. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

Reference Books:

1. Gowri shankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson

Online Learning Resources Links:

1. <https://www.coursera.org/learn/python-for-applied-data-science-ai>
2. <https://www.coursera.org/learn/python?specialization=python#syllabus>

Web References:

1. <https://www.geeksforgeeks.org/python-programming-language-tutorial/>
2. <https://www.python.org/about/gettingstarted/>
3. <https://www.w3schools.com/python/>

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. Define identifiers and give two examples.
2. List any three features of Python.
3. What is the use of the type() function.
4. Define a function. What is a lambda function?
5. What are default and keyword arguments?
6. List any four string methods in Python.
7. Define a dictionary. How is it different from a list?
8. What is a frozenset?
9. What is the syntax of zip()?
10. What is a constructor in Python?
11. List any three file handling functions.
12. Define encapsulation.
13. What is a NumPy array?
14. Define a DataFrame in pandas.
15. Mention any two data visualization functions from matplotlib.

L2: Understand

1. Explain the significance of indentation in Python.
2. Differentiate between is and == with examples.
3. Explain the flow of execution for the if...elif...else structure.
4. Differentiate between *args and **kwargs.
5. Explain string slicing and joining with examples.
6. Explain the scope and lifetime of a variable in Python
7. Explain the relation between tuples and dictionaries.
8. Describe the purpose of defaultdict and OrderedDict.
9. Differentiate between class and object.
10. Explain the difference between class attributes and instance attributes.
11. Explain the difference between .ndim, .shape, and .size in NumPy.
12. Describe the process of reading JSON data in Python.

L3: Apply

1. Write a Python program to swap two variables without a temporary variable.
2. Write a program to print the largest of three numbers using conditional statements.
3. Write a Python program using a lambda function to square each element of a list.
4. Write a function that returns both sum and product of two numbers.
5. Write a program to check if a substring exists in a string.
6. Write a program to count vowels in a string using only string and set operations.

7. Write a program to check if a given key exists in a dictionary.
8. Write a program to concatenate two tuples.
9. Write a Python program to read from a file and print its contents in reverse line order.
10. Write a class representing a rectangle with methods to calculate area and perimeter.
11. Write a Python program to create a NumPy array and perform slicing.
12. Create a dictionary with lists and convert it into a pandas DataFrame.

L4: Analysing

1. Compare the use of for and while loops with scenarios.
2. Analyze how operator precedence affects the outcome of expressions in Python.
3. Analyze how positional arguments differ from keyword arguments in function calls.
4. Compare string formatting methods: %, .format(), and f-strings.
5. Compare the use of sets and dictionaries in Python.
6. Analyze the output when applying slicing on a tuple with an example.
7. Compare inheritance and polymorphism with suitable class examples.
8. Analyze the structure and behavior of a file object during read and write operations.
9. Analyze how data is structured differently in NumPy and pandas.
10. Compare JSON and XML data formats with Python parsing tools.

L5: Evaluating

1. Evaluate the use of dynamic typing in Python. What are its pros and cons?
2. Assess the effectiveness of Anaconda and Jupyter Notebook for beginner programmers
3. Evaluate the usefulness of using regular expressions for string validation.
4. Justify the use of built-in functions over writing manual logic
5. Evaluate the use of immutable data types (tuples, frozensets) in large applications.
6. Assess the performance difference between regular dict and defaultdict when counting frequencies.
7. Evaluate the usefulness of OOP in Python compared to procedural programming.
8. Judge whether using binary files or text files is better for storing structured data.
9. Evaluate the benefits of using pandas over lists/dictionaries for data manipulation.
10. Assess the effectiveness of functional programming in handling large-scale data pipelines.

**Chairperson
Board of Studies (CSE)**

Course Title: Quantitative Aptitude & Logical reasoning	Course Code: R24HS04
Teaching Scheme(L:T:P): 0:1:2	Credits:2
Type of Course: Tutorial+ Practical	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Prerequisites: To succeed in the Quantitative Aptitude & Logical reasoning course, students should have a basic understanding arithmetic, algebra, and geometry from school-level mathematics. Analytical thinking and English comprehension skills for interpreting logical patterns and problems.	

Course objective:

- Build a strong foundation in quantitative aptitude and logical reasoning.
- Enhance problem-solving skills for topics like percentages, profit and loss, time and work, and logical reasoning puzzles.
- Improve speed, accuracy, and critical thinking for efficient problem-solving.
- Prepare students for competitive exams and real-world applications of math and logic.

Course outcome:

COs	PO1	PO2	PO3	PO4	PO5	PO11	PSO1	BT LEVEL
CO1: Mastery of Key Concepts – Understand number systems, percentages, time/work, profit/loss, and series completion.	3	2	1	2	1	-	1	L2
CO2: Improved Problem-Solving Skills – Solve mathematical and logical problems with speed and accuracy.	3	3	2	3	1	-	1	L3
CO3: Enhanced Analytical Thinking – Develop critical thinking for reasoning puzzles and real-life challenges.	2	3	2	3	2	-	1	L4
CO4: Competitive Exam Preparedness – Be well-prepared for exams requiring aptitude and reasoning.	3	2	1	2	2	1	2	L3

Aptitude:

Unit 1: Number System: Speed Maths, Numbers, Factors, Prime & Co-Primes, LCM, HCF, Divisibility rules, finding unit place digit and last two digits of an expression.

Averages and Ages: Average of different groups, change in averages by adding, deleting and replacement of objects, problems on ages.

Ratio, Proportion and Variations: Definition of Ratio, Definition of Proportion, Types of ratios, Types of proportions, mixture model, age model, salary model questions, Direct and indirect

proportion. Allegation and mixtures: Allegation rule.

COs- CO1

Unit 2: Percentages: Converting fractions and decimal into percentages, successive percentage, populations, expenditure and savings.

Profit and loss: Relation between Cost price and Selling price, Discount and Marked price, Gain or Loss percentages on selling price

Simple and Compound Interest: Problems on Interest (I), Amount (A), Principal (P)and Rate of Interest(R), Difference between the simple interest and compound interest for 2 and 3years.

COs- CO1, CO2

Unit 3:Time and Work: Men and Days, Work and Wages, Hours and Work, Alternate days concept, Chain rule.

Time and Distance: Difference between the average and relative speeds, reaching the destination late and early, Stoppage time per hour, time and distance between two moving bodies.

Trains, Boats and Streams: Train crossing man, same and opposite directions, Speed ofboat and stream.

COs- CO2, CO3

Logical Reasoning

Unit 4: Series completion: Number series, Alphabet series and letter series.

Blood Relations: Defining the various relations among the members of a family, Solving Blood Relation Puzzles by using symbols and notations. Problems on Coded relations.

Coding and Decoding: Letter coding, Number coding, Number to letter coding, Matrix coding, Substitution, Mixed letter coding, Mixed number coding, deciphering individual letter codes by analysis.

Direction sense test: Sort of directions in puzzles distance between two points, problems on shadows, Application of triangular triplets.

COs- CO3, CO4

Unit 5: Clocks: Relation between minute-hour hands, anglevs. time, exceptional cases in clocks

Calendars: Definition of a Leap Year, Finding the odd days, finding the day of any random calendar date, repetition of calendar years.

COs- CO3, CO4

Text Books:

1. R.S.Aggarwal“QuantitativeAptitude”,Reviseded.,SChandpublication,2017
ISBN:8121924987
2. R.S.Aggarwal “Verbal- Non verbal Reasoning”,Reviseded.,SChandpublication,2017

E- resources:

1. <https://www.indiabix.com/aptitude/questions-and-answers/>
2. https://www.tutorialspoint.com/quantitative_apititude/
3. <https://www.careerbless.com/aptitude/qa/home.php>

**Chairperson
Board of Studies (CSE)**

Course Title: ENVIRONMENTAL SCIENCE	Course Code: R24MC03
Teaching Scheme(L:T:P):2:0:0	Credits:-
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: To succeed in the Environmental Science course, students should have a basic knowledge of high school-level biology, physics, and chemistry. Awareness of environmental issues like pollution, climate change, and conservation is helpful. Interest in sustainability and nature will enhance understanding and engagement.	

Course Objectives:

- To make the students to get awareness on environment.
- To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day to day activities of human life
- To save earth from the inventions by the engineers.

Course Outcomes:

After completion of the course, students will be able to

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO11	PSO1	BT LEVEL
CO1: Understand the scope, importance, and multidisciplinary nature of environmental studies, and analyze the exploitation of natural resources.	2	3	1	2	2	1	2	2	L2,L4
CO2: Describe the structure, function, and energy flow in ecosystems, and understand the importance of biodiversity and its conservation.	3	2	2	3	2	1	3	1	L2,L3
CO3: Evaluate the causes, effects, and control measures of different types of environmental pollution, and understand the strategies for solid waste management.	3	3	2	3	3	2	2	1	L4,L5
CO4: Examine the concept of sustainable development, urban environmental issues, and the role of	3	2	3	3	3	2	3	2	L4,L5

environmental ethics and legislation in protecting the environment.									
CO5: Analyze the relationship between human population growth and environmental degradation, and evaluate the role of population management and health programs in sustainable development.	3	2	2	2	3	2	3	1	L4,L5

SYLLABUS

UNIT I

Multidisciplinary Nature of Environmental Studies: –Definition, Scope and Importance
 – Need for Public Awareness.

Natural Resources : Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies–Timberextraction–Mining,damsandothereffectsonforestandtribalpeople–Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems – Mineral resources: Use and exploitation, environmentaleffectsofextractingandusingmineralresources,casestudies–Foodresources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. – Energy resources.

COs- CO1

UNIT II

Ecosystems: Concept of an ecosystem.–Structure and function of an ecosystem–Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids – Introduction, types, characteristic features, structure and function of the following ecosystem:

- a. Forest ecosystem.
- b. Grass land ecosystem
- c. Desert ecosystem.

d. Aquatic ecosystems(ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity and its Conservation: Introduction 0 Definition: genetic, species and ecosystem

diversity – Bio-geographical classification of India – Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

COs- CO2

UNIT III

Environmental Pollution: Definition, Cause, effects and control measures of:

- a. Air Pollution.
- b. Water pollution
- c. Soil pollution
- d. Marine pollution
- e. Noise pollution
- f. Thermal pollution
- g. Nuclear hazards

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and landslides.

COs- CO3

UNIT IV

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to energy – Water conservation, rain water harvesting, water shed 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 – Public awareness.

COs- CO4

UNIT V

Human Population and the Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – 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.

Field Work: 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, and birds – river, hill slopes, etc. **COs- CO5**

Textbooks:

1. Textbook of Environmental Studies for Undergraduate Courses ErachBharucha for University Grants Commission, Universities Press.
2. Palaniswamy, “Environmental Studies”, Pearson education
3. S.AzeemUnnisa, “Environmental Studies”Academic Publishing Company
4. K.RaghavanNambiar, “Textbook of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, Scitech Publications (India), Pvt. Ltd.

References:

1. DeekshaDave and E.SaiBabaReddy, “Textbook of Environmental Science”, Cengage Publications.
2. M.AnjiReddy, “Textbook of Environmental Sciences and Technology”, BS Publication.
3. J.P.Sharma, Comprehensive Environmental studies, Laxmi publications.
4. J.Glynn Henry and Gary W.Heinke, “Environmental Sciences and Engineering”, Prentice Hall of India Private limited
5. G.R.Chatwal, “A Textbook of Environmental Studies ”Himalaya Publishing House
6. Gilbert M. Masters and Wendell P. Ela, “Introduction to Environmental Engineering and Science, Prentice Hall of India Private limited.

**Chairperson
Board of Studies (CSE)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B.Tech Computer Science and Engineering

Regulation: R24

III Year I Semester -Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC15	Compiler Design	3	0	0	3
2	PC	R24CSPC16	Computer Networks	3	0	0	3
3	HS	R24HS05	Managerial Economics and Financial Analysis	2	0	0	2
4	PE-1	R24CSPE1.1 R24CSPE1.2 R24CSPE1.3 R24CSPE1.4	1. Data Warehousing and Data Mining 2. Software Testing Methodologies 3. Human Computer Interaction 4. Information Retrieval and Search Engines 5. 12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
5	OE-1		OPEN ELECTIVE-1	3	0	0	3
6	PC	R24CSPC17	Compiler Design Lab	0	0	3	1.5
7	PC	R24CSPC18	Computer Networks Lab	0	0	3	1.5
8	SC	R24CSSC03	Full Stack Development-2	0	0	2	1
9	IN	R24IN01	Evaluation of Community Service Project	2	0	0	2
10	HS	R24HS06	Design Thinking & Innovation	1	0	2	2
11	HS	R24HS09	Advanced Aptitude and Communication Skills	2	0	0	1
12	MC	R24MC05	Constitution of India	2	0	0	0
Total				21	00	10	23

Category	Courses	Credits
PC-Professional Core Courses	4	9
PE- Professional Elective Courses	1	3
OE-Open Elective Courses	1	3
Community Service Project	1	2
SEC-Skill Enhancement course	1	1
HS-Humanities and Social Science including Management Course	3	5
MC-Mandatory Course	1	0
Total	12	23

R24CSPC15

COMPILER DESIGN
(Common to CSE)

3 0 0 3

Course Objectives:

1. Understand the phases of a compiler and language processing systems.
2. Apply lexical and syntax analysis techniques.
3. Analyze syntax – directed translation and semantic processing.
4. Design intermediate representations and run time environments.
5. Implement code optimization and code generation techniques.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPC15.1	Understand the basics of language processing and implement lexical analyzer for any language	3	-	2	-	2	-	-	-	-	1	-	3	-	L1, L2
R24CSPC15.2	Describe the different types of parsing and implement parser for any language	3	3	2	1	2	-	-	-	1	1	-	3	2	L2, L3
R24CSPC15.3	Explain the different intermediate code representations and use Syntax directed	3	3	2	2	2	-	-	-	-	1	-	3	2	L2, L4
R24CSPC15.4	Apply the basics of data flow analysis, optimizations, and runtime environment required for handling recursive procedures	3	2	3	2	3	-	-	-	-	1	1	3	2	L3, L4
R24CSPC15.5	Apply machine-independent and machine-dependent code optimization techniques, data-flow analysis, and target code generation methods to improve the efficiency of compiled programs.	3	3	3	1	2	-	-	-	-	-	-	3	2	L5 L6

SYLLABUS**UNIT-I:****15Hours**

Introduction To Compiling: The evolution of programming languages and basic language processing system; Phases of a Compiler, Bootstrapping and Simple one pass compiler, Cross Compilers.

Lexical Analysis: The Role of Lexical Analyzer, Input Buffering, and Specification of Tokens: Regular Expressions, Regular Definitions- Recognition of Tokens, A language for specifying Lexical Analyzers (LEX).

COs-C01

Self-LearningTopics: LEX/FLEX tool basics, Regular expressions practice, Token design for

Simple languages

UNIT-II: 12Hours

Syntax Analysis: The role of the Parser, Context free Grammars, Elimination of Left Recursion, Left factoring a grammar

Top-Down Parsing: Recursive descent Parsing, First and Follow, Predictive Parsing, LL (1) Grammars.

Bottom-Up Parsing: Shift-Reduce Parser, LR Parsers SLR, Canonical LR, LALR, Operator Precedence Parser

COs-C02

Self-Learning Topics: YACC/Bison basics, Ambiguous grammars, Parser error handling

UNIT-III: 14Hours

Syntax-Directed Translation and Semantic Analysis: Syntax directed definition (SDD) Syntax directed translation (SDT) and translation schemes (TS), Synthesized and Inherited Attributes, S-Attributed SDD, L-Attributed SDD

Type Checking: Type Systems, Specification of a Simple type checker, Equivalence of Type Expressions, Type Conversions

COs-C03

Self-Learning Topics: Attribute grammars, Type checking concepts, Syntax tree applications

UNIT-IV: Run-Time Environments and Intermediate Code Generation: 15Hours

Run-Time Environments: Storage Allocation Strategies: Static Allocation, Stack Allocation, Heap Allocation, Activation records, Access Links, Symbol Tables. Heap management, Introduction to garbage collection

Intermediate Code Generation: Interdiction to intermediate address code: forms of intermediate code, Three Address Code, Implementations, basic blocks, DAG, DAG representation of basic blocks, Applications of DAG, DAG based local optimization.

COs-C04

Self-Learning Topics: Garbage collection, Symbol table implementation

UNIT-V: Code optimization and Target Code Generation Code Generation: 14Hours

Code optimization: Introduction, Principal sources of optimization; Machine-Independent optimizations: The principal sources of optimization, Basic blocks and flow graphs, Introduction to data-flow analysis, Foundations of data-flow analysis,

Target Code Generation Code Generation: Issues in the design of a Code Generator, the target language, addresses in the target code, Machine-dependent Optimizations: Peephole optimization, Register allocation and assignment, Dynamic Programming code generation.

COs-C05

Self-Learning Topics: Code optimization techniques, Register allocation

Board of Studies :Computer Science and Engineering

Approved in BoS No : 03

Approved in ACM No:03

Text Books:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.
2. Compiler Construction-Principles and Practice, Kenneth C. Louden, Cengage Learning

Reference Books:

1. Modern Compiler Implementation in C, Andrew W. Appel, Revised edition, Cambridge University Press.
2. The Theory and Practice of Compiler Writing, J. P. Tremblay and P. G. Sorenson, TMH
3. lex & yacc, 2nd Edition by John Levine, Doug Brown, Tony Mason

Web Resources:

1. <https://www.edx.org/course/compiler>
2. <https://nptel.ac.in/courses/106/108/106108113/>

Internal Assessment Pattern

Cognitive Level	InternalAssessment–1(%)	InternalAssessment–2(%)
L1(Remember)	35	–
L2(Understand)	40	–
L3(Apply)	25	25
L4(Analyze)	–	35
L5(Evaluate)	–	25
L6(Create)	–	15
Total(%)	100	100

Chairperson
Board of Studies (CSE)

R24CSPC16

COMPUTER NETWORKS
(Computer Science and Engineering)

3 0 0 3

Course Objectives:

1. To understand the various functionalities of computer networks topologies, architectures and transmission mediums
2. To understand state-of-the-art in network protocols, architectures, and applications.
3. To explore the various layers of OSI Model and TCP / IP Model

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs														Dok
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
R24CSPC16.1	Demonstrate knowledge of network models such as OSI, TCP/IP, along with communication techniques, transmission media	3	1	-	-	1	-	-	-	-	-	2	3	1	L1, L2	
R24CSPC16.2	Analyze data link layer services, functions and evaluate medium access control protocols including ALOHA, CSMA/CD, CSMA/CA, Token Passing, FDMA, TDMA, and CDMA.	3	3	1	2	2	-	-	-	-	-	2	3	2	L2, L3	
R24CSPC16.3	Analyze network layer services, routing algorithms (shortest path, distance vector, link state) and congestion control mechanisms, and apply them to improve network efficiency. (<i>Analyze</i>)	3	2	2	2	2	-	-	-	-	-	2	3	2	L3, L4	
R24CSPC16.4	Analyze transport layer protocols, including TCP and UDP, their connection management, flow control, error control, and congestion handling mechanisms, and understand end-to-end communication concepts. (<i>Analyze</i>)	3	2	1	2	2	-	-	-	-	-	2	3	2	L4, L5	
R24CSPC16.5	Analyze the working of application layer protocols such as HTTP and DNS, including architecture, message formats, and client-server interactions. (<i>Analyze</i>)	3	2	3	1	1	2	-	-	-	-	1	3	3	L5, L6	

SYLLABUS**UNITI: Introduction & Physical Layer****15Hours**

Introduction: Network Topologies and Types: LAN, MAN, WAN. Network Reference models: The OSI Reference Model, The TCP/IP Reference Model. OSI Vs TCP/IP models comparison, Internet History – Arpanet.

Physical Layer: Introduction to Guided Transmission Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and unguided media: Wireless: Radio waves, microwaves, infrared.

Self-Learning Topics: Digital Modulation and Multiplexing. **CO's–CO1**

UNIT II: Data Link Layer and Media Access Control **15Hours**

Design Issues: Services Provided to the Network Layer

Error Detection and Correction: Error Detection: CRC, Checksum. Error Correction:

Framing: Bit and Byte stuffing; fixed size framing, variable size framing.

Flow Control: Sliding Window Protocols. Stop and wait ARQ; GoBack N, Selective repeat

Channel allocation methods: Channelization: TDM, FDM.

Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance – IEEE standard 802 for LANS –Ethernet,

Controlled Access: Token Bus, Token ring. Bridges and virtual LANs. **CO's–CO2**

Self-Learning Topics: IEEE 802.11 and 802.16. Data link layer switching,

UNIT III: Network Layer and Congestion Control **15Hours**

Design Issues: Services Provided to the Transport layer, Store and Forward Packet Switching, Implementation of Connectionless Service-Implementation of Connection Oriented Service-Comparison of Virtual Circuit and Datagram Networks,

Routing Algorithms: The Optimality principle. Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing,

Congestion Control algorithms–General principles of congestion control, Congestion prevention policies, Choke packets, Load shedding, and Jitter Control. **CO's–CO3**

Self-Learning Topics: Internet Working: Tunnelling, internetworking, Fragmentation, Network layer in the internet–IP protocols-IP Version4 protocol-IPV4 Header Format, IP addresses, Class full Addressing, CIDR, NAT-, Subnets,

UNIT IV: Transport Layer **13Hours**

The Transport Layer: Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, End to end protocols: UDP, Real Time Transport Protocol.

The Internet Transport Protocol: TCP- reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call. **COs–CO4**

Self-Learning Topics: Internet control protocols: OSPF, BGP, Internet multicasting, Mobile IP, IPV6.ICMP-ARP- DHCP

UNIT V: Application Layer **12 Hours**

WWW and HTTP: Architecture- Client (Browser), Server, Uniform Resource Locator

HTTP: HTTP Transaction, HTTP Operational Model and Client / Server Communication, HTTP Generic Message Format, HTTP Request Message Format, HTTP Response Message Format.

The Domain Name System: The DNS namespace, Resource Records, Name Servers

CO's-CO5

Self-Learning Topics: Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/ PO's/PSO's

1. **Computer Networks in Cloud Computing** (Cloud networking in platforms like AWS, Azure, GCP; Virtual networks, load balancing, network security, SDN concepts) CO3–CO5, PO1, PO2, PO3, PO5, PO11, PSO1, PSO2
2. **Network Security and Cyber security Practices** (Firewalls, intrusion detection systems, secure protocols, network attacks and prevention techniques) CO2,CO3,CO5,PO1,PO2, PO3, PO5, PO11, PSO1, PSO2
3. **Emerging Networking Technologies** (IoT networking, 5G communication, network virtualization, software-defined networking (SDN)CO1, CO 3, CO5, PO1, PO2, PO3, PO5, PO11, PSO1, PSO2

Text Books:

1. **Dr.Sanjay Sharma**, *ComputerNetworks*,2ndEdition,KatsonBooks,NewDelhi, 2019
2. **B.A.Forouzan**, *Data Communications and Networking*, 5th Edition, McGraw Hill Education (India), 2017.
3. *Computer Networks*–AndrewS Tanenbaum, Fifth Edition. Pearson Education/PHI
4. *Computer Networks*, 5ed, David Patterson,Elsevier

Reference Books:

1. *Data Communications and Networks*–AchutS Godbole,AtulKahate
2. *Computer Networks*, Mayank Dave,CENGAGE
3. **N.N.SakhareandA.V.Dhumane**, *Computer Network*,1st Edition, Technical Publications, Pune, 2018.
4. **D.S.GaikwadandN.D.Lambe**, *Computer Network Technology*,1st Edition,Technical Publications, Pune, 2016.
5. **AchyutS.Godbole**, *Data Communications and Networks*, 2nd Edition, Mc GrawHill Education (India), 2013.
6. *An Engineering Approach to Computer Networks*- S.Keshav,2ndEdition, Pearson Education

Web References:

1. **NPTEL(SWAYAM)**:<https://nptel.ac.in>
2. **GeeksforGeeks**:<https://www.geeksforgeeks.org>
3. **CiscoNetworkingAcademy**:<https://www.netacad.com>
4. **TutorialsPoint**:<https://www.tutorialspoint.com>

Internal Assessment Pattern

CognitiveLevel	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies (CSE)

Course Objectives:

1. To familiarize the concepts and architectural types of data Warehouses.
2. Provides efficient design and management of data storages using data warehousing and OLAP
3. To understand the fundamental processes concepts and techniques of data mining.
4. To consistently apply knowledge concerns current data mining research and how this may contribute to the effective design and implementation of data mining applications.
5. To provide advance research skills through the investigation of data-mining Literature

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPC 24.1	Design a data mart or data warehouse for any organization.	3	3	2	2	2	-	-	-	-	-	-	3	2	L1, L2
R24CSPC 24.2	Apply Association and classification knowledge to different data sets.	3	2	2	2	3	-	-	-	-	1	-	3	3	L2, L3
R24CSPC 24.3	Apply the clustering Techniques for different data sets.	3	3	2	2	3	-	-	-	-	1	-	3	3	L3, L4
R24CSPC 24.4	Identify the similar objects using clustering techniques.	2	2	2	2	3	-	-	-	-	1	2	2	3	L4, L5
R24CSPC 24.5	Explore recent trends in data mining such as web mining, spatial-temporal mining.	2	2	1	2	2	-	-	-	-	1	2	2	2	L5, L6

SYLLABUS

UNIT-I

15 Hours

Data Warehouse and OLAP Technology: what is a Data Warehouse, Multidimensional Data Model, OLAP Operations on Multidimensional Data, Data Warehouse Architecture Cube computation: Multiway Array Aggregation, BUC

CO's-CO1

Self-Learning Topics: Advanced cube computation techniques, OLAP vs. OLTP in practical business scenarios

UNIT-II

13Hours

Introduction to Data Mining: Fundamentals of data mining, KDD, Challenges, Data Mining Functionalities, Data Mining Task Primitives, Major issues in Data Mining. Data Preprocessing: Needs for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction

CO'S - CO2

Self-Learning Topics: Data quality issues in real-world datasets, Ethical issues in data mining

UNIT - III

13Hours

Mining Frequent Pattern: Associations and Correlations: Basic Concepts, Efficient and Scalable Frequent Item Set Generation, The APRIORI Principle, Support and Confidence Measures, Association Rule Generation; APRIORI Algorithm, The Partition Algorithms, FP Growth Algorithms, Compact Representation of Frequent Item Set- Maximal Frequent Item Set, Closed Frequent Item Set. Classification and Prediction: Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification

Self-Learning Topics: Improvements over APRIORI (ECLAT, FP-Growth variations), Over fitting and under fitting problems

CO's-CO3

UNIT-IV

15Hours

Cluster Analysis: Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods- K- means, PAM, Hierarchical Methods- BIRCH, Density- Based Methods- DBSCAN, and Outlier Detection

Self-Learning Topics: Advanced clustering methods (OPTICS, Mean-Shift), Applications of clustering in image processing

CO's-CO4

UNIT-V:

14 Hours

Overview Sequential pattern mining: Mining time series data, mining sequence patterns in transactional databases, graph mining, spatial data mining, text mining, mining the World Wide Web. Data Mining Applications

CO's-CO5

Self-Learning Topics: Data mining in AI and machine learning pipelines, social network analyses

Board of Studies: Computer Science and Engineering
Approved in BoS No:03

Approved in ACM No:03

Text Books:

1. Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques, Second Edition, Elsevier, 2007.
2. Data Mining– Vikram Pudi & P. Radha Krishna-Oxford Publication.

Reference Books:

1. Data Mining Concepts and Techniques, Han, Morgan Kaufmann publishers.
2. Introduction to Data Mining by Pang-Ning Tan, Vipin Kumar, Michael Steinbach

Internal Assessment Pattern

Cognitive Level	Internal Assessment#1(%)	Internal Assessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

(Professional Elective-I)

Course Objectives:

1. To understand the fundamental concepts of software testing, including testing principles, models, and classification of software defects.
2. To apply structural testing techniques such as path testing, flow graphs, data flow testing, and domain testing for effective test case design.
3. To analyze specification-based testing techniques including syntax testing, decision tables, path expressions, and regular expressions.
4. To understand state-based and graph-based testing methods for modeling and validating system behavior.
5. To gain practical knowledge of modern testing tools and frameworks, including automation tools, CI/CD integration, API testing, and emerging testing technologies.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE1.2.1	Understand the fundamental concepts of software testing, including purpose, models, testing principles, and classification of software defects.	3	2	1	-	-	-	-	-	1	-	2	2	1	L1 L2
R24CSPE1.2.2	Apply structural testing techniques such as flow graph testing, path testing, data flow testing, and domain testing to design effective test cases.	3	3	2	2	1	-	-	-	-	1	2	3	1	L3
R24CSPE1.2.3	Analyze specification-based testing techniques including syntax testing, decision tables, path expressions, and regular expressions for systematic test design.	3	3	3	2	1	-	-	-	-	-	2	3	2	L3, L4
R24CSPE1.2.4	Understand and apply state-based and graph-based testing techniques for modeling and validating system behavior using state graphs and matrices.	3	3	2	3	1	-	-	-	-	-	2	3	2	L4

R24CSP E1.2.5	Apply modern software testing tools and frameworks including automation tools, CI/CD integration, API testing, performance testing, and emerging technologies in real-world testing environments.	2	2	3	2	3	-	-	1	1	1	3	2	3	L5, L6
------------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----------

SYLLABUS

UNIT-I: 12Hours

Introduction: Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs, Taxonomy of Bugs

Flow Graphs and Path Testing: Basic Concepts of Path Testing, Predicates, Path Predicates and Achievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing **CO's-CO1**

Self-Learning: Path Instrumentation (advanced techniques), Deep Path Sensitization methods

UNIT-II: 15Hours

Transaction Flows Testing: Transaction Flows, Transaction Flow Testing Techniques,

Dataflow Testing: Basics of Dataflow Testing, Strategies in Dataflow Testing, Application of Dataflow Testing, Domain Testing, Domains and Paths, Nice and Ugly Domains,

Domain Testing: Domains and Interfaces Testing, Domain and Interface Testing, Domains and Testability, **CO's-CO2**

Self-Learning: Advanced Dataflow Analysis techniques, Formal DU-path analysis methods

UNIT-III: 13Hours

Paths, Path Products and Regular Expressions: Path Products and Path Expression, Reduction Procedure, Applications, Regular Expressions and Flow Anomaly Detection

Syntax Testing: Why What and How, Grammar for Formats, Test Case Generation, Implementation and Application, Testability Tips

Based Testing: Overview, Decision Tables, Path Expressions, KV Charts, Specifications

Self-Learning: Graph reduction algorithms, Regular expression-based flow anomaly detection, KV chart advanced simplification techniques **CO's-CO3**

UNIT-IV: 12Hours

State, State Graphs and Transition Testing: State Graphs, Good and Bad State Graphs, State Testing, Testability Tips

Graph Matrices and Application: Motivational Overview, Matrix of Graph, Relations, and Power of a Matrix, Node Reduction Algorithm **CO's-CO4**

Self-Learning: Advanced node reduction algorithms, Matrix power computation in graph theory, Formal state machine minimization techniques

UNIT-V: 14Hours

Software Testing Tools: Introduction to Testing, Automated Testing, Concepts of Test Automation, Test Automation Pyramid, CI/CD Integration, Selenium, JMeter, Postman, Cypress, Playwright, Basics of Test Scripting, Test Execution and Debugging, Test Reporting and Analysis, Cross Browser Testing, API Testing Basics, Performance Testing Basics, DevOps Testing Concepts, Cloud Based Testing, AI Assisted Testing Introduction **CO's-CO5**

Self-Learning: Legacy tools overview (Win Runner, Load Runner basics), advanced scripting techniques, Tool-specific deep features (optional study)

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. Software testing techniques– Boris Beizer, Dream Tech, second edition.
2. Software Testing- Yogesh Singh, Cambridge

Reference Books:

1. Introduction to Software Testing, P.Ammann & J.Offutt,CambridgeUniv.Press.
2. Foundations of Software Testing, D.Graham & Others, CengageLearning.
3. The Craft of software testing-Brian Marick, Pearson Education.

Web References:

1. https://onlinecourses.nptel.ac.in/noc26_cs59/

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Understand the fundamental concepts, principles, and scope of Human–Computer Interaction and its role in designing interactive systems.
2. Learn usability goals, human factors, and user-centered design (UCD) principles for developing effective interfaces.
3. Analyze different interaction styles such as menus, forms, command languages, and natural language interfaces.
4. Design and evaluate user interfaces based on usability principles, visual design guidelines, and user experience metrics.
5. Understand interaction devices and modern input/output technologies used in computing systems.
6. Explore information visualization, search systems, and emerging technologies in HCI such as intelligent and immersive interfaces.

Course Outcomes:

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

Course Code	Course Outcomes (COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSP E1.3.1	Understand HCI fundamentals, usability principles, accessibility, and design processes.	3	2	1	1	1	1	1	-	1	-	2	2	1	L1, L2
R24CSP E1.3.2	Apply interaction design principles to develop effective and user-friendly interfaces.	2	3	3	2	3	1	1	1	2	1	2	3	3	L3
R24CSP E1.3.3	Analyze different interaction modalities, devices, and interface types.	2	3	2	2	2	1	-	-	1	-	2	3	2	L3, L4
R24CSP E1.3.4	Evaluate user experience based on performance, usability, and design principles.	2	3	2	3	2	2	2	-	1	-	2	2	2	L4
R24CSP E1.3.5	Design user-centered systems, documentation, and information visualization solutions.	2	2	3	2	3	2	1	1	2	2	2	3	3	L5, L6

SYLLABUS**UNITI: Foundations & Design Processes****12Hours**

Introduction to Human–Computer Interaction, usability of interactive systems, usability goals and measures, usability motivations, universal usability and accessibility standards, organizational

design to support usability, User-Centered Design (UCD) lifecycle, design thinking, ethnographic observation, participatory design, scenario development, user journeys, ethical, social and legal issues in HCI, basic usability testing methods. **CO's-CO1**

Self-Learning Topics: Agile UX methodologies, remote usability testing, UX analytics tools.

UNIT II: Interaction Design & Modern Interfaces **12Hours**

Menu selection principles, task-related menu organization, content organization, form fill-in, dialog boxes and alternatives, mobile-first design, responsive design, conversational user interfaces, gesture-based interaction, micro interactions and feedback mechanisms, design for small displays such as smart phones and wearable devices. **CO's-CO2**

Self-Learning Topics: Chatbot and voice-based interfaces, wearable device UI case studies.

UNIT III: Interaction Modalities & Devices **10Hours**

Command-based interfaces, graphical user interfaces, natural language interfaces, interaction devices including keyboards, keypads, pointing devices, touch interfaces, stylus, speech and auditory interfaces, displays including small and large screens. **Self-Learning Topics:** multimodal interaction, eye-tracking systems, gesture recognition systems, introduction to Augmented Reality and Virtual Reality interfaces. **CO's-CO3**

UNIT IV: UX Design, Performance & Evaluation **12 Hours**

Quality of service in user experience, models of response-time impact, user expectations and attitudes, user productivity, variability in response time, frustrating experiences, error messages and feedback systems, display design, web page design, window design, color theory, UX evaluation basics. **CO's-CO4**

Self-Learning Topics: UX analytics and heat maps, personalization using Artificial Intelligence, ethical UX and dark pattern case studies.

UNITV: Information Systems & Visualization **12Hours**

User documentation and online help systems, online tutorials, development process, information search in textual documents, database querying, multimedia document search, basic filtering techniques, introduction to information visualization and dashboards.

Self-Learning Topics: recommendation systems, semantic search techniques, big data Visualization tools. **CO's-CO5**

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley Dreama Tech. Operating Systems- A Concept based Approach, D. M. Dhamdhare, 2nd Edition, Tata McGraw-Hill
2. Designing the User Interface, Strategies for Effective Human Computer Interaction, 5ed, Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven M Jacobs, Pearson

Reference Books:

1. Human Computer, Interaction Dan R.Olsan, Cengage, 2010.
2. Designing the user interface.4/e,Ben Shneidermann,PEA.
3. User Interface Design, Soren Lauesen,PEA
4. User Interface Design,Soren Lauesen,PEA

Web References:

1. https://onlinecourses.nptel.ac.in/noc26_cs70/

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board ofStudies(CSE)**

R24CSPE1.4 INFORMATION RETRIEVAL AND SEARCH ENGINES 3 0 0 3
(Professional Elective-I)

Course Objectives:

1. Understand the fundamentals of information retrieval using Boolean models, including inverted index construction, query processing, and text preprocessing techniques such as tokenization, stemming, and phrase indexing.
2. Understand dictionary structures and indexing techniques in information retrieval systems, along with tolerant retrieval methods such as wildcard queries, spelling correction, and efficient index construction methods.
3. Understand statistical properties of terms in documents and apply compression techniques for indexes, along with term weighting methods such as TF-IDF for effective information retrieval.
4. Understand the vector space model for document scoring and ranking, including query representation, similarity computation, and document weighting schemes.
5. Understand evaluation techniques for information retrieval systems and gain knowledge of web search engine concepts, including crawling, indexing, and web structure analysis.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE1.4.1	Describe the pre-processing methods for effective information retrieval	3	2	--	--	2	--	--	--	--	--	--	3	2	L1, L2
R24CSPE1.4.2	Explain about tolerant information retrieval	3	2	1	--	2	--	--	--	--	--	--	3	2	L2, L3
R24CSPE1.4.3	Illustrate the index compression process	3	3	2	1	2	--	--	--	--	--	--	--	--	L3, L4
R24CSPE1.4.4	Build textual information into vectors	3	3	3	2	3	--	--	--	--	--	--	3	3	L4, L5
R24CSPE1.4.5	Make use of a web search engine for crawling the data	3	3	3	2	3	1	--	--	--	2	--	3	3	L5, L6

SYLLABUS

UNIT I:

15 Hours

BOOLEAN RETRIEVAL: An example information retrieval problem, a first take at building an inverted index, processing Boolean queries, the extended Boolean model versus ranked retrieval.

THE TERM VOCABULARY AND POSTINGS LISTS: Document delineation and character sequence decoding, obtaining the character sequence in a document, choosing a document unit, Determining the vocabulary of terms, Tokenization, dropping common terms: stop words, Normalization (equivalence classing of terms) stemming and lemmatization, Faster postings list

intersection via skip pointers, Positional postings and phrase queries, Bi-word indexes, Positional indexes, Combination schemes.

CO's-CO1

Self-Learning Topics: Lemmatization using NLP Tools, Real-Time Inverted Index Construction

UNIT II:

13 Hours

DICTIONARIES AND TO LERAN TRETRIEVAL: Search structures for dictionaries, Wildcard queries, General wild card queries, k-gram indexes for wild card queries, spelling correction, implementing spelling correction, Forms of spelling correction, Edit distance, k-gram indexes for spelling correction, Context sensitive spelling correction, Phonetic correction.

INDEX CONSTRUCTION: Hardware basics, Blocked sort-based indexing, Single-pass in memory indexing, distributed indexing, dynamic indexing, other types of indexes

CO's-CO2

Self-Learning Topics: Trie Data Structure for Dictionary Search, Search Auto-Completion Techniques

UNITIII:

15Hours

INDEX COMPRESSION: Statistical properties of terms in information retrieval, Heaps' law: Estimating the number of terms, Zipf's law: Modeling the distribution of terms, Dictionary compression, Dictionary as a string, Blocked storage, Postings file compression, Variable byte codes, γ -codes. **SCORING, TERM WEIGHTING:** Parametric and zone indexes, Weighted zone scoring, Learning weights, The optimal weight, Term frequency and weighting, Inverse document frequency, Tf-idf weighting.

CO's-CO3

Self-Learning Topics: Compression Algorithms in Search Engines, Machine Learning for Weight Optimization

UNIT IV:

13Hours

THE VECTOR SPACE MODEL: The vector space model for scoring, Dot products, Queries as vectors, Computing vector scores, Variant tf-Id functions, Sub linear scaling, maximum Formalization, Document and query weighting schemes, Pivoted normalized document length

COs-CO4

Self-Learning Topics: AI-Based Search Ranking Systems, Word Embeddings in Information Retrieval

UNITV:

13Hours

EVALUATIONININ FORMATION RETRIEVAL Information retrieval system evaluation, Standard test collections, Evaluation of unranked retrieval sets, Evaluation of ranked retrieval results.

WEB SEARCH ENGINE: Web search basics, Web characteristics, Index size and estimation, Near-duplicates and shingling, Web crawling and indexes, Distributing indexes, Connectivity servers.

CO's-CO5

Self-Learning Topics: Basics of Google Search Algorithm, Ethical Issues in Web Search and Data Crawling

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/ PO's / PSO's

Modern Information Retrieval Systems and Web Search Engines (Industry practices in search engines, large-scale indexing, web crawling, ranking algorithms, TF-IDF and vector space model applications in systems like Google Search, Elastic search, and enterprise search platforms)

Text Books:

1. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, *An Introduction to Information Retrieval*, 1st Edition, Cambridge University Press, 2008.

Reference Books:

1. G G Chowdhury “Introduction to Modern Information Retrieval” ,3rd Edition, neal-schuman publishers, 2010.
2. Gerald J.Kowalski,Mark T.Maybury,“ Information storage and Retrieval systems: theory and implementation ”, 2nd Edition, kluwer academic publishers, 2009.

Web References:

1. NPTEL(SWAYAM):<https://nptel.ac.in>
2. <http://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board ofStudies(CSE)**

R24CSOE1.1

DATA STRUCTURES

3 0 0 3

(Open Elective-1)

Course Objectives:

1. To introduce fundamental concepts of data structures and algorithms.
2. Be familiar with basic techniques of algorithm analysis.
3. Solve problems using data structures such as linear lists, stacks, queues, hash tables.
4. Explore basic concepts of hashing and apply it to solve problems requiring fast data retrieval and management.
5. To understand real-world applications of data structures.
6. Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B Trees.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs														Do k
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
R24CSOE 01.1	Explain the role of data structures in organizing and data efficiently in algorithms	3	2	2	2	-	-	-	-	-	-	-	-	3	2	L1, L2
R24CSOE 01.2	Develop programs using stacks and Queues to solve related problems.	3	3	2	2	2	-	-	-	-	-	-	-	3	2	L2, L3
R24CSOE 01.3	Use linked lists effectively for dynamic data handling.	3	3	2	2	2	-	-	-	-	-	-	-	3	3	L3, L4
R24CSOE 01.4	Demonstrate the operations such as Insertion, Deletion and Search on Data structures like, AVL trees and B Trees.	3	2	2	2	3	-	-	-	-	-	-	-	3	2	L4, L5
R24CSOE 01.5	Apply trees and graphs in real-world scenarios.	3	2	3	2	3	-	-	-	-	-	-	-	-	-	L5, L6

SYLLABUS**UNIT-I:****14 Hours**

Introduction to Linear Data Structures: Definition and importance of linear data structures, Abstract data types (ADTs) and their implementation, Overview of time and space complexity.

Searching Techniques: Linear & Binary Search.

COs-CO1

Sorting Techniques: Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort.

Self-Learning Topics: Comparison of sorting algorithms with real-time examples.

UNIT II: Stacks, Queues**15 Hours**

Stack Abstract Data Type, Queue Abstract Data Type, Stacks and Queues using arrays, Introduction to Evaluation of Expressions, Evaluating Postfix Expressions, Infix to Postfix and Prefix conversion, Circular Queues using arrays, Dynamically Allocated Storage using pointers.

COs–CO2

Self-Learning Topics: Applications of stack: (Parenthesis checking, Undo/Redo operations)**UNIT III:****15 Hours****Singly Linked Lists:** Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list.**Circular Linked Lists:** Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from Circular Linked Lists**Doubly Linked Lists:** Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from Doubly Linked Lists

COs–CO3

Self-Learning Topics: Skip Lists**UNIT-IV:****13 Hours****Trees:** Introduction to Trees, Properties of Binary Tree, Representation of Binary Tree using Arrays and Linked List, Binary Search Tree – Insertion, Deletion & Traversals, B Trees, B+ Trees. AVL Trees.

COs–CO4

Self-Learning Topics: Segment Trees and Fenwick Trees**UNIT V:****15 Hours****Graphs:** Graph Abstract Data Type, Definitions, Graph Representations, Elementary Graph Operations, Depth First Search, Breadth First Search, Spanning Trees, Minimum Cost Spanning Trees, Prim's and Kruskal's Algorithms, Shortest Paths and Transitive Closure, Single Source All Destination - Dijkstra's Algorithm

COs–CO5

Self-Learning Topics: Graph coloring problems

Board of Studies : Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk (To be Delivered by SMEs from Industries)

Cos

POs/PSOs

1. Graphs Algorithm

CO5

PO1, PO2, PO3

Text Books:

1. Fundamentals of Data Structures in C, 2nd edition, Horowitz, Sahani and Anderson-Freed, Universities Press, 2008.
2. 'Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.
3. 'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, Career Monk Publications.
3. Data Structures: A Pseudo Code Approach with C, 2nd Edition, Richard F. Gilberg, & Behrouz A. Forouzan, Cengage

Web References:

1. <http://www.hackerrank.com/domains/datastructures>
2. http://www.github.com/topics/data_structures_c
3. <http://nptel.ac.in/courses>
4. <http://www.cslibrary.stanford.edu>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand the fundamental concepts of database systems, data models, and database architecture.
2. To design and represent real-world problems using ER diagrams and convert the min to relational schemas.
3. To apply relational model concepts and write SQL queries for data definition, manipulation, and retrieval.
4. To analyze database design using normalization techniques to reduce redundancy and ensure data integrity.
5. To understand transaction management, concurrency control, recovery techniques, and indexing methods for efficient database operations.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 1.2.1	Explain characteristics, architecture, data models, schemas, and components of DBMS.	3	-	-	2	2	-	-	-	-	-	2			L1, L2
R24CSOE 1.2.2	Design ER diagrams and transform the min to relational schemas using constraints and inheritance concepts.	3	3	2	2	3	-	-	-	-	-	2	3	3	L2, L3
R24CSOE 1.2.3	Apply relational algebra, relational calculus, and SQL queries for database creation and manipulation.	3	3	3	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSOE 1.2.4	Analyze and normalize database schemas using functional dependencies and normal forms.	3	3	2	3	3	-	-	-	-	-	2	3	2	L4, L5
R24CSOE 1.2.5	Evaluate transaction management, concurrency control, serializability, deadlock handling, and recovery techniques.	3	3	2	3	3	2	-	-	-	-	3	3	2	L5, L6

SYLLABUS**UNIT I:****15Hours**

Introduction: Characteristics of Database Approach – Advantages of Using DBMS Approach – Data Models, Schemas, and Instances – Three Schema Architecture – Database Languages and Interfaces – The Database System Environment- Centralized and Client-Server Architectures.

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, specialization, generalization using ER Diagrams

Self-Learning Topics: Subclasses, superclass and inheritance **CO's-CO1**

UNIT II: **13Hours**

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. **CO's-CO2**

Self-Learning Topics: Structure of a Relational Database

UNIT III: **15Hours**

Structured Query Language: Introduction, History of SQL Standard – Commands in SQL – Data Types in SQL – Data Definition Language – Selection Operation, Projection Operation –Aggregate Functions – Data Manipulation Language – Table Modification Commands – Table Truncation – Imposition of Constraints – Join Operation – Set Operations – Views – Sub Query–Embedded SQL. **CO's-CO3**

Self-Learning Topics: Advanced Grouping and Aggregation (GROUPBY with ROLLUP, CUBE, and GROUPING SETS, Filtering groups using HAVING)

UNIT IV: **14Hours**

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce Codd-normal form (BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).

Self-Learning Topics: Decomposition Deep Dive (Comparing Lossless vs. Lossy Decomposition, Dependency-Preserving Decomposition: When to compromise) **CO's-CO4**

UNIT V: **13Hours**

Transaction Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Testing for Serializability, lock based, time stamp-based protocols, Deadlocks.

Recovery and Atomicity: Log – Based Recovery – Recovery with Concurrent Transactions – Check Points - Buffer Management – Failure with loss of nonvolatile storage. **CO's-CO5**

Self-Learning Topics: Introduction to Indexing Techniques: B+ Trees, operations on B+ Trees, Hash Based Indexing

Board of Studies:Computer Science and Engineering

Approved in BOS No:03

Approved in ACM No:03

Text Books:

1. **Database System Concepts, Abraham Silberschatz, Henry F.Korth, S.Sudarshan,** McGraw Hill.
2. **Database Management Systems, Raghu Rama krishnan, Johannes Gehrke,TataMcGraw Hill, 3rd Edition.**

Reference Books:

1. **Database Systems Design, Implementation and Management, Peter Rob & Carlos Coronel, Cengage Learning, 7th Edition.**
2. **Fundamentals of Database Systems, Ramez Elmasri,Shamkant B.Navathe,Pearson Education.**
3. **Database Systems:The Complete Book, HectorGarcia-Molina,JeffreyD.Ullman,**

Jennifer Widom,Pearson.

4. **An Introduction to Database Systems**, C.J.Date,A.Kannan,S. Swaminathan,Pearson, 8th Edition.

Web References:

1. <http://www.c4learn.com/>
2. <http://www.geeksforgeeks.org/c/>
3. <http://nptel.ac.in/courses/122104019/>
4. <http://www.learn-c.org/>
5. <https://www.tutorialspoint.com/cprogramming/>

Internal Assessment Pattern

CognitiveLevel	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35%	--
L2	40%	--
L3	25%	25%
L4	--	35%
L5	--	25%
L6		15%
Total(%)	100%	100%

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. Understand the fundamentals of Agile methodology, Agile values, and the Agile Manifesto for effective software development.
2. Apply Agile principles to improve project delivery, customer collaboration, communication, and continuous improvement.
3. Understand Scrum framework, self-organizing teams, sprint planning, and Scrum practices for effective project management.
4. Analyze Extreme Programming (XP) principles and practices for embracing change, improving software quality, and incremental design.
5. Understand Lean, Kanban, and Agile coaching principles for eliminating waste, managing workflow, and continuous process improvement.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE1.3 .1	Explain fundamentals of Agile methodology.	3	2	---	---	1	---	---	---	---	2	---	2	1	L1, L2
R24CSOE1.3 .2	Explain agile principles.	3	2	1	---	1	---	---	---	1	2	---	2	1	L2, L3
R24CSOE1.3 .3	Apply Scrum principles	3	3	3	1	2	---	---	---	2	3	1	3	2	L3, L4
R24CSOE1.3 .4	Apply practices of XP and Incremental design	3	3	3	2	3	---	---	---	2	3	1	3	3	L4, L5
R24CSOE1.3 .5	Develop methods to eliminate waste	3	3	2	2	3	1	1	---	2	3	2	3	3	L5, L6

SYLLABUS**UNIT I:****13 Hours**

LEARNING AGILE: Getting Agile into Your Brain, Understanding Agile Values, No Silver Bullet, Agile to the Rescue, Adding Agile Makes a Difference. A Fractured Perspective, How a Fractured Perspective Causes Project Problems. The Agile Manifesto, Purpose behind Each Practice. Individuals and Interactions Over Processes and Tools, Working Software Over Comprehensive Documentation, Customer Collaboration Over Contract Negotiation, Responding to Change Over Following a Plan, Principles Over Practices. Understanding the Elephant, Methodologies Help You Get It All in Place at Once, Where to Start with a New Methodology

Self-Learning Topics: Agile Software Lifecycle, Agile in Startup Companies.

CO's-CO1**UNIT II:****15Hours**

THE AGILE PRINCIPLES: The 12 Principles of Agile Software, The Customer Is Always

Right...Right? , “Do As I Say, Not As I Said”. Delivering the Project, Better Project Delivery for the EBook Reader Project. Communicating and Working Together, Better Communication for the EBook Reader Project. Project Execution—Moving the Project Along, A Better Working Environment for the E-book Reader Project Team. Constantly Improving the Project and the Team. The Agile Project: Bringing All the Principles Together **CO’s–CO2**

Self-Learning Topics: Agile Requirement Engineering, Agile Project Monitoring Tools.

UNIT III:

14 Hours

SCRUM AND SELF-ORGANIZING TEAMS: The Rules of Scrum, Act I: I Can Haz Scrum?, Everyone on a Scrum Team Owns the Project, The Scrum Master Guides the Team’s Decisions, The Product Owner Helps the Team Understand the Value of the Software, Everyone Owns the Project, Scrum Has Its Own Set of Values, Status Updates Are for Social Networks!, The Whole Team Uses the Daily Scrum, Feedback and the Visibility – Inspection –Adaptation Cycle, The Last Responsible Moment, How to Hold an Effective Daily Scrum. Sprinting into a Wall, Sprints, Planning, and Retrospectives, Iterative or Incremental? The Product Owner Makes or Breaks the Sprint, Visibility and Value, How to Plan and Run an Effective Scrum Sprint.

SCRUM PLANNING AND COLLECTIVE COMMITMENT: Not Quite Expecting the Unexpected, User Stories, Velocity, and Generally Accepted Scrum Practices, Make Your Software Useful, User Stories Help Build Features Your Users Will Use, Conditions of Satisfaction, Story Points and Velocity, Burn down Charts, Planning and Running a Sprint Using Stories, Points, Tasks, and a Task Board. Victory Lap, Scrum Values Revisited, Practices Do Work without the Values (Just Don’t Call It Scrum), is Your Company’s Culture Compatible with Scrum Values

Self-Learning Topics: Agile Metrics and Velocity, Scrum Case Studies in IT Industry. **CO’s–CO3**

UNIT IV:

13 Hours

XP AND EMBRACING CHANGE: Going into Overtime, The Primary Practices of XP, Programming Practices, Integration Practices, Planning Practices, Team Practices, Why Teams Resist Changes, and How the Practices Help. The Game Plan Changed, but We’re Still Losing, The XP Values Help the Team Change Their Mindset, XP Helps Developers Learn to Work with Users, Practices Only “Stick” When the Team Truly Believes in Them, An Effective Mindset Starts with the XP Values, The XP Values, Paved with Good Intentions. The Momentum Shifts, Understanding the XP Principles Helps You Embrace Change, The Principles of XP, XP Principles Help You Understand Planning, XP Principles Help You Understand Practices—and Vice Versa, Feedback Loops.

XP, SIMPLICITY, AND INCREMENTAL DESIGN: Code and Design, Code Smells and Anti-patterns (or, How to Tell If You’re Being Too Clever), XP Teams Look for Code Smells and Fix Them, Hooks, Edge Cases, and Code That Does Too Much. Make Code and Design Decisions at the Last Responsible Moment, Fix Technical Debt by Refactoring Mercilessly, and Use Continuous Integration to Find Design Problems, Avoid Monolithic Design, Incremental Design and the Holistic XP Practices. Teams Work Best When They Feel Like They Have Time to Think, Team Members Trust Each Other and Make Decisions Together. The XP Design, Planning, Team, And Holistic Practices for Man Ecosystem Incremental Design Versus Designing for Reuse, When Units Interact in a Simple Way, the System Can Grow Incrementally; Great Design Emerges from Simple Interactions, Final Score. **COs–CO4**

Self-Learning Topics: Agile Coding Standards, Micro services and Agile Development

UNITV:

13 Hours

LEAN, ELIMINATING WASTE AND SEEING THE WHOLE: Lean Thinking, Commitment, Options Thinking, and Set-Based Development, Creating Heroes and Magical Thinking. Eliminate Waste, use a Value Stream Map to Help See Waste Clearly, gain a Deeper Understanding of the Product, See the Whole, and Find the Root Cause of Problems That You Discover. Deliver As Fast As Possible; Use an Area Chart to Visualize Work in Progress, Control Bottlenecks by Limiting Work in Progress.

KANBAN, FLOW, AND CONSTANTLY IMPROVING: The Principles of Kanban find a Starting Point and Evolve Experimentally from There. Stories Go into the System; Code Comes Out, Improving Your Process with Kanban, Visualize the Workflow, Limit Work in Progress. Measure and Manage Flow, Managing Flow with WIP Limits Naturally Creates Slack. Make Process Policies Explicit So Everyone Is on the Same Page. Emergent Behavior with Kanban.

THEAGILECOACH: Coaches Understand Why People Don't Always Want to Change. The Principles of Coaching.

CO's-CO5

Self-Learning Topics: Agile Process Improvement Techniques, Agile Coaching Skills

Board of Studies : Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/PO's/PSO's

1. Agile and Scrum Practices in Software Industries

This expert talk focuses on the practical implementation of Agile and Scrum methodologies in software industries. The session covers Scrum roles, sprint planning, daily stand-up meetings, backlog management, team collaboration, and real-time project execution practices followed in IT companies.

2. Lean, Kanban, and Agile Transformation in ITProjects

This session explains how Lean principles and Kanban techniques are used in modern IT organizations to improve workflow efficiency, eliminate waste, manage work-in-progress, and achieve continuous improvement during Agile transformation processes.

Text Books:

1. AndrewStellman,JillAlisonHart,*LearningAgile*,O'Reilly,2015

Reference Books:

1. RubinK,*EssentialScrum:ApracticalguidetothemostpopularAgileprocess*, Addison-Wesley, 2013

Web References:

1. NPTEL(SWAYAM):<https://nptel.ac.in>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand operating system overview, functions and services.
2. To analyze process concept, inter process communication and synchronization.
3. Illustrate different conditions for deadlock and their possible solutions.
4. To understand secondary storage management.
5. To understand protection and security.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 1.4.1	Describe operating system overview, functions, and services	3	2	1	1	2	-	-	-	1	-	2	2	1	L1, L2
R24CSOE 1.4.2	Analyze process concept, inter-process communication, and synchronization	3	3	2	1	3	-	-	1	1	-	2	3	2	L2, L3
R24CSOE 1.4.3	Analyze process synchronization mechanisms and dead lock handling techniques.	3	3	2	2	2	-	-	1	1	-	2	3	2	L3, L4
R24CSOE 1.4.4	Describe methods for mapping secondary storage management.	3	2	2	1	3	-	-	-	1	-	2	3	2	L4, L5
R24CSOE 1.4.5	Evaluate protection mechanism and security	3	3	2	1	3	2	2	-	1	-	3	3	3	L5, L6

SYLLABUS**UNITI:****14 Hours**

Introduction to Operating Systems, OS Functions and Services, OS Operations and Types, Computing Environments, Open-Source vs Proprietary OS, System Calls, Types of System Calls, OS Design and Structure, OS Booting Process

CO's-CO1

Self-Learning Topics: Evaluation of operating systems, Comparison of Linux, Windows and Android

UNITII:**13 Hours**

Process Concept, Process States, Process Control Block, Process Scheduling, Context Switching, Inter-Process Communication, Threads, Multithreading Models, Thread Libraries, CPU Scheduling Algorithms, Scheduling Evaluation

CO's-CO2

UNITIII:**13 Hours**

Critical Section Problem, Peterson's Solution, Mutex Locks, Semaphores, Monitors, Classical Synchronization Problems, Deadlock Concept, System Model, Deadlock Characterization, Deadlock Prevention, Deadlock Avoidance

CO's-CO3

Self-Learning Topics: Hardware support for synchronization, Spinlocks and barriers, Resource allocation graphs, Case studies of deadlocks

UNITIV:**14 Hours**

Contiguous Memory Allocation, Paging, Page Tables, Segmentation, Virtual Memory, Demand Paging, Page Replacement Algorithms, Thrashing, File Concepts, Access Methods, Directory Structures, File Allocation Methods

CO's-CO4

Self-Learning Topics: Cache memory and hierarchy, advanced page replacement algorithms

UNITV:**14Hours**

Disk Structure, Disk Scheduling, RAID, Protection Goals, Protection Principles, Access Matrix Model, Security Threats, Network Security Concepts

CO's-CO5

Self-Learning Topics: SSD vs HDD performance, Basics of cryptography in security

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk (To be Delivered by SMEs from Industries)**CO's/PO's/PSO's**

1. Operating Systems in Cloud Computing (Virtualization Techniques in Industry Cloud Platforms (AWS, Azure, GCP), Containerization (Docker, Kubernetes) vs Virtual Machines, Operating System Abstractions in Cloud Architecture) CO1–CO5, PO1, PO2, PO3, PO11
2. Real-Time Operating Systems (RTOS) in Industry Applications (RTOS in Automotive Systems (ADAS, AUTOSAR), RTOS for Industrial Automation and Robotics vs General Purpose OS in Embedded Systems) CO1–CO5, PO1, PO2, PO3, PO11
3. Operating Systems in Cyber security (OS Hardening Techniques in Enterprises, Secure Boot and Trusted Execution Environments, Role of OS in Endpoint Security Solutions) CO1–CO5, PO1, PO2, PO3, PO11

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 10th Edition, John Wiley
2. Operating Systems-A Concept based Approach, D.M. Dhamdhere, 2nd Edition, Tata McGraw-Hill
3. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley

Reference Books:

1. Operating Systems –Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
2. UNIX Concepts and Applications, Sumit Abha Das, 4th Edition, Tata McGraw-Hill
3. Operating System: A Design-Oriented Approach, Charles Patrick Crowley

Web References:

1. <https://archive.nptel.ac.in/courses/106/102/106102132/>
2. https://edurev.in/courses/23057_Operating-System

3. https://www.tutorialspoint.com/operating_system

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Understand the concepts of scripting languages for developing Web Scripting.
2. Create database connections using PHP and build the website for the world.
3. Analyze the internet ware application, security issues and frame work for application.
4. Database driven Web Applications using PHP.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE1.5.1	Identify the basic programming constructs of PHP.	3	2	2	2	2	-	-	-	-	-	-	3	2	L1, L2
R24CSOE1.5.2	Develop programs using functions, strings and arrays.	3	3	2	2	2	-	-	-	-	-	-	3	2	L2, L3
R24CSOE1.5.3	Apply object-oriented principles in PHP	3	3	2	2	2	-	-	-	-	-	-	3	3	L3, L4
R24CSOE1.5.4	Design interactive web pages by using JQuery & AJAX	3	2	2	2	3	-	-	-	-	-	-	3	2	L4, L5
R24CSOE1.5.5	Design data driven applications by using PHP	3	2	3	2	3	-	-	-	-	-	-	3	2	L5, L6

SYLLABUS**UNIT-I:****13Hours**

Introduction to PHP: History, Web Browser, Web Server – XAMPP, Installation and Configuration files. Evaluation of PHP, Basic syntax, defining variable and constant, PHP Data types, Operators and Expressions. Decisions and loops: Making Decisions, doing Repetitive task with looping, mixing decisions and looping with HTML.

COs–CO1

Self-Learning Topics: Advanced XAMPP Configuration

UNIT-II: FUNCTIONS, STRING & ARRAY**15 Hours**

Functions: What is a function, Define a function, Call by value and Call by reference and Recursive functions? **String:** Creating and accessing String, Searching & Replacing String, Formatting String, String Related Library functions **Array:** Anatomy of an Array, Creating Index based and Associative array, Accessing array element, Looping with Index based array, Looping with associative array using each() and foreach(), some useful library functions.

COs–CO2

Self-Learning Topics: Building Reusable Utilities

UNIT-III:**15 Hours**

ADVANCE PHP: Introduction: Objects, declaring a class, the new keyword and constructor,

Destructor, Access method and properties using this variable. Public, private, protected properties and methods, Static properties and method, Class constant, Inheritance & code, reusability, Polymorphism, Parent::&self::keyword, Instance of operator, Abstract method and class, Interface, Final. Exception Handling: Understanding Exception and error. Try, catch, and throw.**COs-CO3**
Self-Learning Topics: Evaluate OOP vs. procedural programming

UNIT-IV: PHP SCRIPT 12Hours

JQuery: Introduction to JQuery, Validation using JQuery, JQuery Forms, JQuery Examples.

AJAX: Introduction to AJAX, PHP with AJAX, Working with database. **COs-CO4**

Self-Learning Topics: Evaluate AJAX vs traditional page reloads

UNIT-V: PHP WEB SERVICES 15Hours

Handling HTML form with PHP: Capturing Form Data, Dealing with Multi-value file, generating File uploaded form, redirecting a form after submission. Web Features: Sessions, Forms GET and POST data, Cookies, HTTP Headers. Database Connectivity with

MySQL: Introduction to RDBMS, Connection with MySQL Database, Performing basic database operations (DML) (Insert, Delete, Update, Select),Setting query parameter, Executing query.

Self-Learning Topics: Perform **CRUD** operations using **PHP & MySQL** **COs-CO5**

Board of Studies:Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)	COs	POs/PSOs
1. Evaluate session vs. cookies for user management	CO5	PO1,PO2,PO3

Text Books:

1. Steven Holzner, —PHP:The Complete Reference, McGraw-HillEducation,2007.
2. Kevin Tatroe, Peter MacIntyre, RasmusLerdorf,—Programming in PHP, O'Reilly,3rd Edition, 2013.
3. LukeWelling,LauraThomson,—PHP and MySQL Web Development, Pearson Education, 5th Edition.

Reference Books:

1. HTML5 Black Book, CoversCSS3, JavaScript, XML, XHTML, AJAX, PHP and jQueryDT Editorial Services Dream Tech Publications.
2. Lynn Beighley, Michael Morrison, —Head First PHP & MySQL: ABrain-FriendlyGuidel, O'Reilly, 1st Edition.
3. DavidPowers,—PHP Solutions: Dynamic Web Design Made Easy, Après, 3rdEdition.

Web References:

1. https://www.w3schools.com/php/php_intro.asp
2. https://www.tutorialspoint.com/php/php_introduction.htm
3. <https://www.tutorialrepublic.com/php-tutorial/php-variables.php>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

3 0 0 3 (Common to CSE)

Course Objectives:

1. Implementation of a compiler for a basic language
2. Lex/Yacc specifications for designing frontend of a compiler .
3. Generate MIPS instruction set

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPC17.1	Understand the different phases of compilation and the working of compilers like gcc, clang etc	3	2	-	-	1	2	1	1	L1,L2,L4
R24CSPC17.2	Implement lexical analyzer for any language	2	3	3	-	2	3	3	2	L2,L3
R24CSPC17.3	Implement parser for any language	2	2	3	2	3	3	3	2	L3,L4
R24CSPE17.4	Implement 3-address code generator for simple programming constructs	1	3	2	2	2	2	3	2	L3,L4
R24CSPC17.5	Implement MIPS code generator by considering simple programming constructs	1	2	1	1	2	1	2	3	L4,L5

List of experiments

1. Write a C program to identify different types of Tokens in a given Program.
2. Write a Lex Program to implement a Lexical Analyzer using Lex tool.
3. Write a C program to Simulate Lexical Analyzer to validating a given input String.
4. Write a C program to implement the Brute force technique of Top down Parsing.
5. Write a C program to implement a Recursive Descent Parser.
6. Write C program to compute the First and Follow Sets for the given Grammar.
7. Write a C program for eliminating the left recursion and left factoring of a given grammar
8. Write a C program to check the validity of input string using Predictive Parser.
9. Write a C program for implementation of LR parsing algorithm to accept a given input string.
10. Write a C program for implementation of a Shift Reduce Parser using Stack Data Structure to accept a given input string of a given grammar.

11. Simulate the calculator using LEX and YACC tool.
12. Generate YACC specification for a few syntactic categories.
13. Write a C program for generating the three address code of a given expression/statement.
14. Write a C program for implementation of a Code Generation Algorithm of a given expression/statement.

Text Books:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson Publishers, 2007.
2. John R Levine, Tony Mason, Doug Brown, "Lex and Yacc", Orielly, 2nd Edition, 2009.

Web References:

- 1.<https://llvm.org/>
- 2.<https://gcc.gnu.org/>
- 3.https://www.dsi.unive.it/~gasparetto/materials/MIPS_Instruction_Set.pdf

**Chairperson
Board of Studies (CSE)**

R24CSPC18

COMPUTERNETWORKSLAB

0 0 3 1.5

(ComputerScienceandEngineering)

Course Objectives:

1. Understand and implement basic networking protocols and algorithms.
2. Develop network programs using C (client-server, routing).
3. Simulate and analyze network performance and behavior.
4. Use tools to capture and study network traffic.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs														
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	Do K	
R24CS PC18.1	Develop and implement computer networking concepts such as framing methods, error detection techniques, routing algorithms, socket programming, and sliding window protocols using C programming.	3	3	3	1	2	0	0	0	1	0	0	3	2	L1, L2	
R24CS PC18.2	Analyze and simulate network routing protocols, broadcast mechanisms, & networking algorithms using C programming and use simulation tools such as NS2/NS3, Packet Tracer, or equivalent Simulators to find the network performance parameters	3	3	2	3	2	0	0	0	1	1	0	3	3	L2, L3	
R24CS PC18.3	Capture, monitor, filter, and troubleshoot network traffic using networking tools such as Wireshark, raw sockets, ARP, and DNS client-server applications to understand network communication and diagnostics.	2	3	1	2	2	0	0	0	1	3	1	3	3	L3	

Board of Studies: Computer Science and Engineering Approved

in BOS No: 03

Approved in ACM No: 03

Developing the following programs:

- | | |
|---|----------------------|
| Week1: | CO's:CO1 |
| 1. Write a C Program to implement the data link layer framing methods such as: | |
| i) Character stuffing | ii) Bit stuffing |
| Week2: | CO's:CO1 |
| 2. Write a C Program to implement on a data set of characters the three CRC polynomials – CRC-12, CRC-16 and CRC CCIP. | |
| Week3: | CO's:CO1 |
| 3. Write a C Program to implement Dijkstra's algorithm to compute the shortest path through a graph. | |
| Week4: | CO's:CO1 |
| 4. Write a C Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes). | |
| Week5: | CO's:CO1 |
| 5. Write a C Program of client-server application for chat using UDP or TCP | |
| Week6: | CO's:CO1 |
| 6. Write a C program to perform sliding window protocol for | |
| i) Go Back-N | ii) Selective Repeat |
| Week7: | CO's:CO2 |
| 7. Write a C Program to Simulate the Open Shortest Path First routing protocol based on the cost assigned to the path | |
| Week8: | CO's:CO2 |
| 8. Write a C Program to implement Broadcast tree by taking subnet of hosts. | |
| Week9: | CO's:CO2 |
| 9. Write a C Program to Implementing Routing Protocols using border gateway protocol (BGP) | |
| Week10: | CO's:CO2 |
| 10. Write a C Program to Get the MAC or Physical address of the system using Address Resolution Protocol. | |
| Week11: | CO's:CO3 |
| 11. Write a C program to develop a DNS client server to resolve the given hostname | |
| Week12: | CO's:CO3 |
| 12. Write a C Program to use raw socket to implement packet capturing and filtering | |
| Week13: | CO's:CO3 |
| 13. Install Wireshark Tool on PC and | |
| a) Capture network traffic | |

- b) examine the traffic captured by Wireshark

Week14:

CO's:CO3

14. Install Wireshark Tool on PC and
- Filter and examine only ICMP traffic using Wireshark Tool
 - Determine default gateway address of your network using Wireshark Tool

Week15:

CO's:CO3

15. Simulate a three nodes' point-to-point network with duplex links between them. Set the queue size, varies the bandwidth and find the number of packets dropped. ns2/ns3/CISCO Packet Tracer /OPNET/any other network simulator may be used for simulation experiments.

Reference Books:

1. W.Richard Stevens – *UNIX Network Programming*, 2nd Edition, 1998
2. Donahoo & Calvert – *TCP/IP Sockets in C*, 2nd Edition, 2009
3. Behrouz A. Forouzan – *TCP/IP Protocol Suite*, 4th Edition, 2010
4. William Stallings – *Data and Computer Communications*, 10th Edition, 2013
5. Yashavant Kanetkar – *Let Us C*, 16th Edition, 2020 (varies by print)
6. Douglas Comer – *Internetworking with TCP/IP*, Vol.1, 6th Edition, 2014

Online Learning Resources:

1. <https://www.geeksforgeeks.org/computer-network-tutorials/>
2. https://www.tutorialspoint.com/data_communication_computer_network/index.htm
3. <https://www.cisco.com/c/en/us/solutions/enterprise-networks/what-is-computer-networking.html>
4. <https://nptel.ac.in/courses/106105081> (Computer Networks – NPTEL)
5. <https://www.javatpoint.com/computer-network-tutorial>
6. <https://www.wireshark.org/docs/> (Wireshark documentation)

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. Make use of router, template engine and authentication using sessions to develop application in ExpressJS.
2. Build a single page application using RESTful APIs in ExpressJS
3. Apply router and hooks in designing ReactJS application

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSSC03 .1	Apply core concepts of ExpressJS to build web applications	3	2	-	-	1	2	1	1	L1,L2, L4
R24CSSC03 .2	Design and develop web applications using ExpressJS by connecting to MongoDB with Mongoose	2	3	3	-	2	3	3	2	L2,L3
R24CSSC03 .3	Creating reusable components using both function-based and class-based approaches.	2	2	3	2	3	3	3	2	L3,L4
R24CSSC03 .4	Implement dynamic user interfaces in ReactJS	1	3	2	2	2	2	3	2	L3,L4
R24CSSC03.5	Install and configure MongoDB (including Atlas) and perform essential database operations	1	2	1	1	2	1	2	3	L4,L5

List of experiments

1. ExpressJS – Routing, HTTP Methods, Middleware.

Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.

Write a program to accept data, retrieve data and delete a specified resource using http methods.

Write a program to show the working of middleware.

2. ExpressJS – Templating, Form Data

Write a program using templating engine.

Write a program to work with form data.

3. ExpressJS – Cookies, Sessions, Authentication

Write a program for session management using cookies and sessions.

Write a program for user authentication.

4. ExpressJS – Database, RESTful APIs

Write a program to connect MongoDB database using Mongoose and perform CRUD operations.

Write a program to develop a single page application using RESTful APIs.

5. ReactJS – Render HTML, JSX, Components – function & Class

Write a program to render HTML to a web page.

Write a program for writing markup with JSX.

Write a program for creating and nesting components (function and class).

6. ReactJS – Props and States, Styles, Respond to Events

Write a program to work with props and states.

Write a program to add styles (CSS & Sass Styling) and display data.

Write a program for responding to events.

7. ReactJS – Conditional Rendering, Rendering Lists, React Forms

Write a program for conditional rendering.

Write a program for rendering lists.

Write a program for working with different form fields using react forms.

8. ReactJS – React Router, Updating the Screen

Write a program for routing to different pages using react router.

Write a program for updating the screen.

9. ReactJS – Hooks, Sharing data between Components

Write a program to understand the importance of using hooks.

Write a program for sharing data between components.

10. MongoDB – Installation, Configuration, CRUD operations

Install MongoDB and configure ATLAS

Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

11. MongoDB – Databases, Collections and Records

Write MongoDB queries to Create and drop databases and collections.

Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate().

Text Books:

AIETM | R24| CSE| R24CSSC03| Full Stack Development-2 LAB

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasam Subramanian, 2nd edition, APress, O'Reilly.
2. Node.Js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich, Manning Publications. (Chapters 1-11)
3. React Quickly, AzatMardan, Manning Publications (Chapters 1-8, 12-14)

Web References:

1. ExpressJS - <https://www.tutorialspoint.com/expressjs>
2. ReactJS - <https://www.w3schools.com/REACT> (and) <https://react.dev/learn#>
3. MongoDB - <https://learn.mongodb.com/learning-paths/introduction-to-mongodb>

**Chairperson
Board of Studies (CSE)**

Course Objectives:

The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSES08.1	Define the concepts related to design thinking.	3	-	-	-	-	3	1	-	L1,L2
R24CSES08.2	Explain the fundamentals of Design Thinking and innovation	2	2	-	2	3	2	3	-	L1,L2
R24CSES08.3	Apply the design thinking techniques for solving problems in various sectors.	2	2	-	2	3	2	2	-	L3
R24CSES08.4	Analyse to work in a multidisciplinary environment	2	-	-	2	3	1	2	-	L4
R24CSES08.5	Evaluate the value of creativity. Formulate specific problem statements of real time issues	1	3	-	1	2	-	1	-	L3,L5

SYLLABUS**UNIT - I:****Introduction to Design Thinking**

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.

COs-CO1**UNIT-II:**

Design Thinking Process: Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development.

Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

COs-CO2

UNIT – III:

Innovation: Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations- Creativity to Innovation- Teams for innovation- Measuring the impact and value of creativity.

Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation. **COs–CO3**

UNIT - IV:

Product Design: Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications- Innovation towards product design- Case studies

Activity: Importance of modelling, how to set specifications, Explaining their own product design. **COs–CO4**

UNIT - V:

Design Thinking in Business Processes: Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs- Design thinking for Startups- Defining and testing Business Models and Business Cases- Developing & testing prototypes.

Activity: How to market our own product, About maintenance, Reliability and plan for startup. **COs–CO5**

Text Books:

1. Tim Brown, Change by design, Harper Bollins (2009)
2. Idris Mootee, Design Thinking for Strategic Innovation, 2013, John Wiley & Sons.

Reference Books:

1. David Lee, Design Thinking in the Classroom, Ulysses press
2. Shruti N Shetty, Design the Future, Norton Press
3. William Lidwell, Universal Principles of Design- Kritinaholden, Jill Butter.
4. Chesbrough. H, The Era of Open Innovation – 2013

Web References:

1. <https://nptel.ac.in/courses/110/106/110106124/>
2. <https://nptel.ac.in/courses/109/104/109104109/>
3. https://swayam.gov.in/nd1_noc19_mg60/preview

**Chairperson
Board of Studies (CSE)**

Course Title: Advanced Aptitude and Communication Skills

Course Duration: 16 Weeks | 3 Hours per Week

Course Outcomes (COs)

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

- **CO1:** Apply quantitative aptitude concepts to solve problems related to arithmetic, percentages, ratio, averages, profit and loss, interest, and data interpretation.
- **CO2:** Analyze and solve logical reasoning and analytical problems involving coding-decoding, puzzles, seating arrangements, syllogisms, and data sufficiency.
- **CO3:** Apply mathematical and analytical techniques to solve campus recruitment and competitive examination problems efficiently.
- **CO4:** Develop professional communication skills including spoken English, workplace communication, email writing, customer interaction, and digital communication.
- **CO5:** Demonstrate employability skills, presentation abilities, interpersonal communication, workplace etiquette, personality development, and interview readiness for corporate careers.

PART – A : APTITUDE, REASONING & ANALYTICAL SKILLS

UNIT–1: Quantitative Aptitude & Problem Solving

Percentages, ratio and proportion, averages, profit and loss, simple and compound interest, time and work, pipes and cisterns, time-speed-distance, boats and streams, permutations and combinations, and probability.

Practice Activities

Shortcut techniques, aptitude worksheets, quantitative mock tests, company-level aptitude practice, and speed calculation exercises.

UNIT–2: Logical Reasoning & Data Interpretation

Coding-decoding, blood relations, number series, seating arrangements, syllogisms, puzzles, statement and conclusions, logical sequences, data sufficiency, non-verbal reasoning, tables, charts, graphs, and data interpretation techniques.

Practice Activities

Reasoning mock tests, puzzle-solving sessions, sectional practice tests, analytical exercises, and company-specific reasoning assessments.

UNIT–3: Basic Mathematics for Placements

Geometry basics, triangles, circles, trigonometric ratios, heights and distances, angles, mensuration concepts, and mathematical reasoning frequently used in aptitude examinations.

Practice Activities

Formula applications, trigonometry exercises, mathematical problem-solving, and placement-oriented practice sessions.

PART – B : PROFESSIONAL COMMUNICATION & EMPLOYABILITY SKILLS

UNIT-4: Professional & Corporate Communication Skills

Sentence structure, vocabulary development, spoken English, pronunciation, fluency development, grammar correction, listening skills, workplace communication, customer handling, telephone etiquette, voice modulation, active listening, chat communication, and professional email writing.

Practice Activities

Speaking practice, pronunciation drills, workplace conversations, mock voice calls, email drafting, role-play activities, and communication fluency sessions.

UNIT-5: Employability, Personality & Interview Skills

Versant test pattern, listening comprehension, reading fluency, sentence mastery, interpersonal skills, teamwork, workplace etiquette, presentation skills, personality development, self-introduction techniques, HR interview preparation, confidence building, and professional grooming.

Practice Activities

Versant mock tests, mock interviews, presentation activities, teamwork exercises, self-introduction practice, placement mock drives, and workplace etiquette sessions.

Reference Text Books

1. R.S. Aggarwal, *Quantitative Aptitude for Competitive Examinations*, S. Chand Publishing.
2. Meenakshi Raman and Sangeeta Sharma, *Technical Communication: Principles and Practice*, Oxford University Press.
3. Barun K. Mitra, *Personality Development and Soft Skills*, Oxford University Press.

Reference Books

1. M. Tyra, *Magical Book on Quicker Maths*, BSC Publishing.
2. S.P. Bakshi, *Objective General English*, Arihant Publications.
3. Edgar Thorpe, *Test of Reasoning*, Pearson Education.

Web Links

1. IndiaBix Aptitude and Reasoning
2. PrepInsta Placement Preparation
3. British Council English Learning

Placement Target Companies

Students completing this course will be prepared for entry-level roles and campus recruitment opportunities in software, IT services, analytics, AI support, cloud, and digital engineering domains.

IT Services & Consulting Companies

- Tata Consultancy Services (TCS)
- Infosys
- Wipro
- HCLTech
- Tech Mahindra
- Accenture
- Cognizant
- Capgemini

Product & Technology Companies

- Zoho Corporation
- Freshworks
- Oracle
- IBM
- Deloitte

AI, Cloud & Data-Oriented Roles

- AI Support Associate
- Data Analyst Trainee
- Junior Software Developer
- Cloud Support Associate
- Technical Support Engineer
- Prompt Engineering Associate
- Graduate Engineer Trainee (GET)
- Associate System Engineer
- Software Testing Associate

Course Strengths

Industry-Oriented Curriculum

Designed based on current campus recruitment requirements and emerging AI-driven industry trends.

Placement-Focused Learning

Covers coding, aptitude-oriented programming, databases, AI tools, cloud basics, and interview preparation essential for placement rounds.

Hands-on Practical Training

Includes coding practice, SQL exercises, API integration, AI tools usage, GitHub workflows, and mini-project development.

Emerging Technology Exposure

Introduces students to AI, LLM applications, prompt engineering, cloud platforms, and MLOps fundamentals.

Interview & Career Readiness

Enhances technical interview confidence, teamwork, project presentation, resume preparation, and problem-solving ability.

Multi-Domain Career Opportunities

Supports preparation for software development, AI support, data engineering, cloud support, and IT services roles.

R24CSMC03|

Constitution of India

3 0 0 3 (Common to CSE)

Course Objectives:

1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSMC03.1	Understand historical background of the constitution making and its importance for building a democratic India.	3	-	-	-	-	3	1	-	L1,L2
R24CSMC03.2	Understand the functioning of three wings of the government i.e., executive, legislative and judiciary.	2	2	-	2	3	2	3	-	L1,L2
R24CSMC03.3	Understand the value of the fundamental rights and duties for becoming good citizen of India.	2	2	-	2	3	2	2	-	L3
R24CSMC03.4	Analyze the decentralization of power between central, state and local self-government.	2	-	-	2	3	1	2	-	L4
R24CSMC03.5	Apply the knowledge in strengthening of the constitutional institutions like CAG, Election Commission and UPSC for sustaining democracy.	1	3	-	1	2	-	1	-	L3,L5

SYLLABUS

UNIT-I: History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working)

Philosophy of the Indian Constitution- Preamble, Salient, Features

COs-CO1

UNIT-II: Contours of Constitutional Rights & Duties: Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties. **COs-CO2**

UNIT-III: Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, **Executive-** President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions **COs-CO3**

UNIT-IV: Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: ZilaPachayat, Elected officials and their roles, CEO ZilaPachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy **COs-CO4**

UNIT-V: Election Commission: Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women. **COs-CO5**

Text Books:

1. The Constitution of India, 1st Edition, (Bare Act), Government Publication, 1950
2. Framing of Indian Constitution, 1st Edition, Dr. S. N. Busi, Dr. B. R. Ambedkar 2015

Reference Books:

1. Indian Constitution Law, 7th Edition, M. P. Jain, Lexis Nexis, 2014

**Chairperson
Board of Studies (CSE)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering

Regulation: R24

III Year II Semester - Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC19	Artificial Intelligence & Machine Learning	3	0	0	3
2	PC	R24CSPC20	Cryptography & Network Security	3	0	0	3
3	PE-2	R24CSPE2.1 R24CSPE2.2 R24CSPE2.3 R24CSPE2.4	1. Cloud Computing 2. Cyber Security 3. DevOps 4. Augmented Reality and Virtual Reality 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	PE-3	R24CSPE3.1 R24CSPE3.2 R24CSPE3.3 R24CSPE3.4	1. Software Project Management 2. Big Data Analytics 3. Distributed Systems 4. Foundation of Data Science 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	OE-2		OPEN ELECTIVE-2	3	0	0	3
6	PC	R24CSPC21	Artificial Intelligence & Machine Learning	0	0	3	1
7	PC	R24CSPC22	Cryptography & Network Security Lab	0	0	3	1
8	SC	R24CSSC04	Competitive Coding Lab	0	0	2	1
9	HS	R24HS15	Corporate Readiness for Emerging Technologies	2	0	0	1
10	MC	R24MC07	Value Education	2	0	0	0
				19	00	8	19

Mandatory Industry Internship of 08 weeks duration during summer vacation

Category	Courses	Credits
PC-Professional Core Courses	4	8
PE- Professional Elective	2	6
OE-Open Elective	1	3
SEC-Skill Enhancement course	1	1
HS-Humanities and Social Science including Management Course	1	1
MC-Mandatory Course	1	0
Total	10	19

R24CSPC19

Artificial Intelligence and Machine Learning

3 0 0 3

(Computer Science and Engineering)

Course Objectives:

1. To train the students to understand different types of AI agents.
2. To understand various AI search algorithms.
3. Fundamentals of knowledge representation, building of simple knowledge-based systems and to apply knowledge representation.
4. To introduce the basic concepts and techniques of machine learning and the need for Machine learning techniques for real world problem.
5. To provide understanding of various Machine learning algorithms and the way to evaluate the performance of ML algorithms.

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSPC 19.1	Understand the informed and uninformed problem types and apply search strategies to solve them.	3	3	1	1	1	-	-	-	-	-	2	3	1	L1, L2
R24CSPC 19.2	Apply difficult real life problems in a state space representation so as to solve those using AI techniques like searching and game playing.	3	3	2	2	1	-	-	-	1	-	2	3	2	L2, L3
R24CSPC 19.3	Apply machine learning techniques in the design of computer systems	3	3	3	2	2	-	-	-	1	-	2	3	3	L3, L4
R24CSPC 19.4	To differentiate between various categories of ML algorithms	3	2	1	1	1	-	-	-	-	-	2	3	1	L4, L5
R24CSPC 19.5	Design and make modifications to existing machine learning algorithms.	3	3	3	2	2	1	-	-	1	-	3	3	3	L5, L6

SYLLABUS**UNIT I:****15Hours**

Introduction: AI problems, Agents and Environments, Structure of Agents, Problem Solving Agents Basic Search Strategies: Problem Spaces, Uninformed Search (Breadth-First, Depth-First Search, Depth-first with Iterative Deepening), Heuristic Search (Hill Climbing, Generic Best-First, A*), Constraint Satisfaction (Backtracking, Local Search)

CO's-CO1

Self-Learning Topics: Real-world applications of search algorithms (Google Maps, games)

UNIT II:**15Hours**

Advanced Search: Constructing Search Trees, Stochastic Search, AO*Search Implementation,

Minimax Search, Alpha-Beta Pruning, Basic Knowledge Representation and Reasoning: Propositional Logic, First-Order Logic, Forward Chaining and Backward Chaining, Introduction to Probabilistic Reasoning, Bayes Theorem **CO's–CO2**

Self-Learning Topics: Game playing AI(Chess,Tic-Tac-Toe using Minimax), Bayesian Networks

UNITIII: 14Hours

Machine-Learning: Introduction, Machine Learning Systems, Forms of Learning: Supervised and Unsupervised Learning, reinforcement – theory of learning – feasibility of learning – Data Preparation–training versus testing and split. **CO's–CO3**

Self-Learning Topics: Data preprocessing techniques (normalization, encoding), Scikit-learn library

UNIT IV: 13Hours

Supervised Learning: Regression: Linear Regression, Multi Linear Regression, Polynomial Regression, Logistic Regression, Non-linear Regression, Model evaluation methods.

Classification: Support Vector Machines(SVM), Naïve Bayes classification **COs–CO4**

Self-Learning Topics: Real-world ML case studies (spam detection, fraud detection), ROC curves

UNIT V: 13Hours

Unsupervised Learning: Nearest neighbor models, K-means, Clustering around k medoids silhouette, Hierarchical clustering, k-d trees, clustering trees, learning ordered rule lists, Learning unordered rule. **Reinforcement learning:** State and Action Spaces. **CO's–CO5**

Self-Learning Topics: Deep Learning, Applications of Reinforcement Learning (robotics, gaming)

Board of Studies : Computer Science and Engineering

Approved in BoSNo:03

Approved in ACM No:03

Text Book:

1. Russell,S.andNorvig,P, Artificial Intelligence: A Modern Approach, Third Edition, Prentice Hall, 2010.
2. MACHINE LEARNING An Algorithmic Perspective 2nd Edition, StephenMarsland,2015, byTaylor & Francis Group, LLC Introduction to Machine Learning, The Wikipedia Guide

Reference Books:

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivasankar B. Nair, The McGraw Hill publications, Third Edition, 2009.
2. Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education, 6th ed., 2009.
3. Introduction to Machine Learning, Second Edition, Ethem Al paydn,theMITPress, Cambridge, Massachusetts, London, England.
4. Machine Learning,TomM.Mitchell,McGraw-HillScience,ISBN: 0070428077
5. Understanding Machine Learning: From Theory to Algorithms, c2014byShaiShalev Shwartz and Shai Ben-David, Published 2014 by Cambridge University Press.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--

AIETM|R24|CSE|R24CSPC19|Artificial Intelligence and Machine Learning

L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand the fundamentals of security including security attacks services mechanisms and network Security concepts
2. To understand and apply public key cryptography hash functions and message authentication techniques for ensuring data security and integrity
3. To learn cryptographic hash functions and message authentication techniques including SHA and HMAC for ensuring data integrity and security
4. To learn electronic mail security techniques including SMIME for secure communication
5. To analyze IP security mechanisms including encapsulating security payload and Internet Key Exchange for secure data transmission

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSPC 22.1	Apply cryptography and security principles to analyze attacks, design systems	3	2	-	-	2	-	-	-	-	-	1	2	1	L1, L2
R24CSPC 22.2	Analyze classical ciphers and apply DES,AES algorithms to ensure strong, secure data encryption	3	3	2	1	3	-	-	-	-	-	1	3	2	L2, L3
R24CSPC 22.3	Apply RSA, SHA, HMAC techniques ensuring data security, integrity, authentication using public key cryptography principles	3	3	2	2	2	-	-	-	-	-	1	3	2	L3, L4
R24CSPC 22.4	Apply digital signatures, key management, authentication mechanisms ensuring secure communication and verified data authenticity	3	2	3	2	3	1	1	-	-	-	2	3	2 3	L4, L5
R24CSPC 22.5	Analyze network, internet security mechanisms including email, IP protocols to protect data during transmission securely	3	3	3	2	3	2	1	1	1	1	2	3	3	L5, L6

SYLLABUS

UNIT I:

12Hours

Introduction to Security: Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, Model for Network Security

Mathematics of Cryptography: Algebraic Structures (Groups, Rings, Fields, Galois Fields), Divisibility and the Division Algorithm, Euclidean Algorithm, Prime Numbers (basis of cryptography), Fermat's and Euler's Theorems (used in encryption) **CO's-CO1**

Self-Learning Topics: Testing for Primality, Chinese Remainder Theorem (optimization in computations), Discrete Logarithms (hard problem used for security)

UNIT II:

12Hours

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Steganography

Block Ciphers: Traditional Block Cipher Structure, The Data Encryption Standard, The Strength of DES, Block Cipher Design Principles, Advanced Encryption Standard, AES Structure, AES Transformation Functions, AES Key Expansion, Block Cipher Modes of Operation **CO's-CO2**

Self-Learning Topics: Multiple Encryption and Triple DES, comparison of AES, DES, and modern ciphers

UNIT III:

15Hours

Public-Key Cryptography: Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie Hellman Key Exchange, Elgamal Cryptographic System, Elliptic Curve Cryptography

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Requirements and Security, Secure Hash Algorithm (SHA)

Message Authentication Codes: Requirements for Message Authentication Codes, HMAC

CO'sCO3

Self-Learning Topics: Implementation of RSA algorithm and Diffie Hellman key exchange, Study of Secure Hash Algorithm and HMAC for message authentication

UNIT IV:

13Hours

Digital Signatures: Digital Signatures, Elgamal Digital Signature Scheme, Schnorr Digital Signature, Scheme Elliptic Curve Digital Signature Algorithm

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public-Key Infrastructure

User Authentication: Remote User-Authentication Principles, Remote User-Authentication Using Symmetric Encryption, Remote User-Authentication Using Asymmetric Encryption: **CO's-CO4**

Self-Learning Topics: Kerberos, NIST Digital Signature Algorithm digital signatures in blockchain systems, , biometric authentication systems.

UNIT V:

15Hours

Network and Internet Security

Electronic Mail Security: Internet Mail Architecture, Email Formats, Email Threats and Comprehensive Email Security, S/MIME.

FIREWALLS: The Need for Firewalls, Firewall Characteristics, Types of Firewalls, Firewall Configurations. **CO's-CO5**

Self-Learning Topics: SSL/TLS vs IPsec, VPN security, Email encryption tools

Board of Studies : Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Text Book:

1. Cryptography and Network Security, William Stallings, 8th Edition, Pearson Education,

Reference Books:

1. Cryptography and Network Security and CyberLaws, Bernard L. Menezes, Ravinder Kumar, Cengage Learning.
2. Cryptography and Network Security, Behrouz A Forouzan, Debdeep Mukhopadhyaya, 3rd Edition, Mc-GrawHill.
3. Network Security Illustrated, Jason Albanese, Wes Sonnenreich, McGrawHill.

Web References:

1. <https://nptel.ac.in/courses/106105162>
2. <https://nptel.ac.in/courses/106/105/106105031>
3. <https://www.geeksforgeeks.org/computer-networks/cryptography-tutorial/>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

1.

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand the fundamentals of cloud computing and virtualization.
2. To learn cloud architecture and service models like IaaS, PaaS and SaaS.
3. To understand cloud services, scaling and disaster recovery mechanisms.
4. To gain knowledge of cloud application platforms and development tools.
5. To explore real-world cloud computing systems and industry applications.

Course Outcomes:

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

Mapping with POs and PSOs															Dok
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSPE2 .1.1	Understand the concept of virtualization and how this has enabled the development of Cloud Computing.	3	2	1	-	2	-	-	-	-	-	-	3	2	L1, L2
R24CSPE2 .1.2	Explain cloud architecture and different service models	3	3	2	-	1	-	-	-	-	-	-	2	3	L2, L3
R24CSPE2 .1.3	Analyze scaling, security, and disaster recovery in cloud systems.	3	3	2	2	2	-	-	-	-	1	-	2	3	L3, L4
R24CSPE2 .1.4	Design simple cloud-based applications using platform tools.	2	2	3	2	3	-	-	-	-	1	2	3	3	L4, L5
R24CSPE2 .1.5	Identify and evaluate real-world cloud platforms and applications.	2	2	1	1	3	-	-	-	-	1	2	2	2	L5, L6

SYLLABUS

UNIT I:

15 Hours

Introduction to Cloud Computing & Virtualization Basics of Cloud Computing, what is Cloud Computing, Vision and need of cloud, Basic cloud characteristics, Benefits of cloud computing Virtualization Basics Introduction to virtualization, Virtual machines (VM concept), Basic types of virtualizations, Introduction to hypervisors, Simple tools: VMware Microsoft Hyper-V

CO's –CO1

Self-Learning Topics: Evolution of Cloud Computing (history and development)

UNIT II:

13 Hours

Cloud Architecture: Cloud Basics, Cloud architecture Overview, Cloud service models: IaaS, PaaS, SaaS Types of Cloud: Public cloud, Private cloud, Hybrid cloud, Basic Concepts: Scalability, Fault tolerance, Introduction to cloud security

CO's –CO2

Self-Learning Topics: Micro services vs Monolithic architecture (basic idea)

UNIT III:

14 Hours

Cloud Services & Disaster Recovery: Cloud Services Basics, Storage as a Service, Database as a Service Software/Process as a Service, Scaling Basics: What is scaling, Basic capacity planning, Disaster Recovery: Backup concept, Disaster recovery meaning, Importance of data recovery

CO's–CO3

Self-Learning Topics: Real-time backup systems in cloud (Google backup, AWS backup basics)

UNIT IV: 15Hours

Cloud Application Platform (Aneka Basics): Platform Overview, Introduction to Aneka, Basic idea of cloud application Platforms, Basic Components: Container concept, Simple service layers, Introduction to SDK Deployment Models: Private cloud, public cloud, Hybrid cloud

CO's–CO4

Self-Learning Topics: Cloud SDK usage and simple application deployment concepts

UNIT V: 13 Hours

Cloud Applications & Industry Basics: Cloud Applications, Basic idea of cloud applications, Scientific applications (health, biology basics), Business applications (CRM, ERP basics), Cloud Platforms, Introduction to: Amazon Web Services, Google App Engine, and Microsoft Azure:Basic Idea, what cloud platforms do, Simple usage of cloud services.

CO's-CO5

Self-Learning Topics: Server fewer computing basics (Function as a Service concept)

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Text Book:

1. Mastering Cloud Computing by Rajkumar Buyya,Christian Vecchiola,andS.ThamaraiSelvi from TMH 2013.
2. GeorgeReese Cloud Application Architectures, First Edition, O'Reilly Media2009.

Reference Books:

1. Cloud Computing and SOA Convergence in Your Enterprise A Step-by-Step Guide byDavid S. LinthicumfromPearson2010.
2. Cloud Computing 2nd Edition by Dr.Kumar Saurabh fromWileyIndia 2012.
3. Cloud Computing–web-based Applications that change the way you work and collaborate Online – Michael Miller. Pearson Education.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To introduce the fundamental concepts of cyber security, including cyber threats, cybercrimes, and the need for security policies.
2. To provide knowledge of web security principles and network vulnerabilities, along with techniques to identify and prevent attacks such as SQL injection, XSS, and DDoS.
3. To develop understanding of security threats, malware, and essential protection mechanisms like authentication, access control, firewalls, and intrusion detection systems.
4. To explain the working and importance of intrusion detection systems(IDS),including different types, tools, and honey pots for threat detection.
5. To familiarize students with cyber laws, digital forensics, and legal aspects related to cyber-crimes, including the Indian IT Act.

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSPE 2.2.1	Understand basic cyber security concepts, threats, and tools.	3	2	-	-	3	2	-	2	-	-	-	3	2	L1, L2
R24CSPE 2.2.2	Analyze web security issues and network attack techniques.	3	3	2	2	3	-	-	-	-	-	-	3	3	L2, L3
R24CSPE 2.2.3	Identify security threats and apply suitable counter measures.	3	3	2	2	3	2	-	2	-	-	-	3	3	L3, L4
R24CSPE 2.2.4	Explain intrusion detection systems and related tools.	2	3	2	3	3	-	-	-	-	-	-	2	2	L4, L5
R24CSPE 2.2.5	Understand cyber laws and digital forensics fundamentals.	2	2	-	2	2	3	-	3	-	2	-	2	2	L5, L6

SYLLABUS**UNIT I:****15Hours**

Introduction to Cyber Security: What is Cyber Security, its need, cyber-threats, Cyber Warfare-Cyber Crime-Cyber Terrorism-Cyber Espionage), Career Growth, Statistics, Inferences, Need for a Comprehensive Cyber Security Policy, Classification of Cyber Crimes, kinds of cyber-crimes, ReasonsforCyberCrimes,CyberSecurityTools:Nmap,Metasploit,Wireshark,Tcpdump,Snort.

CO's-CO1

Self-Learning Topics: Cyber security awareness, social engineering, cyber talking

UNIT II:**15Hours**

Web security: Same origin Policy, Cross Origin Resource Sharing, DDOS, SQL Injection, XSS, Homograph, Generating and storing session tokens.

Networking Scanning & Security Measures: Packet Sniffing and spoofing, Network scanning types, port scanning & its tools, and Network Architecture **CO's–CO2**

Self-Learning Topics: Cross-Site Request Forgery (XSRF/CSRF), spear phishing.

UNIT III: **15Hours**

Security Threats and Vulnerabilities: Malware, Types of malwares, Overview of Security Threats and Vulnerability: Types of attacks on Confidentiality, Integrity and Availability, Security Counter-Measures: Passwords, Access Control (Authenticating users, Handling User Access), Antivirus Software, Firewalls, Intrusion Detection Systems & IPS. **CO's–CO3**

Self-Learning Topics: Types of firewalls, advanced persistent threat, Buffer Overflow

UNIT IV: **15Hours**

Intrusion Detection System: Intrusion Detection System Intruders, Intrusion Detection, Analysis Approaches, Network-Based IDS, Host-Based IDS, signature-based IDS, anomaly-based IDS, advantages and disadvantages of NIDS and HIDS Intrusion Detection Tools, snort architecture, snort rules, case studies of intrusion detection systems, Intrusion detection exchange format. Honey pots, different types of honey pots, benefits and dangers of honey pots. **COs–CO4**

Self-Learning Topics: firewall vs IDS, Physical IDS, honey net

UNIT V: **13Hours**

Cyber Laws and Digital Forensics:

Digital Forensics: Introduction to Digital Forensics, historical background of digital forensics, Forensic Software, and Hardware, need for computer forensics science, special tools and techniques digital forensic life cycle, challenges in digital forensic.

Law Perspective: Introduction to the Legal Perspectives of Cybercrimes and Cyber security, Cybercrime and the Legal Landscape around the World, Why Do We Need Cyber laws, The Indian IT Act, Cybercrime Scenario in India, Digital Signatures and the Indian IT Act **CO's-CO5**

Self-Learning Topics: Cybercrime and Punishment

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. William Stallings, Lawrie Brown, Computer Security Principles and Practice Third Edition, 2015
2. Nihad A. Hassan, Rami Hijazi, Digital Privacy and Security Using Windows: A Practical Guide, Apress, 2017
3. Sunit Belapure and Nina Godbole, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India Pvt. Ltd, 2011.

Reference Books:

1. Michael T. Goodrich and Roberto Tamassia, Introduction to Computer Security, Addison Wesley, 2011.
2. Ross Anderson, Security Engineering: A Guide to Building Dependable Distributed, 3rd Edition, John Wiley & Sons Inc, 2023.
3. Kozierok, Charles M. The TCP/IP guide: a comprehensive, illustrated Internet protocols reference, 2005
4. Harris, Shon, "CISS Pall-in-one exam guide." Sixth edition (2013)

WebReferences:

1. <https://www.nist.gov/cyberframework>
2. <https://www.cisa.gov>
3. <https://owasp.org>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSPE2.3**DEVOPS****3 0 0 3**

(Professional Elective-II)

Course Objectives:

The objectives of this course for the student are to:

1. Learn why automation, culture and metrics are essential to a successful DevOps project.
2. Learn how DevOps can positively impact your business's bottom line. Learn which major Companies are successfully utilizing
3. DevOps in their own engineering processes.
4. DevOps improves collaboration and productivity by automating infrastructure and
5. Workflow and continuously measuring applications performance.

Course Outcomes

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

Mapping with POs and PSOs															Dok
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSP E2.3.1	CO1: Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration and IT service agility	2	2	1	1	1	1	-	1	2	1	-	2	1	L1, L2
R24CSP E2.3.2	CO2: Describe DevOps & DevSecOps methodologies and their key concepts	2	2	1	1	1	2	1	2	2	1	-	2	1	L2, L3
R24CSP E2.3.3	CO3: Illustrate the types of version control systems, continuous integration tools.	2	2	2	1	3	1	-	1	2	1	-	3	2	L3, L4
R24CSP E2.3.4	CO4: Continuous monitoring tools, and cloud models and set up complete private infrastructure using version control systems.	2	2	2	2	3	1	1	1	2	1	-	3	3	L4, L5
R24CSP E2.3.5	CO5:CI/CD tools to Know about DevOps maturity model.	2	3	3	2	2	2	1	2	3	2	1	3	3	L5, L6

SYLLABUS**UNITI:****14 Hours**

Phases of Software Development life cycle. Values and principles of agile software development. Software Development Life Cycle (SDLC) models (Waterfall, Agile, Spiral, DevOps lifecycle), Agile principles (Scrum, Kanban), Limitations of traditional models

CO's-CO1**Self-Learning Topics:** Need for DevOps culture, Foundation for continuous development

UNITII:**15Hours**

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system, Infrastructure as Code (IaC), DevOps ecosystem tools (Git, Jenkins, Docker, Kubernetes)

CO's–CO2**Self-Learning Topics:** Understanding DevOps workflow**UNITIII:****15 Hours**

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes, Automation strategies, CI tools (Jenkins, GitHub Actions)

Self-Learning Topics: Real-world DevOps implementation**UNITIV:****13 Hours**

CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment, Benefits of CI/CD, Metrics to track CI/CDpractices, Monitoring tools(Prometheus,Nagios,Grafana),Metrics: deployment frequency and lead time.

COs–CO4**Self-Learning Topics:** Automation & performance measurement**UNITV:****12 Hours**

Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment. DevSecOps basics (security integration), Case studies of DevOps adoption

COs–CO5**Self-Learning Topics:** Automation & performance measurement

Board of Studies : Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk(To be Delivered by SMEs from Industries)**CO's/PO's/PSO's**

1. From Traditional SDLC to Agile and DevOps Transformation-Limitations of Waterfall model in modern systems.
2. DevOps Architecture and Tool chain in Modern Software Development-
3. ImplementingDevOpsinOrganizations:Challenges&BestPractices-BuildingScalable CI/CD Pipelines and Monitoring Systems
4. DevOps Maturity Models and DevOps Integration

Text Books:

1. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in TechnologyOrganizationsisconsideredtheDevOpsbible.ItiswrittenbyGeneKim,Jez Humble, Patrick Debois, and John Willis
2. Continuous Delivery: Reliable Software Releases Through Build, Test and Deployment Automation, is by Jez Humble and David Farley

Reference Books:

1. Effective DevOps: Building A Culture of Collaboration, Affinity and Tooling at Scale by Jennifer Davis & Ryn Daniels.

Web References:

1. <https://www.udacity.com/course/intro-to-devops--ud611-Goodonlinecoursewithsampleexercises>.
2. [http://www.edureka.co/devops-OnlineTrainingcoveringhighlevelprocessandtools.\(NeedsRegistration\)](http://www.edureka.co/devops-OnlineTrainingcoveringhighlevelprocessandtools.(NeedsRegistration))
3. https://www.edx.org/course?search_query=devops-Hasno.ofcoursesfromMSand

[Redhat.](#)

- [https://www.codementor.io/devops/tutorial -BasicTutorialonDevOps.](https://www.codementor.io/devops/tutorial-BasicTutorialonDevOps)

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSPE2.4 AUGMENTED REALITY AND VIRTUAL REALITY 3 0 0 3
(Professional Elective-II)

Course Objectives:

1. To understand the fundamentals and concepts of Augmented Reality (AR) and Virtual Reality (VR).
2. To analyze AR/VR hardware, devices and interaction techniques.
3. To understand 3D visualization, rendering and tracking technologies.
4. To illustrate AR/VR development tools and application development.
5. To understand the applications, challenges and future trends in AR and VR technologies.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 2.4.1	Describe operating system overview, functions, and services	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2
R24CSPE 2.4.2	Analyze process concept, inter-process communication, and synchronization	3	3	-	2	3	-	-	2	-	-	2	3	2	L2, L3
R24CSPE 2.4.3	Analyze process synchronization mechanisms and dead lock handling techniques.	3	3	2	2	3	-	-	-	2	-	2	3	3	L3, L4
R24CSPE 2.4.4	Describe methods for mapping secondary storage management.	3	2	2	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSPE 2.4.5	Evaluate protection mechanism and security	3	2	3	2	3	2	2	-	-	2	3	-	-	L5, L6

SYLLABUS

UNIT I

15Hours

Introduction to Augmented Reality and Virtual Reality: Definitions, History and Evolution of AR/VR, Characteristics of AR and VR, Components of AR/VR Systems, Types of VR Systems, Applications of AR/VR. **Human Perception and Interaction:** Human Vision, Depth Perception, Motion Tracking, User Interaction Techniques, Challenges in AR/VR Systems. **CO's-CO1**

Self-Learning Topics: Recent Trends and Innovations in AR and VR Technologies.

UNIT II

15Hours

AR/VR Hardware and Devices: Head Mounted Displays (HMDs), Sensors, Cameras, Tracking Devices, Input and Output Devices, Wearable Technologies. **Interaction Techniques:** Gesture Recognition, Voice Interaction, Haptic Devices, Motion Controllers, Eye Tracking Techniques. **CO's-CO2**

Self-Learning Topics: AR/VR Hardware Platforms and Commercial Devices.

UNIT III

15Hours

3D Modeling and Rendering: 3D Graphics Fundamentals: Coordinate Systems, Transformations, 3D Modeling Concepts, Computer Graphics Basics. Rendering and Tracking: Rendering Techniques, Lighting and Shading, Marker-Based Tracking, Markerless Tracking, Simultaneous Localization and Mapping (SLAM), Object Recognition Techniques. **CO's–CO3**

Self-Learning Topics: Advanced Rendering Techniques and Real-Time Tracking.

UNIT IV

15Hours

AV/VR Development and Applications: Development Platforms and Tools: Unity 3D, Unreal Engine, AR Core, AR Kit, Vuforia SDK. AR/VR Application Development: Mobile AR Applications, VR Simulations, Educational Applications, Gaming Applications, Healthcare and Industrial Applications. **COs–CO4**

Self-Learning Topics: Cross-Platform AR/VR Application Development.

UNIT V

10Hours

Future Trends and Challenges in AR/VR: Challenges in AR/VR: Technical Challenges, User Experience Issues, Privacy and Security Concerns, Ethical Issues. Future Trends: Metaverse Concepts, Mixed Reality, Extended Reality(XR), AI Integration with AR/VR, Future Scope of AR and VR Technologies. **CO's–CO5**

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/PO's/PSO's

1. AR and VR Applications in Gaming and Entertainment Industries. CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Industrial Applications of AR/VR in Manufacturing, Healthcare, and Education. CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
3. Metaverse, Mixed Reality, and Future Scope of Immersive Technologies. CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Augmented Reality: Principles and Practice–Dieter Schmalstieg and Tobias Hollerer.
2. Virtual Reality–Steven M. LaValle.
3. Learning Virtual Reality–Tony Parisi

Reference Books:

1. Understand Virtual Reality–William Sherman and Alan Craig.
2. Practical Augmented Reality– Steve Aukstakalnis.
3. Developing Virtual Reality Applications–Alan B. Craig.
4. Augmented Reality for Developers–Jonathan Linowes.

Web References:

1. <https://nptel.ac.in/>
2. <https://learn.unity.com/>
3. <https://developer.apple.com/augmented-reality/>
4. <https://developers.google.com/ar>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSPE3.1

Software Project Management 3 0 0 3
(Professional Elective-III)

Course Objectives:

1. To understand conventional software management and software economics, and apply cost estimation and improvement strategies to enhance software productivity and project performance.
2. To understand the transition from conventional to modern software management principles and apply iterative development practices across software life cycle phases.
3. To understand model-based software architectures and apply workflow-driven iterative planning techniques for effective cost, schedule, and project management.
4. To understand project organization structures and apply metrics, process control techniques, and COCOMO-based cost estimation for effective software project management.
5. To apply effort estimation, scheduling, quality planning, and risk management techniques for effective and reliable software project execution.

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSPE 3.1.1	Estimate overall cost of a software project.	3	2	1	-	2	-	-	-	-	3	1	-	1	L1, L2
R24CSPE 3.1.2	Explain software development process.	3	2	2	-	2	-	-	-	-	3	1	1	-	L2, L3
R24CSPE3 .1.3	Distinguish workflows of process.	3	2	2	1	3	-	-	1	-	3	1	1	1	L3, L4
R24CSPE 3.1.4	Design project organization structure & analyze quality.	3	2	1	2	3	-	-	1	-	3	1	-	1	L4, L5
R24CSPE 3.1.5	Estimate effort and schedule needed for project.	3	3	2	2	2	1	1	2	1	-	-	1	-	L5, L6

SYLLABUS

UNIT I:

13Hours

Conventional Software Management: The Waterfall Model, Conventional Software Management Performance.

Evolution of Software Economics: Software Economics, Pragmatic Software Cost Estimation.

Improving Software Economics: Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation through Software Economics.

CO's-CO1

Self-Learning Topics: Limitations of Waterfall model, Process improvement models

UNIT II:

14Hours

The Old Way and the New:The Principles of Conventional Software Engineering, The Principles Of Modern Software Management, Transitioning to an Iterative Process.

Life Cycle Phases: Engineering and Production Stages, Inception Phase, Elaboration Phase, Construction Phase, Transition Phase. **CO's–CO2**

Self-Learning Topics: Limitations of conventional Software engineering principles, Differences between Waterfall and Iterative processes

UNIT III: **13Hours**

Model Based Software Architectures: A Management Perspective, a Technical Perspective.

Workflows of the Process: Software Process Workflows, Iteration Workflows.

Iterative Process Planning: Work Breakdown Structures, Planning Guidelines, the Cost and Schedule Estimating Process, the Iteration Planning Process. **CO's–CO3**

Self-Learning Topics: Basics of model-based software architecture (UML), Cost and schedule estimation techniques

UNIT IV: **15Hours**

Project Organization and Responsibilities: Line-Of-Business Organizations, Project Organizations, Evolution of Organizations.

Project Control and Process Instrumentation: The Seven Core Metrics, Management Indicators, Quality Indicators Modern Project Profiles. The COCOMO Cost Estimation Model: COCOMO. **COs–CO4**

Self-Learning Topics: Basics of project control and monitoring, Fundamentals of COCOMO model

UNIT V: **12Hours**

Effort Estimation and Scheduling: Effort Estimation, Scheduling.

Quality Planning: Quality Concepts, Quantitative Quality Management Planning.

RISK MANAGEMENT: Risk Assessment, Risk Control. **CO's-CO5**

Self-Learning Topics: Quality planning techniques, Risk identification methods

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Textbooks:

1. WalkerRoyce, “Software Project Management–A Unified Framework”,1st Edition, Pearson Education,2002.
2. PankajJalote, “Software Project Management in Practice”,1st Edition, Pearson Education, 2005.
3. Software Project Management, Bob Hughes&MikeCotterell,TATAMcGraw-Hill.

Reference Books:

1. BobHughes, “Mike Cotterell,RajibMall, Software Project Management”,5thEdition, McGraw-Hill Higher Education, 2011.
2. JoelHenry, “Software Project Management”,1st Edition,PearsonEducation,2006.
3. Norman E. Fenton, Shari Lawrence Pfleeger, “Software Metrics: A Rigorous and Practical Approach “, 1st Edition, PWS Publishing Company,1997

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies(CSE)

Course Objectives:

1. To understand BigData concepts and platforms.
2. To learn stream processing and real-time analytics.
3. To study Hadoop and distributed data processing.
4. To use BigData frameworks like Hive, Pig, and Spark.
5. To apply predictive analytics and visualization techniques.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 3.2.1	Understand Big Data fundamentals, characteristics, architectures, and challenges.	3	2	-	-	1	-	-	-	-	-	2	3	2	L1, L2
R24CSPE 3.2.2	Analyze and process streaming data using real-time analytics and stream processing techniques.	3	3	2	-	2	-	-	2	-	-	2	3	2	L2, L3
R24CSPE 3.2.3	Apply Hadoop eco system tools like HDFS and Map Reduce for large-scale distributed processing.	3	2	3	-	2	-	-	-	-	-	2	3	3	L3, L4
R24CSPE 3.2.4	Utilize Big Data frameworks such as Pig, Hive, HBase, Spark, and PySpark for data handling.	3	2	2	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSPE 3.2.5	Develop predictive analytics models and visualize data using machine learning and visualization tools.	3	3	3	3	3	2	-	-	2	3	3	3	2	L5, L6

SYLLABUS**UNITI:****15 Hours**

Introduction to Big Data: Introduction to Big Data Platform, Challenges of Conventional Systems, Intelligent Data Analysis, Nature of Data, Analytic Processes and Tools, , Cloud-based Big Data Platforms, Introduction to NoSQL Databases, Data Governance and Privacy.**CO's–CO1 Self-Learning Topics:** Blockchain for Big Data, Green Computing in Data Centers.

UNITII:**13Hours**

Mining Data Streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, , Decaying Window, Real-Time Analytics Platform (RTAP) Applications, Introduction to Apache Kafka, Real-Time Data Pipelines, Case Studies – Real-Time

Sentiment Analysis, Stock Market Predictions.

CO's–CO2

Self-Learning Topics: Edge Analytics, Smart City Data Applications.

UNITIII:

13Hours

Hadoop: History of Hadoop, Hadoop Distributed File System (HDFS), Components of Hadoop, Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Design of HDFS, , Developing a Map Reduce Application, How Map Reduce Works, Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Hadoop Environment, YARN Architecture, Hadoop Security Basics.

CO's–CO3

Self-Learning Topics: Container-based Big Data Deployment, Hadoop in Cloud Environments.

UNITIV:

14Hours

Frameworks: Applications on Big Data Using Pig and Hive, Hive Services, HiveQL, Querying Data in Hive, Fundamentals of HBase and ZooKeeper, Introduction to Apache Spark, Spark SQL Basics, Introduction to PySpark.

COs–CO4

Self-Learning Topics: Graph Databases, Introduction to Apache Flink.

UNITV:

13Hours

Predictive Analytics and Visualizations: Predictive Analytics, Simple Linear Regression, Multiple Linear Regression, Interpretation of Regression Coefficients, Visualizations, Visual Data Analysis Techniques, , Systems and Applications, Introduction to Machine Learning, Classification and Clustering Basics, Interactive Dashboards using Tableau/Power BI.

CO's-CO5

Self-Learning Topics: Auto ML Concepts, Augmented Analytics.

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. Tom White, “Hadoop: The Definitive Guide”, Third Edition, O’reilly Media, Fourth Edition,2015.
2. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, McGrawHill Publishing,2012.
3. AnandRajaramanandJeffreyDavidUllman, “Mining of Massive Datasets”,CUP,2012

Reference Books:

1. Bill Franks, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, John Wiley& sons,2012.
2. PaulZikopoulos, DirkdeRoos, KrishnanParasuraman, Thomas Deutsch, JamesGiles, David Corrigan, “Harnessthe PowerofBig Data:The IBMBig Data Platform”, Tata McGraw Hill Publications,2012.
3. Arshdeep Bahga and Vijay Madiseti, “Big Data Science & Analytics: A Hands On Approach “, VPT, 2016.
4. Bart Baesens, “Analytics in a Big Data World: The Essential Guide to Data Science and its Applications (WILEY Big Data Series)”, John Wiley & Sons,2014.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board ofStudies(CSE)**

Course Objectives:

1. To understand the foundations of distributed systems.
2. To learn issues related to clock Synchronization, the need for global state and remote invocation in distributed systems.
3. To learn distributed mutual exclusion and deadlock detection algorithms.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2	
R24CSPE 3.3.1	Understand foundations and issues of distributed systems.	3	2	-	-	-	-	-	-	-	-	-	2	3	2	L1, L2
R24CSPE 3.3.2	Illustrate synchronization, global state, and remote invocation	3	3	2	-	2	-	-	-	-	-	-	2	3	2	L2, L3
R24CSPE 3.3.3	Develop mutual exclusion and deadlock detection algorithms.	3	3	3	2	2	-	-	-	-	-	-	2	3	2	L3, L4
R24CSPE 3.3.4	Apply P2P systems, distributed shared memory, and security	3	2	3	2	3	-	-	1	-	-	-	2	3	3	L4, L5
R24CSPE 3.3.5	Analyze transactions, consensus, and fault tolerance	3	3	2	3	3	-	-	-	-	-	-	2	3	3	L5, L6

SYLLABUS**UNIT– I****15Hours**

Introduction to Distributed Systems, Examples of Distributed Systems, Resource Sharing, Challenges in Distributed Systems, Relation to Computer System Components, Motivation, Message Passing vs Shared Memory Systems, Primitives for Distributed Communication, Synchronous vs Asynchronous Execution, Distributed Program Model, Model of Distributed Executions, Communication Network Models, Global State, Cuts, Logical Clocks Framework, Scalar Time, Vector Time, Physical Clock Synchronization, NTP

CO's-CO1

Self-Learning Topics: Relation to Parallel Systems, Process Communication Models, Past and Future Cones of Events

UNIT– II**13 Hours**

Message Ordering Paradigms, Asynchronous Execution with Synchronous Communication, Program Order in Distributed Systems, Group Communication, Causal Ordering, Total Ordering, Global State Concepts, Snapshot Recording Basics, FIFO Channel Snapshots, Remote Invocation Introduction, RMI Basics, RPC Concepts, Events and Notifications

CO's-CO2

Self-Learning Topics: Detailed RMI Implementation, Advanced Snapshot Algorithms

UNIT– III

13Hours

Distributed Mutual Exclusion Basics, Lamport’s Algorithm, Ricart-Agrawala Algorithm, Maekawa’s Algorithm, Suzuki-Kasami Algorithm, Deadlock Detection Basics, Deadlock Models, Knapp’s Classification, Single Resource Model, AND Model, OR Model

CO’s-CO3

Self-Learning Topics: Performance Comparison of Mutual Exclusion Algorithms

UNIT– IV

12 Hours

Peer-to-Peer Computing Basics, Overlay Networks, Chord DHT, Content Addressable Networks, Distributed Shared Memory Concepts, Memory Consistency Models, Shared Memory Mutual Exclusion, Security Basics, Cryptographic Algorithms, Digital Signatures

CO’s-CO4

Self-Learning Topics: Tapestry Architecture, Practical Cryptography Applications

UNIT– V

13Hours

Distributed Transactions Basics, Flat Transactions, Nested Transactions, Atomic Commit Protocols, Concurrency Control, Check pointing Basics, Failure Recovery Issues, Checkpoint-Based Recovery, Log-Based Recovery, Coordinated Check pointing, Consensus Problem, Agreement in Failure-Free Systems

CO’s-CO5

Self-Learning Topics: Advanced Consensus Algorithms, Fault Tolerance Enhancements

Board of Studies: Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Text Books:

1. Distributed Systems: Principles and Paradigms–AndrewS.Tanenbaum&MaartenVan Steen
2. Distributed Systems: Concepts and Design–GeorgeCoulouriset al.
3. Distributed Computing: Principles and Applications–M.L. Liu
4. Distributed Systems–K. S. Easwarkumar

Reference Books:

1. Distributed Computing–SunitaMahajan&SeemaShah
2. Parallel and Distributed Systems– ArunKulkarniet al.
3. Distributed Systems–AshishKhannaet al.
4. Building Distributed Systems–RanjitAneesh

Web References:

1. [NPTEL Distributed Systems Course \(IIT Patna\)](#)
2. [NPTEL Cloud Computing & Distributed Systems Course](#)
3. [Distributed Systems Tutorial \(GeeksforGeeks\)](#)
4. [MIT 6.824 Distributed Systems Course](#)

Internal Assessment Pattern:

Cognitive Level	Internal Assessment #1 (%)	Internal Assessment #2 (%)
L1	35	--
L2	40	--
L3	25	25

AIETM|R24|CSE|R24CSPE3.3|Distributed Systems

L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand the fundamentals of data science and its applications.
2. To learn data preprocessing, visualization and statistical analysis.
3. To study probability and statistical methods for data analysis.
4. To explore data mining and machine learning basics.
5. To apply tools and techniques for real-world data analysis.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs														Dok
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
R24CSPE 3.4.1	Understand data science concepts and data types	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2	
R24CSPE 3.4.2	Apply data preprocessing and visualization techniques	3	3	-	2	3	-	-	2	2	-	2	3	2	L2, L3	
R24CSPE 3.4.3	Analyze data using statistical methods	3	3	-	2	3	-	-	-	-	-	2	3	3	L3, L4	
R24CSPE 3.4.4	Apply basic machine learning techniques	3	3	2	3	3	-	-	-	-	-	2	3	2	L4, L5	
R24CSPE 3.4.5	Evaluate data-driven solutions for real-world problems	3	2	3	2	3	2	3	-	-	2	3	-	-	L5, L6	

SYLLABUS**UNIT I:****12Hours**

Introduction to Data Science: Definition, importance, applications, data science lifecycle.

Types of Data: Structured, semi-structured, unstructured data.

Data Collection & Sources: Databases, APIs, web data.

Data Preprocessing: Data cleaning, handling missing values, normalization.

CO's–CO1

Self-Learning Topics: Data science tools (Python, R).

UNIT II:**15Hours**

Data Visualization: Importance, visualization techniques, charts (bar, line, histogram and scatter).

Exploratory Data Analysis (EDA): Summary statistics, correlation, covariance.

Tools: Python libraries (Matplotlib, Seaborn).

CO's–CO2

Self-Learning Topics: Dashboard tools (Tableau, Power BI).

UNIT III:**13Hours**

Probability Theory: Basic probability, conditional probability, Bayes theorem

Statistics: Mean median, mode, variance, standard deviation.

Distributions: Normal distribution, binomial distribution.

Hypothesis Testing: t-test, chi-square test.

CO's–CO3

Self-Learning Topics: Central Limit Theorem

UNIT IV:

15Hours

Machine Learning Basics: Supervised vs Unsupervised learning.

Algorithms: Linear regression, logistic regression, k-mean clustering.

Model Evaluation: Accuracy, precision, recall.

COs–CO4

Self-Learning Topics: Decision trees.

UNIT V:

13Hours

Applications of Data Science: Healthcare, finance, social media.

BigData Concepts: Introduction to Hadoop and Spark.

Ethics in Data Science: Privacy, bias, security.

Future Trends: AI integration, automation.

CO's-CO5

Self-Learning Topics: Data science in industry

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/PO's/PSO's

1. Data Science in Industry (Netflix, Amazon use cases)CO1–CO5, PO1, PO2, PO3, PO11,PSO1, PSO2
2. Real-time analytics and dashboardsCO1–CO5, PO1,PO2,PO3, PO11, PSO1, PSO2
3. Machine Learning in production systemsCO1–CO5,PO1,PO2,PO3,PO11,PSO1,PSO2
4. Ethical AI and data governance CO1–CO5, PO1,PO2, PO3,PO11,PSO1, PSO2

Text Books:

1. JoelGrus–*Data Science from Scratch*
2. CathyO'Neil–*Doing Data Science*

Reference Books:

1. Hastie, Tibshirani–*Elements of Statistical Learning*
2. Python Data Science Handbook–JakeVanderPlas

Web References:

1. NPTEL Data Science Courses
2. Kaggle Learning Platform
3. Coursera Data Science Specialization

Internal Assessment Pattern

CognitiveLevel	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. **To understand** the fundamental concepts of computer networks, including network models (OSI and TCP/IP) and different types of networks (LAN, MAN, WAN).
2. **To analyze** the functions and protocols of various network layers such as Physical, Data Link, Network, Transport, and Application layers.
3. **To develop** the ability to evaluate networking mechanisms like routing, congestion control, error control, and reliable data transmission in modern communication systems.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs														Dok
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
R24CSOE0 6.1	Explain the basic concepts of computer networks, network types, reference models, and transmission media.	3	2	-	-	1	-	-	-	-	-	-	-	2	1	L1, L2
R24CSOE0 6.2	Analyze data link layer protocols, error detection and correction techniques, flow control, and multiple access methods.	3	3	1	-	2	-	-	-	-	-	-	-	2	2	L2, L3
R24CSOE0 6.3	Evaluate routing algorithms, packet switching techniques, and congestion control mechanisms in network layer.	3	3	2	1	2	1	-	-	-	-	-	-	3	2	L3, L4
R24CSOE0 6.4	Understand transport layer protocols (TCP/UDP), connection management, and reliability mechanisms.	3	2	2	-	2	-	-	-	-	-	-	-	3	2	L4, L5
R24CSOE0 6.5	Describe application layer protocols such as HTTP, DNS, and web architecture.	2	1	-	-	2	-	-	-	1	-	1	2	3	L5, L6	

SYLLABUS**UNIT I: Introduction & Physical Layer****15Hours**

Introduction: Network Types: LAN, MAN, WAN. Network Reference models: The OSI Reference Model, The TCP/IP Reference Model. Software Defined Networking – Introduction

Physical Layer: Introduction to Guided Transmission Media-Twisted-pair cable, Coaxial cable and Fiber optic cable. Unguided media: Wireless: Radio waves, microwaves, infrared.

Physical Layer Impairments & Signal Integrity–Basics

CO's-CO1

Self-Learning Topics: Digital Modulation and Multiplexing.

UNIT II: Data Link Layer and Media Access Control

15Hours

Design Issues: Services Provided to the Network Layer

Error Detection and Correction: Error Detection: CRC, Checksum.

Error Correction: Framing -Bit and Byte stuffing; Adaptive Frame Size Mechanisms – Basics

Flow Control: Sliding Window Protocols. Stop and wait ARQ; Go Back N, Selective repeat

Channel allocation methods: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, Machine Learning–Based Access Control (in Random Access Networks) – Introduction

Controlled Access: Token Bus, Token Ring. Bridges and Virtual LANs, Energy-Efficient Ethernet (IEEE802.3az)-Introduction

CO's–CO2

Self-Learning Topics: IEEE802, 802.11 and 802.16. Data link layer switching,

UNITIII: Network Layer and Congestion Control

12Hours

Design Issues: Services provided to the Transport layer, Store and Forward Packet Switching, Implementation of Connectionless Service (Datagram Networks), Implementation of Connection Oriented Service (Virtual Circuits), Comparison of Virtual Circuit and Datagram Networks, Hybrid Networking Models Overlay Networks – Basics

Routing Algorithms: The Optimality principle, shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Multicast & Event Distribution Models – Introduction

Congestion Control algorithms: General principles of congestion control, Congestion prevention polices, choke packets, Load shedding, and Jitter Control, Modern Choke Mechanism – Introduction

CO's–CO3

Self-Learning Topics: Internetworking, Fragmentation, Network layer in the internet

UNITIV: Transport Layer

15 Hours

The Transport Layer: Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, UDP, QUIC Protocol – Basics

The Internet Transport Protocol: TCP- reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, Connection Pooling and Keep-Alive Mechanisms – Introduction

COs–CO4

Self-Learning Topics: OSPF, BGP, Internet multicasting, MobileIP, IPV6.ICMP-ARP-DHCP

UNITV: Application Layer

13Hours

WWW: Architecture- Client (Browser), Server, Uniform Resource Locator, **Statelessness & Session Management Techniques – Introduction (New)**

HTTP: HTTP Transaction, HTTP Operational Model and Client/Server Communication, HTTP Generic Message Format, HTTP Request Message Format, HTTP Response Message Format, Error Handling & Status Code Design – Basics

The Domain Name System: The DNS Name Space, Resource Records, Name Servers, DNSSEC & Security Mechanisms – Introduction

CO's–CO5

Self-Learning Topics: Electronic Mail

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk (To be Delivered by SMEs from Industries)

CO's/ PO's/PSO's

1. **Computer Networks in Cloud Computing** (*Cloud networking in platforms like AWS, Azure, GCP; Virtual networks, load balancing, network security, SDN concepts*) CO3–CO5, PO1, PO2, PO3, PO5
2. **Network Security and Cyber security Practices** (*Firewalls, intrusion detection systems, secure protocols, network attacks and prevention techniques*) CO2, CO3, CO5, PO1, PO2, PO3, PO5, PO11
3. **Emerging Networking Technologies** (*IoT networking, 5G communication, network virtualization, software-defined networking (SDN)*) CO1, CO3, CO5, PO1, PO2, PO3, PO5, PO11

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Text Books:

1. **Dr.SanjaySharma**, *Computer Networks*,2ndEdition,KatsonBooks,NewDelhi,2019
2. **B.A.Forouzan**,*Data Communications and Networking*,5thEdition,McGrawHill Education (India), 2017.
3. *Computer Networks*–AndrewSTanenbaum,FifthEdition.PearsonEducation/PHI
4. *Computer Networks*,5ed,DavidPatterson,Elsevier

Reference Books:

1. *Data Communications and Networks* –AchutSGodbole,AtulKahate
2. *Computer Networks*, MayankDave,CENGAGE
3. **N.N.SakhareandA.V.Dhumane**,*Computer Network*,1stEdition,TechnicalPublications, Pune, 2018.
4. **D. S. Gaikwad and N. D. Lambe**, *Computer Network Technology*, 1st Edition, Technical Publications, Pune, 2016.
5. **AchyutS.Godbole**,*Data Communications and Networks*,2ndEdition,McGrawHill Education (India), 2013.
6. *An Engineering Approach to Computer Networks*-S.Keshav,2ndEdition,PearsonEducation

Web References:

1. **NPTEL(SWAYAM)**:<https://nptel.ac.in>
2. **GeeksforGeeks**:<https://www.geeksforgeeks.org>
3. **CiscoNetworkingAcademy**:<https://www.netacad.com>
4. **TutorialsPoint**:<https://www.tutorialspoint.com>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

AIETM|R24|CSE|R24CSOE2.1|Computer Networks

L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand the fundamental concepts, architecture, and paradigms of mobile computing and GSM.
2. To comprehend the motivation for specialized MAC techniques in wireless communications, addressing issues such as hidden and exposed terminals, near and far terminals and IEEE 802.11.
3. To analyze the IP and Mobile IP network layers, focusing on packet delivery, handover management, location management, registration, tunneling, encapsulation and DHCP.
4. To understand conventional TCP/IP protocols and specialized protocols like Indirect TCP, Snooping TCP, and Mobile TCP for mobile networks and database issues in mobile computing
5. To introduce the concept, applications, and challenges of Mobile Ad hoc Networks (MANETs), and to explore various routing algorithms such as DSR, AODV, and DSDV.

Course Outcomes:

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSOE 2.2.1	Understand fundamentals of mobile communication systems, GSM architecture, services, and evolution from 1G to 5G, including QoS, security, and mobile device limitations.	3	2	1	-	2	1	-	-	-	-	1	2	1	L1, L2
R24CSOE 2.2.2	Analyze wireless MAC protocols and understand multiple access techniques, IEEE 802.11 standards, and energy-efficient MAC mechanisms.	3	3	2	1	2	-	-	-	-	-	1	2	2	L2, L3
R24CSOE 2.2.3	Explain mobile network layer concepts including Mobile IP, routing, tunneling, IPv6, DHCP, and mobility management.	3	3	2	2	2	1	-	-	-	-	1	3	2	L3, L4
R24CSOE 2.2.4	Evaluate transport layer protocols in mobile environments and analyze performance issues, congestion control, and enhancements like Indirect TCP and Mobile TCP.	3	2	2	2	3	-	-	-	-	-	1	3	2	L4, L5

R24CSOE 2.2.5	Understand MANETs, routing protocols, mobile ad hoc challenges, IoT integration, and mobile computing platforms and security mechanisms.	3	3	2	2	2	2	1	1	1	1	2	3	3	L5, L6
------------------	--	---	---	---	---	---	---	---	---	---	---	---	---	---	-----------

SYLLABUS

UNIT–I: Introduction

15Hours

Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture, Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices, GSM–Services, System Architecture, Radio Interfaces, Protocols, Localization and Calling, Handover, Security, New Data Services, Quality of Service (QoS) in Mobile Networks, Evolution of Mobile Communication Systems (1G to 5G).

CO’s-CO1

Self-Learning Topics: 5G and Beyond (6G concepts, future trends in mobile communication)

Case Studies of Mobile Computing Applications (Smart Cities, Mobile Banking, Telemedicine)
Comparison of GSM with LTE and 5G architectures.

UNIT–II: Wireless Medium Access Control (MAC)

13Hours

Motivation for a specialized MAC (Hidden and Exposed Terminals, Near and Far Terminals), SDMA, FDMA, TDMA, CDMA, MAC Protocols for Wireless LANs (IEEE802.11, CSMA/CA, RTS/CTS), Energy-Efficient MAC Protocols.

CO’s-CO2

Self-Learning Topics: Advanced IEEE 802.11 Standards (802.11ac, 802.11ax – Wi-Fi 6), MAC Layer Security Issues and Solutions, Cognitive Radio and Dynamic Spectrum Access.

UNIT–III: Mobile Network Layer

13 Hours

IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP, IPv6 in Mobile Networks, Network Security in Mobile IP.

CO’s-CO3

Self-Learning Topics: Software Defined Networking (SDN) in Mobile Networks, Network Function Virtualization (NFV), Mobility Management in 5G Networks

UNIT–IV: Mobile Transport Layer

14Hours

Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks, Congestion Control in Mobile Networks, Performance Issues of TCP in Wireless Networks.

CO’s-CO4

Self-Learning Topics: QUIC Protocol and HTTP/3, Cross-layer Optimization in Mobile Networks, SCTP (Stream Control Transmission Protocol), Impact of Wireless Errors on Transport Protocols

UNIT–V: Mobile Ad hoc Networks (MANETs)

15Hours

Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, Mobile Agents, Service Discovery, Protocols and Platforms for Mobile Computing: WAP, Bluetooth, XML, J2ME, JavaCard,

Palm OS, Windows CE, Symbian OS, Linux for Mobile Devices, Android, Security in MANETs (Attacks and Defense Mechanisms), Internet of Things (IoT) in Mobile Computing.

CO’s-CO5 Self-Learning Topics:

Vehicle Ad hoc Networks (VANETs), Delay Tolerant Networks (DTNs), Block chain in Mobile Ad hoc Networks, Edge Computing and Fog Computing in Mobile Environments.

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books

1. Mobile Communications, Addison-Wesley, 2nd Edition, 2009.
2. Mobile Computing, Oxford University Press, 1st Edition, 2007.
3. Wireless Communications: Principles and Practice, Pearson Education, 2nd Edition, 2002.
4. Computer Networking: A Top-Down Approach, Pearson, Latest Edition.
5. Wireless and Mobile Networks, Pearson Education, 2nd Edition.

Reference Books:

1. **Jochen Schiller** *Mobile Communications*, Pearson Education, (Comprehensive book covering GSM, wireless systems, and mobile networking concepts)
2. **Raj Kamal** *Mobile Computing*, Oxford University Press, (Covers mobile architecture, protocols and applications in detail)
3. **William Stallings** *Wireless Communications and Networks*, Pearson Education Strong foundation for MAC, routing and wireless protocols)
4. **Asoke K. Talukder & Roopa R. Yavagal**, *Mobile Computing* Tata McGraw Hill (Good for Mobile IP, transport layer, and application-level concepts)
5. **Sandeep Singha et al.**, *The Wireless Application Protocol (WAP)* Addison-Wesley (Useful for mobile application protocols and early mobile platforms)

Web References:

1. <https://www.3gpp.org>
(Official standards for GSM, LTE, and 5G technologies)
2. <https://www.cisco.com>
(Networking concepts, wireless technologies, and mobility solutions)
3. <https://www.geeksforgeeks.org/mobile-computing/>
(Simple tutorials and explanation of mobile computing concepts)
4. https://www.tutorialspoint.com/mobile_computing/
(Structured notes on mobile communication and networking)
5. <https://www.ieee.org>
(Research paper on MANETs, MAC protocols, and wireless networks)
6. <https://nptel.ac.in>
(Free video lectures on Mobile Computing and Wireless Networks)

Internal Assessment Pattern

CognitiveLevel	InternalAssessment #1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To understand conventional software management and software economics, and apply cost estimation and improvement strategies to enhance software productivity and project performance.
2. To understand the transition from conventional to modern software management principles and apply iterative development practices across software life cycle phases.
3. To understand model-based software architectures and apply workflow-driven iterative planning techniques for effective cost, schedule, and project management.
4. To understand project organization structures and apply metrics, process control techniques, and COCOMO-based cost estimation for effective software project management.
5. To apply effort estimation, scheduling, quality planning, and risk management techniques for effective and reliable software project execution.

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSOE 2.3.1	Estimate overall cost of a software project.	3	2	1	-	2	-	-	-	-	3	1	-	1	L1, L2
R24CSOE 2.3.2	Explain software development process.	3	2	2	-	2	-	-	-	-	3	1	1	-	L2, L3
R24CSOE 2.3.3	Distinguish workflows of process.	3	2	2	1	3	-	-	1	-	3	1	1	1	L3, L4
R24CSOE 2.3.4	Design project organization structure & analyze quality.	3	2	1	2	3	-	-	1	-	3	1	-	1	L4, L5
R24CSOE 2.3.5	Estimate effort and schedule needed for project.	3	3	2	2	2	1	1	2	1	-	-	1	-	L5, L6

SYLLABUS

UNIT I:

13Hours

Conventional Software Management: The Waterfall Model, Conventional Software Management Performance.

Evolution of Software Economics: Software Economics, Pragmatic Software Cost Estimation.

Improving Software Economics: Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation through Software Economics.

CO's-CO1

Self-Learning Topics: Limitations of Waterfall model, Process improvement models

UNIT II:

14Hours

The Old Way and the New:The Principles of Conventional Software Engineering, ThePrinciples

Of Modern Software Management, Transitioning to an Iterative Process.

Life Cycle Phases: Engineering and Production Stages, Inception Phase, Elaboration Phase, Construction Phase, Transition Phase. **CO's–CO2**

Self-Learning Topics: Limitations of conventional Software engineering principles, Differences between Waterfall and Iterative processes

UNIT III: 13Hours

Model Based Software Architectures: A Management Perspective, a Technical Perspective.

Workflows of the Process: Software Process Workflows, Iteration Workflows.

Iterative Process Planning: Work Breakdown Structures, Planning Guidelines, the Cost and Schedule Estimating Process, the Iteration Planning Process. **CO's–CO3**

Self-Learning Topics: Basics of model-based software architecture (UML), Cost and schedule estimation techniques

UNIT IV: 15Hours

Project Organization and Responsibilities: Line-Of-Business Organizations, Project Organizations, Evolution of Organizations.

Project Control and Process Instrumentation: The Seven Core Metrics, Management Indicators, Quality Indicators Modern Project Profiles. The COCOMO Cost Estimation Model: COCOMO. **COs–CO4**

Self-Learning Topics: Basics of project control and monitoring, Fundamentals of COCOMO model

UNIT V: 12Hours

Effort Estimation and Scheduling: Effort Estimation, Scheduling.

Quality Planning: Quality Concepts, Quantitative Quality Management Planning.

RISK MANAGEMENT: Risk Assessment, Risk Control. **CO's-CO5**

Self-Learning Topics: Quality planning techniques, Risk identification methods

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Textbooks:

1. WalkerRoyce, “Software Project Management–A Unified Framework”,1st Edition, Pearson Education,2002.
2. PankajJalote, “Software Project Management in Practice”,1st Edition, Pearson Education, 2005.
3. Software Project Management, Bob Hughes&MikeCotterell,TATAMcGraw-Hill.

Reference Books:

1. BobHughes, “Mike Cotterell,RajibMall, Software Project Management”,5thEdition, McGraw-Hill Higher Education, 2011.
2. JoelHenry, “Software Project Management”,1st Edition,PearsonEducation,2006.
3. Norman E. Fenton, Shari Lawrence Pfleeger, “Software Metrics: A Rigorous and Practical Approach “, 1st Edition, PWS Publishing Company,1997

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies(CSE)

Course Objectives:

1. Identify Java language components and how they work together in applications
2. Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
3. Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
4. Understand how to design applications with threads in Java
5. Understand how to use Java APIs for program development

Course Outcomes

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

Mapping with POs and PSOs															Dok
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSOE 09.1	Analyze problems, design solutions using OOP principles, and implement them efficiently in Java	3	2	1	-	2	-	-	-	-	-	1	1	-	L1, L2
R24CSOE 09.2	Design and implement classes to model real-world entities, with a focus on attributes, behaviors, and relationships between objects	3	2	2	-	3	-	-	1	1	-	2	1	-	L2, L3
R24CSOE 09.3	Demonstrate an understanding of inheritance hierarchies and polymorphic behaviors, including method overriding and dynamic method dispatch.	3	3	2	1	3	-	1	1	1	-	2	0	-	L3, L4
R24CSOE 09.4	Apply Competence in handling exceptions and errors to write robust and fault-tolerant code.	3	2	3	2	3	-	-	2	1	1	2	-	1	L4, L5
R24CSOE 09.5	Perform file input/output operations, including reading from and writing to files using Java I/O classes, graphical user interface(GUI) programming using JavaFX.	3	2	2	1	3	-	-	1	1	-	2	-	1	L5, L6

SYLLABUS**UNIT I****15 Hours****Object Oriented Programming:** Basic concepts, Principles, Program Structure in Java:

Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments.

Data Types, Variables, and Operators : Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf () Method.

Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (--) Operators, Ternary Operator, Relational Operators.

Control Statements: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator? Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, for–Each for Loop, Break Statement. **CO's–CO1**

Self-Learning Topics: Programming Style, Static Variables and Methods, Attribute Final.

UNIT II

15 Hours

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Constructor Methods for Class, Overloaded Constructor Methods, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods. **CO's–CO2**

Self-Learning Topics: Accessing Private Members of Class, Nested Classes.

UNIT III

13 Hours

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Two-dimensional Arrays.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Static Methods in Interface, Functional Interfaces. **CO's – CO3**

Self-Learning Topics: Arrays of Varying Lengths, Default Methods in Interfaces.

UNIT IV

15 Hours

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Access Control, Packages in Java SE, Java. lang Package and its Classes, Class Object, Enumeration, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Class Instant (java. time. Instant).

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

JavaI/OandFile:JavaI/OAPI,standardI/Ostreams,types,Bytestreams,Characterstreams, Scanner class, Files in Java(Text Book 2) **CO's – CO4**

Self-Learning Topics: Path and Class Path, class Math.

UNIT V

14 Hours

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching;

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and Image, event handling, laying out nodes in scene graph. (Text Book 3) **CO's–CO5**

Self-Learning Topics: Class String Buffer, Thread Priority-Synchronization.

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Text Books:

1. JAVA one step a head, Anitha Seth, B.L. Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object-Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.
3. JAVA for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

1. The complete Reference Java, 11th edition, Herbert Schildt, TMH
2. Introduction to Javaprogramming, 7th Edition, Y Daniel Liang, Pearson

Online Resources:

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Understand different types of computing (parallel, cloud, etc.).
2. Learn basics of cloud computing and why it is needed.
3. Know cloud architecture and how it works.
4. Study different cloud services (IaaS, PaaS, SaaS).
5. Get idea about cloud providers and new technologies.

Course Outcomes:

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

Mapping with Pos and PSOs															Dok
Course Code	Course Outcomes(COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24CSOE 2.5.1	Understand the concepts of computing paradigms.	3	2	1	-	2	-	-	-	-	-	-	3	1	L1, L2
R24CSOE 2.5.2	Understand cloud computing concepts and models.	3	2	-	-	2	-	-	-	-	-	-	3	2	L2, L3
R24CSOE 2.5.3	Describe cloud architecture and migration.	3	3	2	-	3	-	-	-	-	-	-	3	3	L3, L4
R24CSOE 2.5.4	Understanding cloud service providers.	3	2	-	-	3	-	-	-	-	2	-	3	2	L4, L5
R24CSOE 2.5.5	Know basic use of platforms like Amazon Web Services, Google Cloud Platform, and Microsoft Azure.	3	1	-	-	3	-	-	-	-	-	2	2	3	L5, L6

SYLLABUS

UNIT-I

12Hours

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing. Introduction to hypervisors, Simple tools: VMware Microsoft Hyper-V

CO's-CO1

Self-Learning Topics: Difference between parallel and distributed computing

UNIT-II

12Hours

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing, Five Essential Characteristics, Deployment Model Basic Concepts: Scalability, Fault tolerance, Introduction to cloud security

Self-Learning Topics: Micro services vs Monolithic architecture (basic idea) **CO's- CO2**

UNIT-III

15Hours

Cloud Computing Architecture and Management: Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration, Introduction to SDK.

Self-Learning Topics: Simple use of Cloud SDK tools **CO's-CO3**

UNIT-IV

15Hours

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models, Scaling Basics: What is scaling, Basic capacity planning

Self-Learning Topics: Cloud SDK usage and simple application deployment concepts **CO's-CO4**

UNITV:

12 Hours

Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue, and service, Microsoft, Microsoft Assessment and Planning Toolkit. Cloud Application Platform (AnekaBasics): Platform Overview, Introduction to Aneka.

Self-Learning Topics: Server fewer computing basics (Function as a Service concept)

CO's-CO5

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

TEXT BOOK:

- Essentials of cloud Computing:K.Chandrasekharan,CRCpress,2014

REFERENCE BOOKS:

- Cloud Computing: Principles and Paradigms by RajkumarBuyya,JamesBrobergand Andrzej M. Goscinski, Wiley, 2011.
- Distributed and Cloud Computing, KaiHwang,GeofferyC.Fox,JackJ.Dongarra, Elsevier, 2012.
- Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance,Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp 2011.

Internal Assessment Pattern

CognitiveLevel	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

AIETM|R24|CSE|R24CSOE2.5|Fundamentals of Cloud Computing

L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To provide student with an academic environment aware of various AI Algorithms.
2. To train Students with python programming as to comprehend, analyze design and create AI platforms and solutions for the real-life problems.
3. Learn usage of Libraries for Machine Learning in Python
4. Describe appropriate supervised/ Unsupervised learning algorithms for a given problem

Course Outcomes:

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

Course Code	Course Outcomes	Mapping with POs and PSOs													
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	Do K
R24CSPC 21.1	Implement and analyze classical AI search algorithms and problem-solving techniques	3	3	2	2	2	-	-	-	1	-	2	3	2	L1, L2
R24CSPC 21.2	Apply data preprocessing, visualization, and statistical learning methods using Python libraries.	3	2	2	2	3	-	-	-	1	-	2	2	3	L2, L3
R24CSPC 21.3	Design and implement machine learning models and neural networks for classification, regression, and clustering tasks.	3	3	3	3	3	-	-	-	2	-	3	3	3	L3

Developing the following programs using Python**Week1:**

CO's-CO1

1. a) Write a program to implement Breadth First Search.
- b) Write a program to implement Depth First Search.

Week2:

CO's-CO1

2. Write a program to implement Hill Climbing Algorithm.

Week3:

CO's-CO1

3. Write a program to implement A*Algorithm.

Week4:

CO's-CO1

4. Write a program to implement Tic-Tac-Toe game.

Week5:

CO's:CO1

5. Write a program to implement Water Jug Problem.

Week6:

CO's-CO2

6. a) Write a python program to import and export data using Panda's library functions.
- b) Demonstrate various data pre-processing techniques for a given dataset.

Week7:

CO's-CO2

7. a) Implement Dimensionality reduction using Principal Component Analysis (PCA) method.
b) Write a Python program to demonstrate various Data Visualization Techniques.

Week8: CO's-CO3

8. a) Implement Simple and Multiple Linear Regression Models.
b) Develop Logistic Regression Model for a given dataset.

Week9: CO's-CO3

9. Develop Decision Tree Classification model for a given dataset and use it to classify a new sample.

Week10: CO's-CO3

10. a) Implement Naïve Bayes Classification in Python
b) Build KNN Classification model for a given dataset.

Week11: CO's-CO3

11. Build Artificial Neural Network model with back propagation on a given dataset.

Week12: CO's-CO3

12. Write a python program to implement K-Means clustering

Algorithm. Board of Studies: Computer Science and Engineering

Approved in BOS No:03

Approved in ACM No:03

Reference Books:

1. Stuart Russell, Peter Norvig, "Artificial Intelligence: A Modern Approach", Pearson Education
2. Tom M. Mitchell, "Machine Learning", McGraw-Hill
3. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow", O'Reilly
4. Jake VanderPlas, "Python Data Science Handbook", O'Reilly

Online Learning Resources:

1. <https://nptel.ac.in>
2. <https://www.coursera.org>

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. To understand the basic principles of cryptography and network security.
2. To study and implement classical and modern encryption algorithms.
3. To develop programming skills in C and Java for cryptographic applications.
4. To apply mathematical concepts like modular arithmetic in security algorithms.
5. Understand various cryptographic algorithms.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs													
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	DoK
R24CSPC 24 .1	Apply basic programming and bitwise operations to manipulate strings using C language.	3	2	2	-	3	1	-	1	1	-	1	3	2	L1, L2
R24CSPC 24 .2	Implement and demonstrate classical and symmetric encryption algorithms such as Caesar Cipher, Substitution Cipher Hill Cipher, DES, Blowfish, and Rijndael (AES).	3	2	3	2	3	1	1	1	1	-	2	3	3	L2, L3
R24CSPC 24 .3	Analyze and implement modern cryptographic techniques including public key algorithms (RSA, Diffie-Hellman) and hashing techniques (SHA-1) for secure communication and data integrity.	3	3	2	3	3	2	3	1	1	1	2	3	3	L3

Developing the following programs:

WEEK1

1. Write a C program that contains a string (char pointer)with a value\HelloWorld'. The program should XOR each character in this string with 0 and displays the result.

CO's-CO1

WEEK2

2. Write a C program that contains a string (char pointer) with a value \Hello World'. The program should AND or and XOR each character in this string with 127 and display the result

CO's-CO1

WEEK3

3. Write a Java program to perform encryption and decryption using the following algorithms:

- a) Ceaser Cipher
- b) Substitution Cipher
- c) Hill Cipher. **CO's-CO1**

WEEK4

- 4. Write a Java program to implement the DES algorithm logic **CO's-CO2**

WEEK 5

- 5. Write a C/JAVA program to implement the BlowFish algorithm logic **CO's-CO2**

WEEK 6

- 6. Write a C/JAVA program to implement the Rijndael algorithm logic. **CO's-CO2**

WEEK 7

- 7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool. **CO's-CO2**

WEEK8

- 8. Write a Java program to implement RSA Algorithm` **CO's:CO2**

WEEK 9

- 9. Implement GCD, Modular Inverse, and Euler's Totient Function **CO's-CO3**

WEEK 10

- 10. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob). **CO's-CO3**

WEEK11

- 11. Calculate the message digest of a text using the SHA-1 algorithm in JAVA. **COs'-CO3**

WEEK 12

- 12. Calculate the message digest of a text using the MD5 algorithm in JAVA. **CO's-CO3**

WEEK 13

- 13. Write a Java program to implement Digital Signature using RSA Algorithm for signing and Verification of a message. **CO's-CO3**

WEEK14

- 14. Write a Java program to implement Secure Socket Communication using SSL/TLS. **CO's-CO3**

Board of Studies :Computer Science and Engineering

Approved in BOS No : 03

Approved in ACM No: 03

Text books:

- 1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition.
- 2. Cryptography and Network Security: AtulKahate, McGrawHill,3rd Edition.

Reference books:

- 1. Cryptography and Network Security: CKShyamala,NHarini,DrTRPadmanabhan, Wiley India, 1st Edition.
- 2. Cryptography and Network Security: Forouzan Mukhopadhyay,McGrawHill,3rd Edition.
- 3. Information Security, Principles and Practice: MarkStamp,WileyIndia.
- 4. Principles of Computer Security: WM.ArthurConklin,GregWhite, TMH.
- 5. Introduction to Network Security: NealKrawetz, CENGAGE Learning.
- 6. Network Security and Cryptography: BernardMenezes,CENGAGE Learning

Online Learning Resources:

1. <https://cacr.uwaterloo.ca/hac/>
2. <https://williamstallings.com/Cryptography/>

**Chairperson
Board of Studies(CSE)**

Prerequisites: Any Programming language, Data Structures & Algorithms.

Course Objective: the main objective of this course is to make the student familiar with competitive coding by addressing different algorithm-based problems.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE1.1	Implement various algorithms		3	3	3	2		2	2	L3
R24CSPE1.2	Analyze solutions of different problems using Backtracking and dynamic programming		3	3	3	2		2	2	L4
R24CSPE1.3	Develop codes for implementation of recursion and other techniques		3	3	3	2		2	2	L4

SYLLABUS

List of Experiments:

1. Maximum subarray sum

Solve the Maximum subarray sum with different time complexities.

2. The Median of Two Sorted Arrays :

Program to find the median of two sorted arrays of same size and different size are discussed here. Firstly, let us see what is median of the array? Median is an element which divides the array into two parts - left and right. So the number of elements on the left side of the array will be equal or less than the number of elements on the right side. Now, let us consider the case of an array with odd number of elements. Array = [9,11,16,7,2] Sorted array = [2,7,9,11,16]. In this case, the median of this array is 9, since it divides the array into two parts: [2,7] and [11,16]. Further, let us consider the case of an array with even elements. Array = [1,2,3,4,5,6]. In such a case, we will take the average between the last element of the left part and the first element of the right part. In this case, the median equals = $(3 + 4) / 2 = 3.5$.

Input Format

The input should contain 3 lines.

- I. First line of the input should contain two integer values which specify the number of elements in array1 and array2.
- II. Second line of the input should contain the elements of the first array.
- III. Third line of the input should contain the elements of the second array.

Constraints:

All elements must be Integers

Output Format

The output should print only the median value.

Sample Input 1

5 6 -5 3 6 12 15 -12 -10 -6 -3 4 10

Sample Output 1

3

Sample Input 2

4 6 2 3 5 8 10 12 14 16 18 20

Sample Output 2

11

3. Division with Binary Search

We can modify binary search algorithm to perform division of two numbers, by defining range [0, infinity] which serves as initial low and high for the binary search algorithm. Now we need to find a mid that satisfies $x/y = \text{mid}$ or $x*\text{mid}$ for given two numbers x and y . Based on the comparison result based on x and $y*\text{mid}$, we either update low, update high or return mid. 1. If $y*\text{mid}$ almost equal to x , we return mid. 2. If $y*\text{mid}$ is less than x , we update low to mid 3. If $y*\text{mid}$ is more than x , we update high to mid We need to care about division by zero and sign of the result etc. Input: one line of input should contain two numbers separated by space. Output: should print division of the numbers as a result.

Input Format

Input: one line of input should contain two numbers separated by space.

Input1: 22 7

Constraints

$x, y < \text{infinity}$

Output Format

Output: should print division of the numbers as a result.

Sample Input 1

22 7

Sample Output 1

3.14286

4. Find the Triplet

Given an array of integers, find a triplet having maximum product in the array.

Input: First line of input should specify the number of elements in the array.

Second line of input should specify each element separated by space.

Output: should print Triplets.

Test Cases:**Sample Input 1**

5 -4 1 -8 9 6

Sample Input 1

-4 -8 9

Sample Input 2

5 1 7 2 -2 5

Sample Input 2

7 2 5

5. 3n+1

Consider the following algorithm to generate a sequence of numbers. Start with an integer n . If n is even, divide by 2. If n is odd, multiply by 3 and add 1. Repeat this process with the new value of n , terminating when $n = 1$. For example, the following sequence of numbers will be generated for $n = 22$: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1 It is conjectured (but not yet proven) that this algorithm will terminate at $n = 1$ for every integer n . Still, the conjecture holds for all integers up to at least 1, 000, 000. For an input n , the cycle-length of n is the number of numbers generated up to and including the 1. In the example above, the cycle length of 22 is 16. Given any two numbers i and j , you are to determine the maximum cycle length over all numbers between i and j , including both endpoints. The input will consist of a series of pairs of integers i and j , one pair of integers per line. All integers will be less than 1,000,000 and greater than 0. Output For each pair of input integers i and j , output i, j in the same order in which they appeared in the input and then the maximum cycle length for integers between and including i and j . These three numbers should be separated by one space, with all three numbers on one line and with one line of output for each line of input.

Sample Input	Sample Output
1 10	1 10 20
100 200	100 200 125
201 210	201 210 89
900 1000	900 1000 174

6. Matching –String with wild card –pattern.

Check the given string is matches with pattern containing wild card characters („*“ and „?“), where the „*“ can match to any number of characters including zero characters and „?“ can match to any single character in the given input string. Check if the given input string is matches with given input pattern or not.

Input: Input should contain two lines.

First line of input should contain input string.

Second line of input should contain pattern string.

Output: The output should print either „0“ or „1“.

‘1’ in the output indicates that the given string is matches with the given pattern.

‘0’ in the output indicates that the given string is not matched with the given pattern.

Sample Input 1:

abcabcccd a?c*d

Sample Output1

1

Sample Input 2:

abcabcccd a?c*c

Sample Output2

0

7. Rotten Oranges

Given a grid of dimension nxm where each cell in the grid can have values 0, 1 or 2 which has the following meaning:

0 : Empty cell

1 : Cells have fresh oranges

2 : Cells have rotten oranges

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

Example 1:

Input: grid = {{0,1,2},{0,1,2},{2,1,1}}

Output: 1

Explanation: The grid is-

0 1 2

0 1 2

2 1 1

Oranges at positions (0,2), (1,2), (2,0)

will rot oranges at (0,1), (1,1), (2,2) and

(2,1) in unit time.

8. Minimum-Cost –Path

Find a path in an $n \times n$ grid from the upper-left corner to the lower-right corner such that we only move down and right and diagonally lower cells from a given cell, i.e., from a given cell (i, j) , cells $(i+1, j)$, $(i, j+1)$ and $(i+1, j+1)$ can be traversed. Assume that all costs are positive integers. Each square contains a number, and the path should be constructed so that the sum of numbers along the path is as small as possible.

1	2	3
4	8	2
1	5	3

1	2	3
4	8	2
1	5	3

The path is $(0, 0) \rightarrow (0, 1) \rightarrow (1, 2) \rightarrow (2, 2)$. The cost of the path is 8 $(1 + 2 + 2 + 3)$

Sample Input 1

1 2 3
4 8 2
1 5 3

Sample Output 1

8

9. 15-Puzzle Problem

The 15-puzzle is a very popular game: you have certainly seen it even if you don't know it by that name. It is constructed with 15 sliding tiles, each with a different number from 1 to 15, with all tiles packed into a 4 by 4 frame with one tile missing. The object of the puzzle is to arrange the e tiles so that they are ordered as below:

The only legal operation is to exchange the missing tile with one of the 2, 3, or 4 tiles it shares an edge with. Consider the following sequence of moves:

We denote moves by the neighbor of the missing tile is swapped with it. Legal values are "R," "L," "U," and "D" for right, left, up, and down, based on the movements of the hole. Given an initial configuration of a 15-puzzle you must determine a sequence of steps that take you to the final state. Each solvable 15-puzzle input requires at most 45 steps to be solved with our judge solution; you are limited to using at most 50 steps to solve the puzzle.



Input The first line of the input contains an integer n indicating the number of puzzle set inputs. The next $4n$ lines contain n puzzles at four lines per puzzle. Zero denotes the missing tile.

Output For each input set you must produce one line of output. If the given initial configuration is not solvable, print the line “This puzzle is not solvable.” If the puzzle is solvable, then print the move sequence as described above to solve the puzzle.

Sample Input 1

```
2
2 3 4 0
1 5 7 8
9 6 10 12
13 14 11 15
13 1 2 4
5 0 3 7
9 6 10 12
15 8 11 14
```

Sample Output 1

```
LLLDRDRDR
This puzzle is not solvable
```

10. Tug of War

Tug of war is a contest of brute strength, where two teams of people pull in opposite directions on a rope. The team that succeeds in pulling the rope in their direction is declared the winner. A tug of war is being arranged for the office picnic. The picnickers must be fairly divided into two teams. Every person must be on one team or the other, the number of people on the two teams must not differ by more than one, and the total weight of the people on each team should be as nearly equal as possible.

Input The input begins with a single positive integer on a line by itself indicating the number of test cases following, each described below and followed by a blank line. The first line of each case contains n, the number of people at the picnic. Each of the next n lines gives the weight of a person at the picnic, where each weight is an integer between 1 and 450. There are at most 100 people at the picnic. Finally, there is a blank line between each two consecutive inputs.

Output For each test case, your output will consist of a single line containing two numbers: the total weight of the people on one team, and the total weight of the people on the other team. If these numbers differ, give the smaller number first. The output of each two consecutive cases will be separated by a blank line.

Sample Input 1

```
1
3
100
90
200
```

Sample Output 1

```
190 200
```

11. Find first set bit

Problem Statement : Given an integer an N. The task is to return the position of first set bit found from the right side in the binary representation of the number. Note: If there is no set bit in the integer N, then return 0 from the function.

Test case 1

Input: N = 18

Output: 2

Test case 2

Input: N = 12

Output: 3

Expected Time Complexity: O(log N).

Constraints: 0 <= N <= 10⁸

12. Euclid Problem

From Euclid, it is known that for any positive integers A and B there exist such integers X and Y that $AX + BY = D$, where D is the greatest common divisor of A and B. The problem is to find the corresponding X, Y, and D for a given A and B.

Input The input will consist of a set of lines with the integer numbers A and B, separated with space ($A, B < 1,000,000,001$).

Output For each input line the output line should consist of three integers X, Y, and D, separated with space. If there are several such X and Y, you should output that pair for which $X \leq Y$ and $|X| + |Y|$ is minimal.

Sample Input 1

4 6
17 17

Sample Output 1

-1 1 2
0 1 17

13. Coin Problem

Given a value V. You have to make change for V cents, given that you have infinite supply of each of C { C1, C2, .., Cm } valued coins. Find the minimum number of coins to make the change and print the coins that appear in an optimal solution.

Input:

The first line of input contains an integer T denoting the number of test cases. The first line of each test case is V and N, V is the value of cents and N is the number of coins. The second line of each test case contains N input C[i], value of available coins.

Output:

Print the coins appear in an optimal solution and in a newline print the minimum number of coins to make the change and, if not possible print "-1".

Constraints:

$1 \leq T \leq 100$
 $1 \leq V \leq 10^6$
 $1 \leq N \leq 10^6$
 $1 \leq C[i] \leq 10^6$

Sample Input 1

1
7 2
2 1

Sample Input 1

2 2 2 1
4

Explanation :

Testcase 1: We can use coin with value 2 three times, and coin with value 1 one times to change a total of 7.

14. spirally traversing a matrix

Given a matrix of size r*c. Traverse the matrix in spiral form.

Test case 1:

Input:

r = 4, c = 4
 matrix[][] = {{1, 2, 3, 4},
 {5, 6, 7, 8},
 {9, 10, 11, 12},
 {13, 14, 15, 16}}

Output:

1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10

Test case 2:

Input:

r = 3, c = 4
 matrix[][] = {{1, 2, 3, 4},

{5, 6, 7, 8},
{9, 10, 11, 12}}

Output:

1 2 3 4 8 12 11 10 9 5 6 7

Text Books:

1. Steven Halim, Felix Halim, “Competitive programming 3”, Handbook for ACMICPC and IOI contestants.

Reference Books:

1. Steven S. Skiena Miguel A. Revilla “programming challenges”, The Programming Contest Training Manual, Springer.
2. Antti Laaksonen . “Competitive Programmer’s Handbook”.
3. Ahmed Shamsul Arefin, “Art of Programming Contest” , special online edition for UVA online judge users.

Online Resources:

1. <https://jadi.net/wp-content/uploads/2017/07/competitive-programmers-handbook.pdf>
2. http://acm.cs.buap.mx/downloads/Programming_Challenges.pdf
3. https://www.comp.nus.edu.sg/~stevenha/myteaching/competitive_programming/cp1.pdf
4. https://www.comp.nus.edu.sg/~stevenha/database/Art_of_Programming_Contest_SE_for_uva.pdf

Problem reading & Practice:

<https://uva.onlinejudge.org/index.php>

**Chairperson
Board of Studies (CSE)**

Course Objective

The course is designed to provide core engineering students with essential programming, AI, machine learning, data engineering, cloud, and software deployment fundamentals required for campus placements and modern technology roles. It focuses on developing coding ability, analytical thinking, AI integration skills, data handling, deployment knowledge, and project development capabilities through practical and industry-oriented learning.

Course Outcomes (COs)

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

1. Apply programming and data structure concepts to solve coding and technical interview problems.
2. Use database and data engineering concepts for data processing and application development.
3. Integrate AI/ML models and LLM APIs into software applications using prompt engineering techniques.
4. Demonstrate foundational knowledge of cloud platforms, MLOps practices, Docker, and deployment workflows.
5. Develop collaborative AI-enabled projects with proper documentation, teamwork, and technical

UNIT-I: ADVANCED PROGRAMMING & DSA FOUNDATIONS

This unit develops programming and analytical problem-solving skills through important programming concepts and data structures required for campus placements. Students learn recursion, time complexity analysis, arrays, linked lists, stacks, queues, hashing basics, trees overview, sorting, searching algorithms, and coding optimization techniques used in technical interviews.

Hands-on Skills

- Solving coding problems
- Implementing data structures
- Complexity analysis
- Competitive coding practice

UNIT-II: DATA ENGINEERING & DATABASE FUNDAMENTALS

This unit introduces data engineering and database concepts required for software and AI applications. Topics include SQL basics, SQL/NoSQL systems, data preprocessing, ETL/ELT processes, data transformation workflows, streaming vs batch processing, and pipeline automation concepts.

Hands-on Skills

- Writing SQL queries
- Data cleaning and transformation

- Building basic data pipelines
- Pipeline workflow simulation

UNIT-III: AI FOUNDATIONS & LLM APPLICATIONS

This unit introduces foundational AI and machine learning concepts along with LLM applications. Topics include datasets, NumPy and Pandas basics, machine learning pipeline overview, LLM fundamentals, prompt engineering, embeddings, vector database basics, AI APIs, and AI application design.

Hands-on Skills

- Working with datasets
- Using AI APIs
- Prompt engineering practice
- Building simple AI-powered features

UNIT-IV: CLOUD, MLOPS & MODEL DEPLOYMENT

This unit covers cloud computing and deployment concepts for AI and software systems. Topics include AWS/Azure/GCP basics, Docker basics, CI/CD fundamentals, model deployment strategies, monitoring systems, model versioning, and MLOps lifecycle concepts.

Hands-on Skills

- Deploying ML models
- Creating Docker-based applications
- Setting up CI/CD workflows
- Monitoring deployed applications

UNIT-V: PROJECT DEVELOPMENT & INTERVIEW PREPARATION

This unit focuses on collaborative project development and placement readiness. Topics include teamwork, documentation practices, Git/GitHub workflows, resume preparation, technical interview preparation, coding interview basics, and project presentation techniques.

Hands-on Skills

- Team collaboration and presentation
- Git workflow management
- Resume and interview practice
- AI-enabled mini project development

Text Books (Indian Authors)

1. Programming in ANSI C – E. Balagurusamy, McGraw Hill Education India.
2. Data Structures Using C – Reema Thareja, Oxford University Press India.
3. Database Management Systems – P. Radha Krishna, Himalaya Publishing House.

Reference Books (Indian Authors)

1. Artificial Intelligence – S. Kaushik, Cengage India.
2. Cloud Computing – Wiley India.
3. Machine Learning Using Python – Wiley India.

Web Resources

1. NPTEL
2. SWAYAM
3. AICTE Internship Portal

R24MC05|

Value Education**3 0 0 3** (Common to CSE)**Course Objectives:**

A value system is a set of consistent values and measures. Knowledge of the values is inculcated through education. It contributes in forming true human being, who is able to face life and make it meaningful. People should be treated with respect and dignity. Values tend to influence attitudes and behavior and help to solve common human problems. Students are trained to get rid of all sort of social evils such as injustice, dishonesty, misbehaviors, ragging and violation of human rights etc.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24MC05.1	to understand the concepts of value education	3	2	-	-	1	2	1	1	L1,L2, L4
R24MC05.2	to understand the advantages of the virtues of doing good deeds and disadvantages of bad deeds.	2	3	3	-	2	3	3	2	L2,L3
R24MC05.3	to develop the sense of accountability towards Almighty Creator and understand the disadvantages of neglecting the commandments of Almighty Creator.	2	2	3	2	3	3	3	2	L3,L4
R24MC05.4	to learn philosophy of life, personal values and individual qualities.	1	3	2	2	2	2	3	2	L3,L4
R24MC05.5	to learn and practice social values and responsibilities.	1	2	1	1	2	1	2	3	L4,L5

SYLLABUS

Unit-I: Balancing the outer and inner - Body, Mind and Intellectual level- Duties and responsibilities- Virtues of Islam (Page No.15 to 20)

Unit-II: Punctuality-Time, task and resource management - Problem solving and decision-making skills, Interpersonal and Intra personal relationship Team work-Positive and creative thinking Salath or Prayer (Page No.21 to 45)

Unit-III Dr. APJ Kalam's ten points for enlightened citizenship - Social Values and Welfare of the citizen - The role of media in value building. Zakath (Page No.46 to 51) – Invocations, Supplications mentioned in Hadith (Page No.52 to 60)

Unit-IV The binding of man and nature - Environment conservation and enrichment - About the prophets (Page No. 61 to 65) - Seetathun Nabi-Part II (Page No.66 to 72)

Unit-V Dowry - Domestic violence - untouchability-female infanticide-atrocities against women - How to tackle them - Seetathun Nabi-Part II (Page No.73 to 87) Seetathun Nabi-Part II (Page No.73 to 87)

Text Books:

1. M.G. Chitakra: Education and Human Values, A.P.H. Publishing Corporation, New Delhi, 2003
2. Deeniyath, Book-I, Third Edition-June 2004, Dr.Syed Rahmathullah, Former Head, P.G. & Research Dept. of Arabic, The New College, Chennai-14.

**Chairperson
Board of Studies (CSE)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B.Tech Computer Science and Engineering

Regulation: R24

IV Year I Semester -Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PC	R24CSPC23	Neural Networks & Deep Learning	3	0	0	3
2	PC	R24CSPC24	Object Oriented Analysis and Design	3	0	0	3
3	HS	R24HS14	Human Resources & Project Management	2	0	0	2
4	PE-4	R24CSPE4.1 R24CSPE4.2 R24CSPE4.3 R24CSPE4.4	1. Software Architecture & Design Patterns 2. Block chain Technology 3. Quantum Computing 4. AGENTIC AI SYSTEMS 5.12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	OE-3		OPEN ELECTIVE-3	3	0	0	3
6	SC	R24CSSC05	Prompt Engineering/ SWAYAM Plus - Certificate program in Prompt Engineering and ChatGPT	0	0	2	2
7	IN	R24INO2	Evaluation of Industry Internship (Done during III-II Summer vacation- 8 weeks)	2	0	0	2
8	MC	R24MC06	Technical paper writing and IPR	2	0	0	0
TOTAL				18	00	06	18

Category	Courses	Credits
PC-Professional Core Courses	2	6
PE- Professional Elective	1	3
OE-Open Elective	1	3
SC-Skill Enhancement course	1	2
MC-Mandatory Course	1	0
HS-Humanities and Social Science including Management Course	1	2
IN- Evaluation of Industry Internship	1	2
Total	9	18

Course Objectives:

1. To understand fundamentals of artificial neural networks, learning algorithms, and perception models.
2. To study deep learning architectures, optimization techniques, and regularization methods.
3. To analyze and implement convolution and recurrent neural network models for real- world applications.
4. To understand deep generative models such as RBMs, DBNs, and Boltzmann Machines.
5. To develop problem-solving skills for designing intelligent systems using neural network and deep learning techniques.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPC25 .1	Understand and apply neural network fundamentals, perceptions, learning rules and back propagation algorithms.	3	2	2	1	2	-	-	-	-	-	-	3	2	L1, L2
R24CSPC25 .2	Analyze deep learning concepts, optimization methods and regularization techniques for training deep neural networks.	3	3	2	2	3	-	-	-	-	-	1	3	2	L2, L3
R24CSPC25 .3	Design and implement convolution neural networks for image-related applications.	3	3	3	2	3	-	-	-	1	-	-	3	3	L3, L4
R24CSPC25 .4	Apply recurrent neural networks and sequence models for temporal and sequential data processing.	3	3	3	2	3	-	-	-	1	-	-	3	3	L4, L5
R24CSPC25 .5	Understand and apply deep generative models for representation learning and probabilistic modeling.	3	2	2	2	3	-	-	-	-	-	1	3	2	L5, L6

SYLLABUS

UNITI:

14Hours

Introduction to Neural Networks: Introduction to Neuron Model, Neural Network Architecture,

Learning Rules, Perceptions, Single Layer Perceptions, Multilayer Perceptions, Back propagation Networks, Over fitting and Under fitting, Applications of Neural Networks. **CO's –CO1**

Self-Learning Topics: Kohonen's self-organizing networks, Hopfield network

UNIT II:

13 Hours

Introduction to Deep Learning: History of Deep Learning, Applications of Deep learning, Greedy layer wise training, Optimization for training DNNs, Newer optimization methods for neural networks (AdaGrad, RMSProp, Adam), Second order methods for training, Regularization methods (dropout, drop connect, batch normalization) **CO's –CO2**

Self-Learning Topics: Hessian-based optimization, Data augmentation as regularization

UNIT III:

13Hours

Convolutional Neural Networks: Introduction to CNN, From Fully-Connected Layers to Convolutions, Convolutions for Images, Multiple Input and Multiple Output Channels, Pooling, LeNet, AlexNet, Networks Using Blocks (VGG), Networks with Parallel Concatenations (GoogLeNet), Residual Networks (ResNet), Densely Connected Networks (DenseNet) **CO's–CO3**

Self-Learning Topics: Batch Normalization, Network in Network (NiN)

UNIT IV:

14Hours

Recurrent Neural Networks: Introduction to RNN, Implementation of Recurrent Neural Networks from Scratch, Back propagation Through Time, Gated Recurrent Units (GRU), Long Short-Term Memory (LSTM), Bidirectional Recurrent Neural Networks. **CO's–CO4**

Self-Learning Topics: Concise Implementation of Recurrent Neural Networks, Sequence to Sequence

UNIT V:

15 Hours

Deep Generative Models: Boltzmann Machines, Restricted Boltzmann Machines (RBMs), Deep Belief Networks (DBNs), Deep Boltzmann Machines (DBMs), Boltzmann Machines for Real-Valued Data, Convolutional Boltzmann Machines, Boltzmann Machines for Structured or Sequential Outputs, Other Boltzmann Machines. **CO's–CO5**

Self-Learning Topics: Back-Propagation through Random Operations, Directed Generative Net

Board of Studies : Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Text Book:

1. Goodfellow, I., Bengio, Y., and Courville, A., Deep Learning, MIT Press, 2016.
2. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, Dive into Deep Learning, 2021
3. Learning Deep Learning by Magnus Ekmen, Edison Wisley publishers; 2022
4. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006.

Reference Books:

1. Chollet, Francois, Deep learning with Python, New York: Manning Publications; 2018
2. Richard Wilson, Mastering Deep Learning Fundamentals with Python, 2019
3. Cloud Computing – web-based Applications that change the way you work and collaborate Online – Michael Miller. Pearson Education.
4. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
5. Matrix Computations, Golub, G., H., and Van Loan, C., F, JHU Press, 2013.
6. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education,

2004

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies(CSE)

Course Objectives:

1. Become familiar with all phases of OOAD.
2. Master the main features of the UML.
3. Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains .

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE1.1.1	Select the basic elements of modeling such as Things, Relationships and Diagrams depending on the views of UML Architecture and SDLC.	1	2	3	-	2	1	2	-	L1,L2, L4
R24CSPE1.1.2	Apply basic and Advanced Structural Modeling Concepts for designing real time applications	2	1	2	-	2	2	2	-	L2,L3
R24CSPE1.1.3	Design Class and Object Diagrams that represent Static Aspects of a Software System.	2	1	3	1	2	3	1	-	L3,L4
R24CSPE1.1.4	Analyze Dynamic Aspects of a Software System using Use Case, Interaction and Activity Diagrams	1	3	2	1	1	2	2	-	L3,L4
R24CSPE1.1.5	Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems.	2	2	1	2	3	2	1	-	L4,L5

SYLLABUS

UNIT I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation **COs–CO1**

UNIT II:

Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural**

Modeling: Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management. **COs–CO2**

UNIT III:

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis. **COs–CO3**

UNIT IV:

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System **COs–CO4**

UNIT V:

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams. **Case Study:** Weather Forecasting **COs–CO5**

Expert Talk (To be delivered by SMEs from Industries)

	COs	POs / PSOs
1. Case Study: Weather Forecasting	CO5	PO1,PO2,PO4,PSO1,PSO2,PSO3

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , “Object- Oriented Analysis and Design with Applications”, 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. AtulKahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

Web References:

**Chairperson
Board of Studies (CSE)**

R24CSPE4.1|

**Software Architecture & Design Patterns
(PROFESSIONAL ELECTIVE-IV)**

3 0 0 3 (Common to CSE)

Course Objectives:

1. Define software architecture and design patterns.
2. Understand the conceptual classes and relationships using object-oriented modeling techniques.
3. Implement patterns such as Adapter, Bridge, Composite, Decorator, Facade, Flyweight, and Proxy.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE4.1.1	Understand the basic concepts to identify state behavior of real world objects.	3	3	-	-	-	-	-	-	L1,L2, L4
R24CSPE4.1.2	Apply Object Oriented Analysis and Design concepts to solve complex problems.	3	3	3	3	-	2	-	-	L2,L3
R24CSPE4.1.3	Construct various UML models using the appropriate notation for specific problem context.	3	3	3	3	3	2	-	-	L3,L4
R24CSPE4.1.4	Design models to Show the importance of systems analysis and design in solving complex problems using case studies.	3	3	3	3	3	2	-	-	L3,L4
R24CSPE4.1.5	Study of Pattern Oriented approach for real world problems.	3	3	3	3	3	2	2	-	L4,L5

SYLLABUS

UNIT - I Introduction: What is a design pattern? Describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern, What is object oriented development? key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm. **COs–CO1**

UNIT – II Analysis a System: Overview of the analysis phase, stage 1 gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain Design and Implementation, discussions and further reading. **COs–CO2**

UNIT – III

Design Pattern Catalog: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy. **COs–CO3**

UNIT – IV Interactive systems and the MVC architecture: Introduction The MVC architectural pattern, analyzing a simple drawing program designing the system, designing of the subsystems, getting into implementation, implementing undo operation drawing incomplete items, adding a new feature pattern based solutions. **COs–CO4**

UNIT – V Designing with Distributed Objects: Client server system, java remote method invocation, implementing an object oriented system on the web, Web services (SOAP, Restful), Enterprise Service Bus. **COs–CO5**

Text Books:

1. Object oriented analysis, design and implementation, brahma dathan, sarnathrammath , universities press,2013
2. Design patterns, Erich Gamma, Richard helan , Ralph johman , john vlissides, PEARSON Publication,2013

Reference Books:

1. Frank Bachmann, RegineMeunier, Hans Rohnert “Pattern Oriented Software Architecture” Volume 1, 1996.
2. William J Brown et al., "Anti Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998

Web References:

1. [Coursera Software Architecture Courses](#)
 - o Courses on software architecture, system design, and design patterns.
2. [edX Software Engineering Courses](#)
 - o University-level courses covering architecture and software design.
3. Udemy Design Patterns Courses
 - o Practical implementation of design patterns in various programming languages.
4. [NPTEL Software Architecture Courses](#)
 - o Free lectures from IITs on software engineering and architecture.

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. Understand the fundamentals of block chain technology, consensus mechanisms.
2. Analyze the concepts of decentralization, smart contracts and platforms supporting block chain-based ecosystems.
3. Gain knowledge of Bit coin architecture, including digital keys, transactions.
4. Understand the Ethereum platform, including smart contracts.
5. Explore Hyper ledger frameworks and analyze challenges and opportunities in block chain.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 4.2.1	Define and explain the fundamentals of Block chain.	3	2	1	-	2	-	-	-	-	-	1	1	-	L1, L2
R24CSPE 4.2.2	Understand decentralization and the role of Block chain in it.	3	3	2	1	3	1	-	1	1	-	2	-	1	L2, L3
R24CSPE 4.2.3	Understand and analyze Bitcoin Crypto currency and underlying Block chain network.	3	2	2	1	3	-	-	-	1	-	2	-	-	L3, L4
R24CSPE 4.2.4	Understand Ethereum currency and platform, and develop applications using Solidity.	3	2	3	1	3	1	-	1	1	1	2	1	-	L4, L5
R24CSPE 4.2.5	Understand Hyper ledger project and its components; critically analyze the challenges and future opportunities in Block chain technology.	3	3	2	2	3	2	1	1	1	-	3	-	1	L5, L6

SYLLABUS**UNIT I:****15 Hours**

Introduction: History and basics, Types of Blockchain, Consensus, CAP Theorem.

Cryptographic Hash Functions: Properties of hash functions, Secure Hash Algorithm, Merkle trees, Patricia trees.

CO's-CO1

Self-Learning Topics: Distributed Systems Basics, Digital Signatures and Public Key Cryptography

UNIT II:**13Hours**

Decentralization: Decentralization using Blockchain, Methods of decentralization,

decentralization framework, Blockchain and full ecosystem decentralization, Smart contracts, Decentralized Organizations, Platforms for decentralization. **CO's–CO2**

Self-Learning Topics: Tokenization and Digital Assets, Interoperability in Block chain

UNITIII: 13Hours

Bitcoin: Introduction to Bitcoin, Digital keys and addresses, Transactions, Block chain, The Bitcoin network, Bitcoin payments, Bitcoin Clients and APIs, Alternatives to Proof of Work, Bitcoin limitations. **CO's–CO3**

Self-Learning Topics: Structure of a Bitcoin Block, Security and Attacks in Bitcoin

UNIT IV: 14Hours

Ethereum: Smart Contracts, Introduction to Ethereum, The Ethereum network, Components of the Ethereum ecosystem, Blocks and Blockchain, Fee schedule, Ethereum Development Environment, Solidity. **COs–CO4**

Self-Learning Topics: Account Types in Ethereum, Smart Contract Security Basics

UNIT V: 14Hours

Hyperledger: Introduction, Hyperledger Projects, Protocol, Architecture, Hyperledger Fabric, Saw tooth Lake, Corda.

Challenges and Opportunities: Scalability, Privacy, Blockchain for IoT, Emerging trends

CO's-CO5

Self-Learning Topics: Consensus Mechanisms in Hyper ledger, Scalability Solutions in Blockchain

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Book:

1. Mastering Blockchain, ImranBashir,SecondEdition,PacktPublishing.

Reference Books:

1. Mastering Bitcoin: Unlocking Digital Cryptocurrencies,AndreasAntonopoulos,O'Reilly.
2. Blockchain Blue print for a New Economy, MelanieSwan,O'Reilly.
3. Mastering Bitcoin: Programming the Open Blockchain,Antonopoulos,AndreasM.O'Reilly.
4. Blockchain Technology: Cryptocurrency and Applications,S.Shukla,M.Dhawan,S. Sharma, S. Venkatesan, Oxford University Press.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To introduce the fundamental principles of quantum computing, including qubits, superposition, entanglement, and quantum measurement, and to build a strong foundation in quantum mechanics concepts relevant to computation.
2. To understand the design and operation of quantum gates and circuits, and to enable students to represent and manipulate quantum information using various quantum gate operations.
3. To study key quantum algorithms such as Deutsch, Deutsch–Jozsa, and Grover’s algorithm, and to analyze their computational advantages over classical algorithms.
4. To understand the challenges of quantum noise and decoherence, and to explore quantum error correction techniques along with applications in quantum cryptography and secure communication.
5. To explore different physical models of quantum computing, analyze implementation challenges, and evaluate real-world applications and future directions of quantum technologies.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE4.3.1	Understand basic concepts of quantum computing and quantum mechanics	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2
R24CSPE4.3.2	Apply quantum gates and circuits for computation	3	3	-	2	3	-	-	2	2	-	2	3	2	L2, L3
R24CSPE4.3.3	Analyze quantum algorithms and their efficiency	3	3	-	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSPE4.3.4	Understand quantum error correction and cryptography	3	-	-	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSPE4.3.5	Evaluate quantum computing models and future trends	3	2	3	2	3	2	3	-	-	2	3	-	-	L5, L6

SYLLABUS**UNIT I:****15Hours**

Quantum Computing Fundamentals: Introduction to Quantum Computing, Classical vs Quantum Computing, Qubits, Bloch Sphere Representation, Quantum States and Superposition, Quantum Measurement, Density Matrices and Quantum Entanglement.

CO’s–CO1

Self-Learning Topics: Applications of Quantum Computing.

UNIT II: 14Hours
Quantum Gates and Circuits: Quantum Gates: Pauli Gates (X, Y, Z), Hadamard Gate, Phase Shift and Rotation Gates, Controlled Gates (CNOT, Toffoli), Quantum Circuits Representation, Quantum Circuit Design Principles and Quantum Parallelism. **CO's-CO2**

Self-Learning Topics: IBM Qiskit basics.

UNITIII: 13Hours
Quantum Algorithms: Deutsch Algorithm, Deutsch–Jozsa Algorithm, Grover’s SearchAlgorithm, Shor’s Algorithm, Quantum Fourier Transform and Complexity of Quantum Algorithms. **CO's-CO3**

Self-Learning Topics: Comparison with classical algorithms

UNIT IV: 15Hours
Quantum Error Correction & Security: Quantum Noise and Decoherence, Quantum Error Correction Codes, Fault Tolerance, Quantum Cryptography, Quantum Key Distribution (QKD) and Security Challenges in Quantum Systems. **COs-CO4**

Self-Learning Topics: Post-quantum cryptography.

UNITV: 12Hours
Quantum Computing Models & Applications: Quantum Computing Models: NMR Model, Spintronics Model, Linear Optical Model. Physical Realization of Quantum Computers, Scalability Issues, Industrial Applications and Future of Quantum Computing. **CO's-CO5**

Self-Learning Topics: Quantum supremacy

Board of Studies: Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries) CO's/PO's/PSO's

1. Quantum Computing in Cloud Platforms (IBMQ, Azure Quantum)CO1–CO5,PO1,PO2, PO3, PO11, PSO1, PSO2
2. Quantum Machine LearningCO1–CO5,PO1,PO2,PO3,PO11,PSO1,PSO2
3. Quantum Cryptography in CybersecurityCO1–CO5,PO1,PO2,PO3,PO11,PSO1,PSO2
4. Real-time Applications of Quantum ComputingCO1–CO5,PO1,PO2,PO3,PO11, PSO1, PSO2

Text Books:

1. EricR. Johnston, NicHarrigan–*Programming Quantum Computers*
2. ChristineCorbettMoran–*Mastering Quantum Computing with IBMQX*

Reference Books:

1. MichaelNielsen&Isaac Chuang–*Quantum Computation and Quantum Information*
2. PhillipKaye–*Introduction to Quantum Computing*

WebReferences:

1. NPTEL Quantum Computing Courses
2. IBM Quantum Experience
3. Qiskit Documentation

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Demonstrate an understanding of generative AI and agentic systems by explaining core concepts, identifying model types and applications, and analyzing the architecture, autonomy, and behavior of intelligent and multi-agent systems.
2. Explain and apply the essential components and reflective capabilities of intelligent agents to design adaptive, generative AI-based systems.
3. Design intelligent agents by integrating knowledge representation, reasoning, learning, and generative AI to build adaptive and reflective agentic systems.
4. Design and develop trustworthy agentic AI systems by applying environment modeling, memory architecture, processing workflows, and ethical principles such as transparency and explainability.
5. Demonstrate an understanding of safety, ethical considerations, and real-world applications in the design and deployment of agentic AI systems.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE4.4.1	Explain the fundamental concepts, model types, and applications of generative AI and agentic systems.	3	3	-	-	-	-	-	-	L2
R24CSPE4.4.2	Apply the principles of knowledge representation, reasoning, and learning to design intelligent and adaptive agents.	3	3	3	3	-	2	-	-	L3
R24CSPE4.4.3	Develop generative AI-based agentic systems with reflective capabilities for real-world problem-solving.	3	3	3	3	3	2	-	-	L5
R24CSPE4.4.4	Implement environment modeling, memory management, and workflow techniques to build safe, transparent, and trustworthy agentic AI systems.	3	3	3	3	3	2	-	-	L5
R24CSPE4.4.5	Analyze ethical, safety, and societal implications of agentic AI and apply responsible practices in various application domains.	3	3	3	3	3	2	2	-	L4

SYLLABUS

UNIT-I: Fundamentals of Generative AI: Introduction to generative AI, Types of generative AI models, Applications of generative AI, Challenges and Limitations of generative AI. Principles of Agentic Systems: Technical requirements, Understanding self-governance, agency and autonomy, Exploring the architecture of agentic systems, Reviewing intelligent agents and their characteristics.

COs–CO1

UNIT-II: Essential Components of Intelligent Agents: Technical requirements, Knowledge representation in intelligent agents, Reasoning in intelligent agents, Learning mechanisms for adaptive agents, Decision-making and planning in agentic systems, Enhancing agent capabilities with generative AI. Designing and Implementing Generative AI-Based Agents: Technical requirements, The importance of reflection in agents, Use cases and examples, Introspection in intelligent agents.

COs–CO2

UNIT-III: Enabling Tool Use and Planning in Agents: Technical requirements, Understanding the concept of tool use in agents, Planning algorithms for agents, Integrating tool use and planning. Exploring the Coordinator, Worker, and Delegator Approach: Technical requirements, Understanding the CWD model, Designing agents with role assignments, Communication and collaboration between agents, Implementing the CWD approach in generative AI systems. **COs–CO3**

UNIT-IV: Effective Agentic System Design Techniques: Technical requirements, Focused system prompts and instructions for agents, State spaces and environment modeling, Agent memory architecture and context management, Sequential and parallel processing in agentic workflows. Trust, Safety, Ethics, and Applications: Technical requirements, Importance of trust in AI, Techniques for establishing trust, Implementing transparency and explainability. **COs–CO4**

UNIT -V: Managing Safety and Ethical Considerations: Understanding potential risks and challenges, Ensuring safe and responsible AI, Exploring ethical guidelines and frameworks. Common Use Cases and Applications: Creative and artistic applications, Natural language processing and conversational agents, Decision support and optimization. **COs–CO5**

Text Books:

1. Building Agentic AI Systems by Anjanava Biswas & Wrick Talukdar, Packt Publishing, 2025

References Books:

1. AI Agents in Action by Michael Lanham, Manning Publications Co. (Shelter Island, NY), 2025

Web links and Video Lectures (e-Resources):

1. https://www.youtube.com/watch?v=BkedAnQfJ_U&pp=ygUqRXNzZW50aWFsIENvbXBvbmVudHMgb2YgSW50ZWxsaWdlbnQgQWdlbnRz
2. <https://www.youtube.com/watch?v=nNL7UfOVcsY>
3. <https://www.youtube.com/watch?v=rV3HJ4LEZ7k&t=28s>
4. <https://www.youtube.com/watch?v=9bGYJ68qvAA>
5. <https://www.youtube.com/watch?v=eTPiL3DF27U&list=PL2hXNYim5xvop-LH93T2P75Y-09GIyHG6>

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. To understand the fundamentals of Python programming, including syntax, data types, and operators.
2. To apply control structures such as conditional statements and loops for problem-solving.
3. To demonstrate the use of strings, lists, tuples and dictionaries for data manipulation.
4. To develop modular programs using functions and basic object-oriented programming concepts.
5. To understand file handling, input/output operations and basic debugging techniques.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with POs and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE3.1 .1	Apply Python syntax, data types, and operators to develop basic programs.	3	1	-	1	2	-	-	-	-	-	1	2	2	L1, L2
R24CSOE3.1 .2	Design solutions using conditional statements and looping constructs.	3	2	1	2	3	-	-	1	-	-	1	2	2	L2, L3
R24CSOE3.1 .3	Manipulate and manage data using strings, lists, tuples and dictionaries.	2	3	-	2	3	-	-	-	-	-	1	2	3	L3, L4
R24CSOE3.1 .4	Develop modular and reusable code using functions and basic object-oriented concepts.	3	2	1	3	3	-	-	-	-	-	1	3	2	L4, L5
R24CSOE3.1 .5	Perform file handling operations and implement debugging techniques for error-free programs.	3	2	2	3	3	1	1	-	-	1	2	2	2	L5, L6

SYLLABUS**UNIT I:****15 Hours**

Introduction to Python Programming: Features/Characteristic of Python - Basic Syntax - Basic Data Types - Simple Input-Output - Precedence of Operators - Type Conversion - Conditional Statements- Looping: for, while, Nested Loops; Terminating Loops- Skipping Specific Conditions - Scope and Extent of Variables - Testing and Debugging Principles.

CO's-CO1

Self-Learning Topics: Command line execution of Python programs

UNIT II:**15 Hours**

Strings & Lists: Concept - Escape Characters - String Special Operations - String Formatting Operator –Single Quotes – Double Quotes -Triple Quotes –Raw String- Unicode Strings - Built-in String Methods - List Type Built-in Methods - Special Features of Lists. **CO’s–CO2**

Self-Learning Topics: Advanced stringslicing and indexing techniques.

UNIT III: 14 Hours

Tuples, Dictionaries & Functions: Tuples- Tuple Operators and Built-in Functions- Special Features of Tuples- Dictionaries, Operators, Built-in Functions, Built-in Methods, Dictionary Keys - Functions - Calling Functions - Creating Functions - Formal Arguments - Positional Arguments – Default Arguments-Variable-Length Arguments. **CO’s–CO3**

Self-Learning Topics: Nested dictionaries and real-world applications

UNIT IV: 10 Hours

Object Oriented Programming: Introduction to Classes, Objects and Methods; Standard Libraries **COs–CO4**

Self-Learning Topics: Encapsulation and abstraction concepts.

UNIT V: 14 Hours

Files and Input/Output: File Objects-File Built-in Function -File Built-in Methods-File Built-in Attributes - Standard Files - Command-Line Arguments - File System - File Execution – Persistent Storage Modules. **CO’s-CO5**

Self-Learning Topics: Exception handling in files operations

Board of Studies : Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Text Books:

1. Hetland,MagnusLie, “Beginning Python from Novice to Professional,” Apress,2nd Edition, 2017.
2. Lutz,M., “Learning Python: Powerful Object-Oriented Programming, ”O’ReillyMedia, Inc. 5th Edition, 2013.

Reference Books:

1. FabeoNelli,“Python Data Analytics,”Apress,2ndEdition, 2018.
2. PaulBarry, “Head first Python: A Brain-Friendly Guide, ”O’ReillyMedia, Inc.,2nd,2016.

WebReferences:

1. <https://www.python.org/doc/>
2. <https://docs.python.org/3/tutorial/>
3. <https://www.w3schools.com/python/>
4. <https://www.geeksforgeeks.org/python/>
5. <https://www.programiz.com/python-programming/>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Understand the structure and nature of complex systems.
2. Learn basic structural concepts like classes, relationships and diagrams.
3. To develop the ability to model system structure using class and object diagrams, including advanced Structural concepts.
4. Learn system behavior using use case, interaction, and activity diagrams.
5. To explore advanced behavioral and architectural modeling techniques, including state machines, Component and Deployment diagrams.

Course Outcomes:

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

Course Code	Course Outcomes (COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSO E3.2.1	Understand methods to design and manage complexity in system architecture.	3	2	3	1	1	0	0	0	1	1	0	2	1	L2 L3
R24CSO E3.2.2	Understand UML concepts and the importance of modeling in software development.	3	2	2	1	2	0	0	0	1	1	0	2	2	L2 L3
R24CSO E3.2.3	Understand and apply class and object diagram concepts for system modeling.	3	3	2	2	2	0	0	0	1	1	0	3	2	L3 L4
R24CSO E3.2.4	Understand behavioral modeling using use case, activity, and interaction diagrams.	3	3	3	2	2	0	0	0	2	2	1	3	2	L3 L4
R24CSO E3.2.5	Understand advanced behavioral concepts like state machines and processes.	3	3	3	2	2	1	0	0	2	2	1	3	3	L4 L5

SYLLABUS

UNITI:

15 Hours

The Structure of Complex Systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems.

CO's-CO1

Self-Learning Topics: Banking Transaction Systems

UNITII: **13 Hours**

Introduction to UML: Importance of modeling, principles of modeling, object-oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. **CO's–CO2**

Self-Learning Topics: Software Design Documentation, Traffic management system modeling.

UNITIII: **15 Hours**

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **CO's–CO3**

Self-Learning Topics: Library Management Systems, CryptAnalysis.

UNITIV: **15 Hours**

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **COs–CO4**

Self-Learning Topics: Online booking systems, Vacation Tracking System.

UNITV: **13 Hours**

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams. **CO's–CO5**

Self-Learning Topics: Web Applications (Weather Forecasting)

Board of Studies :Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No: 03

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, “Object- Oriented Analysis and Design with Applications”, 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

Reference Books:

1. MeilirPage-Jones: Fundamentals of Object-Oriented Design in UML, PearsonEducation.
2. PascalRoques: Modeling Software Systems Using UML2, WILEY-DreamtechIndiaPvt.Ltd.
3. AtulKahate: Object Oriented Analysis & Design, The McGraw-HillCompanies.
4. Applying UML and Patterns: An introduction to Object–Oriented Analysis and Designand Unified Process, Craig Larman, Pearson Education.

Web References:

1. <https://nptel.ac.in/courses/106105153>
2. https://www.tutorialspoint.com/object_oriented_analysis_design/index.htm
3. <https://www.geeksforgeeks.org/software-engineering/object-oriented-analysis-and-design/>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board ofStudies(CSE)**

Course Objectives:

1. To understand Big Data concepts and platforms.
2. To learn stream processing and real-time analytics.
3. To study Hadoop and distributed data processing.
4. To use Big Data frame works like Hive,Pig, and Spark.
5. To apply predictive analytics and visualization techniques.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 3.3.1	Understand Big Data fundamentals, characteristics, architectures, and challenges.	3	2	-	-	1	-	-	-	-	-	2	3	2	L1, L2
R24CSOE 3.3.2	Analyze and process streaming data using real-time analytics and stream processing techniques.	3	3	2	-	2	-	-	2	-	-	2	3	2	L2, L3
R24CSOE 3.3.3	Apply Hadoop ecosystem tools like HDFS and Map Reduce for large-scale distributed processing.	3	2	3	-	2	-	-	-	-	-	2	3	3	L3, L4
R24CSOE 3.3.4	Utilize Big Data frameworks such as Pig, Hive, HBase, Spark, and PySpark for data handling.	3	2	2	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSOE 3.3.5	Develop predictive analytics models and visualize data using machine learning and visualization tools.	3	3	3	3	3	2	-	-	2	3	3	3	2	L5, L6

SYLLABUS**UNITI:****15 Hours**

Introduction to Big Data: Introduction to Big Data Platform, Challenges of Conventional Systems, Intelligent Data Analysis, Nature of Data, Analytic Processes and Tools, , Cloud-based Big Data Platforms, Introduction to NoSQL Databases, Data Governance and Privacy.**CO's–CO1 Self-Learning Topics:** Blockchain for Big Data, Green Computing in Data Centers.

UNITII:**13Hours**

Mining Data Streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, , Decaying Window, Real-Time Analytics Platform (RTAP) Applications, Introduction to Apache Kafka, Real-Time Data Pipelines, Case Studies – Real-Time

Sentiment Analysis, Stock Market Predictions.

CO's–CO2

Self-Learning Topics: Edge Analytics, Smart City Data Applications.

UNITIII:

13Hours

Hadoop: History of Hadoop, Hadoop Distributed File System (HDFS), Components of Hadoop, Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Design of HDFS, , Developing a Map Reduce Application, How Map Reduce Works, Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Hadoop Environment, YARN Architecture, Hadoop Security Basics.

CO's–CO3

Self-Learning Topics: Container-based Big Data Deployment, Hadoop in Cloud Environments.

UNITIV:

14Hours

Frameworks: Applications on Big Data Using Pig and Hive, Hive Services, HiveQL, Querying Data in Hive, Fundamentals of HBase and ZooKeeper, Introduction to Apache Spark, Spark SQL Basics, Introduction to PySpark.

COs–CO4

Self-Learning Topics: Graph Databases, Introduction to Apache Flink.

UNITV:

13Hours

Predictive Analytics and Visualizations: Predictive Analytics, Simple Linear Regression, Multiple Linear Regression, Interpretation of Regression Coefficients, Visualizations, Visual Data Analysis Techniques, , Systems and Applications, Introduction to Machine Learning, Classification and Clustering Basics, Interactive Dashboards using Tableau/Power BI.

CO's–CO5

Self-Learning Topics: Auto ML Concepts, Augmented Analytics.

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. Tom White, “Hadoop: The Definitive Guide”, Third Edition, O'reilly Media, Fourth Edition,2015.
2. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, McGrawHill Publishing,2012.
3. AnandRajaramanandJeffreyDavidUllman, “Mining of Massive Datasets”,CUP,2012

Reference Books:

1. Bill Franks, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, John Wiley& sons,2012.
2. PaulZikopoulos, DirkdeRoos, KrishnanParasuraman, Thomas Deutsch, JamesGiles, David Corrigan, “Harness the Power of Big Data: The IBM Big Data Platform”, Tata McGraw Hill Publications,2012.
3. Arshdeep Bahga and Vijay Madisetti, “Big Data Science & Analytics: A Hands On Approach “, VPT, 2016.
4. Bart Baesens, “Analytics in a Big Data World: The Essential Guide to Data Science and its Applications (WILEY Big Data Series)”, John Wiley & Sons,2014.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--

AIETM|R24|CSE|R24CSOE3.3|Big Data Analytics

L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSOE3.4**WEB SERVICES****3 0 0 3**

(Open Elective-III)

Course Objectives:

1. To understand different Internet Technologies
2. To learn java-specific web services architecture
3. To Develop web applications using frameworks

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 3.4.1	Construct a basic web site using HTML and Cascading Style Sheets	3	2	3	-	2	2	-	1	2	-	1	3	2	L1, L2
R24CSOE 3.4.2	Build dynamic web page with validation using Java Script objects and by applying different event handling mechanisms	3	3	3	1	3	1	-	2	2	-	2	3	3	L2, L3
R24CSOE 3.4.3	Develop server-side programs using Servlets and JSP.	3	3	3	2	3	2	-	2	2	2	2	3	3	L3, L4
R24CSOE 3.4.4	Construct simple web pages in PHP and to represent data in XML format.	3	2	3	1	3	2	-	2	2	1	2	3	3	L4, L5
R24CSOE 3.4.5	Develop interactive web applications.	3	3	3	2	3	2	1	3	3	2	3	3	3	L5, L6

SYLLABUS**UNITI: Website Basics, HTML5,CSS3, Web 2.0****13Hours**

Web Essentials: Clients, Servers and Communication, The Internet, World Wide Web, HTTP Request Message, HTTP Response Message, Web Clients, Web Servers, HTML5, Forms, Images, Tables, Lists, Audio, Video controls, CSS3, External style sheets, CSS selectors, Colors, Text styling, Layout basics, Responsive design, Bootstrap

CO's-CO1

Self-learning topics: Responsive web design, Flexbox and Grid layouts, HTTP vs HTTPS basics, UI design principles

UNITII: Client-Side Programming**15Hours**

JavaScript, Introduction to JavaScript, DOM manipulation, Form validation, Events handling, JSON, API calls using fetch, Local Storage and Session Storage in web applications, JavaScript error handling and debugging techniques

CO's-CO2

Self-learning topics: ES6 features, Asynchronous JavaScript (Promises, async/await), Working with REST APIs, Browser developer tools

UNITIII: Server-Side Programming**12Hours**

Servlets, Java Servlet basics, Servlet life cycle, Request handling (GET, POST), Session management, Cookies, JDBC database connectivity using JDBC. **CO's–CO3**

Self-learning topics: REST API basics, MVC architecture, Authentication basics, SQL Injection prevention

UNITIV: PHP and XML **13Hours**

Introduction to PHP, Variables, Forms handling, CRUD operations with MySQL, XML basics, XML vs JSON **CO's–CO4**

Self-learning topics: PHP with MySQL projects, File handling, RESTAPI integration, XPath basics

UNITV: Web Frame works & Tools **12Hours**

Introduction to React, Basics of Node.js, Introduction to Angular JS, Basics of Firebase, UI/UX fundamentals, Web application structure, Component lifecycle in React, State management basics

Self-learning topics: React vs Angular comparison, Node.js backend basics, Firebase authentication, Modern frontend development **CO's–CO5**

Board of Studies : Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries) **CO's/PO's/PSO's**

1. Modern web development uses HTML5 for semantic structure and multimedia support, CSS3 for responsive design and advanced styling, and Bootstrap for mobile-first UI development. HTTP request-response model forms the base of client-server communication in web applications. Web 2.0 enables interactive and user-driven websites like social platforms. CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. JavaScript is used for building interactive web applications through DOM manipulation and event handling. Form validation ensures data correctness on the client side. JSON is widely used for data exchange between front end and backend systems.CO1–CO5,PO1,PO2,PO3, PO11, PSO1, PSO2
3. Java Servlets handle server-side processing of client requests using GET and POST methods. Session management and cookies are used to maintain user state in web applications. JDBC provides database connectivity for enterprise-level applications. CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Web Technologies–UttamK.Roy(OxfordUniversityPressIndia)
2. Internet & World Wide Web: Howto Program– Deitel&Deitel(PearsonIndia)
3. Java: The Complete Reference –HerbertSchildt(McGrawHillIndia)
4. Web Technologies(TheoryandPractice)–S.PadmaPriya(DreamtechPress India)
5. Beginning PHP, MySQL, Java Script and HTML5 for Web Development–Wrox (Wiley India)

Reference Books:

1. Web Technologies–UttamK.Roy(OxfordUniversityPressIndia),
2. Internet & World Wide Web: How to Program–Deitel&Deitel(PearsonIndia),
3. Java: The Complete Reference–HerbertSchildt(McGrawHillEducationIndia),

Web References:

1. <https://developer.mozilla.org>(MDNWebDocs–HTML5,CSS3, JavaScript, DOM,APIs)

2. <https://www.w3schools.com>(HTML, CSS,JavaScript,PHP,XMLbeginnertutorials)
3. <https://www.javatpoint.com>(Servlets,JDBC,PHP,WebTechnologiessimplified notes)
4. <https://docs.oracle.com>(JavaServlet,JDBC,officialJavabackend documentation)

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies(CSE)

Course Objectives:

1. To introduce the fundamental principles of quantum computing, including qubits, superposition, entanglement, and quantum measurement, and to build a strong foundation in quantum mechanics concepts relevant to computation.
2. To understand the design and operation of quantum gates and circuits, and to enable students to represent and manipulate quantum information using various quantum gate operations.
3. To study key quantum algorithms such as Deutsch, Deutsch–Jozsa, and Grover’s algorithm, and to analyze their computational advantages over classical algorithms.
4. To understand the challenges of quantum noise and decoherence, and to explore quantum error correction techniques along with applications in quantum cryptography and secure communication.
5. To explore different physical models of quantum computing, analyze implementation challenges, and evaluate real-world applications and future directions of quantum technologies.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE3.5.1	Understand basic concepts of quantum computing and quantum mechanics	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2
R24CSOE3.5.2	Apply quantum gates and circuits for computation	3	3	-	2	3	-	-	2	2	-	2	3	2	L2, L3
R24CSOE3.5.3	Analyze quantum algorithms and their efficiency	3	3	-	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSOE3.5.4	Understand quantum error correction and cryptography	3	-	-	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSOE3.5.5	Evaluate quantum computing models and future trends	3	2	3	2	3	2	3	-	-	2	3	-	-	L5, L6

SYLLABUS**UNIT I:****15Hours**

Quantum Computing Fundamentals: Introduction to Quantum Computing, Classical vs Quantum Computing, Qubits, Bloch Sphere Representation, Quantum States and Super position, Quantum Measurement, Density Matrices and Quantum Entanglement.

CO’s–CO1

Self-Learning Topics: Applications of Quantum Computing.

UNIT II: **14Hours**
Quantum Gates and Circuits: Quantum Gates: Pauli Gates (X, Y, Z), Hadamard Gate, Phase Shift and Rotation Gates, Controlled Gates (CNOT, Toffoli), Quantum Circuits Representation, Quantum Circuit Design Principles and Quantum Parallelism. **CO's-CO2**

Self-Learning Topics: IBM Qiskit basics.

UNIT III: **13Hours**
Quantum Algorithms: Deutsch Algorithm, Deutsch–Jozsa Algorithm, Grover's Search Algorithm, Shor's Algorithm, Quantum Fourier Transform and Complexity of Quantum Algorithms. **CO's-CO3**

Self-Learning Topics: Comparison with classical algorithms

UNIT IV: **15Hours**
Quantum Error Correction & Security: Quantum Noise and Decoherence, Quantum Error Correction Codes, Fault Tolerance, Quantum Cryptography, Quantum Key Distribution (QKD) and Security Challenges in Quantum Systems. **COs-CO4**

Self-Learning Topics: Post-quantum cryptography.

UNIT V: **12Hours**
Quantum Computing Models & Applications: Quantum Computing Models: NMR Model, Spintronics Model, Linear Optical Model. Physical Realization of Quantum Computers, Scalability Issues, Industrial Applications and Future of Quantum Computing. **CO's-CO5**

Self-Learning Topics: Quantum supremacy

Board of Studies: Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries) **CO's/PO's/PSO's**

1. Quantum Computing in Cloud Platforms (IBMQ, Azure Quantum) CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Quantum Machine Learning CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
3. Quantum Cryptography in Cyber security CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
4. Real-time Applications of Quantum Computing CO1-CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Eric R. Johnston, Nic Harrigan – *Programming Quantum Computers*
2. Christine Corbett Moran – *Mastering Quantum Computing with IBM QX*

Reference Books:

1. Michael Nielsen & Isaac Chuang – *Quantum Computation and Quantum Information*
2. Phillip Kaye – *Introduction to Quantum Computing*

Web References:

1. NPTEL Quantum Computing Courses
2. IBM Quantum Experience
3. Qiskit Documentation

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSSC05|

Prompt Engineering Lab
(SKILL ENHANCEMENT COURSE)

3 0 0 3 (Common to CSE)

Course Objectives:

1. Apply iterative prompting for clarity and context.
2. Create varied prompts to steer model outputs.
3. Construct chain-of-thought and structured prompts.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSSC05.1	Understand prompt engineering as the process of crafting inputs to elicit desired outputs from large language models (LLMs).	3	2	-	-	1	2	1	1	L1,L2,L4
R24CSSC05.2	Design prompts that include contextual details and explicit output formats to guide the model towards desired responses.	2	3	3	-	2	3	3	2	L2,L3
R24CSSC05.3	Understand the Importance of Structured Outputs and Reasoning Techniques in Real-World Applications.	2	2	3	2	3	3	3	2	L3,L4
R24CSSC05.4	Recognize the constraints of large language models (LLMs) regarding static knowledge and the necessity for integrating external data sources to enhance their capabilities.	1	3	2	2	2	2	3	2	L3,L4
R24CSSC05.5	Understand the concept of using LLMs for automated evaluation of prompts.	1	2	1	1	2	1	2	3	L4,L5

List of experiments

Unit I: Foundations of Prompt Engineering: Definition of prompt engineering, Distinction between prompt engineering and model fine-tuning, Motivation and benefits of prompt engineering, Core principles of effective prompt design, Anatomy of a prompt, Setting up the Python environment for LLM interaction, Iterative prompting lifecycle, Common prompt pitfalls and remediation

Lab Experiments:

1. Environment & Connectivity: Install required packages (e.g., transformers, openai); securely configure the API key; run a simple “Hello, world” prompt to verify model access.
2. Baseline vs. Enhanced Prompts: Execute a naïve prompt (“Write a one-paragraph bio of Ada Lovelace.”) and an enhanced prompt that adds role framing, specificity, and explicit format instructions; compare both outputs for relevance, completeness, and style.
3. Iterative Refinement on a Simple Task: Summarize the plot of the Shakespearean play Romeo and Juliet in two sentences through three rounds of prompt tweaking:
 - a) Minimal instruction.
 - b) Addition of length and style constraints
 - c) Specification of key content elements (setting and theme)

Document how each iteration changes and improves the result.

4. Diagnosing Prompt Failures & Edge Cases: Craft a vague or contradictory prompt; analyze the failure mode (ambiguity, missing context, or format errors); refine the prompt by adding examples or clarifying instructions.

Unit II: Advanced Prompt Patterns & Techniques: Enhanced prompt anatomy: contextual detail and explicit output specifications, Few-shot in-context prompting, Prompt structuring and template design, Role-based prompting to establish personas or system behavior, Negative prompting to filter or suppress undesired content, Constraint specification and instruction enforcement (e.g., length, format), Iterative prompt refinement and optimization

Lab Experiments:

1. Few-Shot vs. Zero-Shot Comparison: Design and execute a zero-shot prompt and a few-shot prompt (with 2–3 exemplar input-output pairs) for a chosen text task (e.g., sentiment classification or translation); compare outputs for accuracy, consistency, and adherence to examples.
2. Role-Based & Negative Prompting: Craft a role-based prompt to establish a specific persona (e.g., “You are a financial advisor...”); then create a negative prompt to suppress undesired content (e.g., “Do not mention any brand names”); evaluate how each influences the model’s response.
3. Constraint Specification & Iterative Refinement: Select an open-ended task (e.g., summarizing a technical article); issue a basic prompt; identify failures in length or format; refine the prompt by adding explicit constraints (word count, bullet format, etc.); document improvements over two refinement cycles.

Unit III: Structured Output & Reasoning Techniques: Importance of structured outputs for real-world applications, Prompting for specific formats (lists, tables, Markdown), Generating valid JSON

and YAML via explicit instructions, Eliciting chain-of-thought reasoning in zero-shot prompts, Decomposing complex tasks into manageable sub-tasks

Lab Experiments:

1. **Structured Format Prompting:** Instruct the model to output information as bullet lists and Markdown tables (e.g., “List three benefits of daily exercise in a Markdown table with columns ‘Benefit’ and ‘Description.’”); verify the output matches the requested structure.
2. **JSON/YAML Generation:** Provide a brief dataset description (e.g., three books with title, author, publication year) and prompt the model to produce valid JSON or YAML; use a parser to validate syntax and refine the prompt if errors occur.
3. **Chain-of-Thought & Task Decomposition:** Present a multi-step problem (e.g., a logic puzzle) and apply zero-shot CoT prompting (e.g., “Let’s think step by step. Explain your reasoning before the final answer.”); separately, decompose the problem into sequential sub-questions, collect partial answers, combine them, and compare accuracy against a direct-answer baseline.

Unit IV: Retrieval-Augmented Generation & LangChain Workflows: Limitations of LLM internal knowledge, Need for external data sources, Introduction to Retrieval-Augmented Generation (RAG), Overview of RAG architecture (indexing vs. retrieval + generation), Getting started with LangChain for LLM applications, Basics of LangChain Expression Language (LCEL), Simplified indexing pipeline: document loading & text splitting, Fundamentals of embeddings and vector stores, Building a basic retrieval-generation pipeline with an LCEL chain

Lab Experiments:

1. **Building a Simple LCEL Chain:** Create a minimal LCEL script that accepts a fixed instruction (e.g., “Summarize this text: ...”), passes it to an LLM, and prints the result; verify end-to-end execution.
2. **Basic Data Indexing for RAG:** Load a small collection of documents; split into uniform chunks (e.g., 200 tokens); generate embeddings for each chunk; store them in an in-memory vector store; inspect for consistency.

Constructing & Running a Basic RAG Chain: Build a pipeline that: Receives a user query

Retrieves the top-k relevant chunks

Constructs a combined prompt with context + query

Send it to the LLM

Returns the answer

Test with sample queries and compare factual accuracy against a prompt without retrieval.

Unit V: Agents, Multimodal AI & Ethical Evaluation: Introduction to LLM agents and their basic architecture, Overview of multimodal AI models (VLMs), Prompting for text-to-image generation and image understanding, Importance of prompt evaluation beyond subjective judgment, Manual evaluation techniques (heuristic checks for accuracy, relevance, format), Introduction to “LLM-as-Judge” for

automated evaluation, Security considerations (prompt injection, sensitive-information risks), Prompt-based mitigation strategies for safety and robustness, Ethical concerns (bias, misinformation, data privacy), Brief exploration of UI frameworks (Streamlit/Gradio) for deploying prompt-driven apps, Adapting to the evolving nature of prompt engineering through continuous learning

Lab Experiments:

1. Building a Simple LLM Agent: Register a tool (e.g., a calculator function) and craft prompts that instruct the agent to invoke it when required; implement using Lang Chain or a function-calling API; test on queries requiring tool execution.
2. Multimodal Prompting Exploration: Generate images from detailed text prompts; feed one generated image into an image-understanding model or API with an appropriate prompt; compare the returned caption to the original prompt to evaluate alignment.

Prompt Evaluation & Ethics Workshop: Select two existing prompts and generate multiple outputs; apply manual heuristic checks for accuracy, relevance, and format compliance.

Use an “LLM-as-Judge” prompt (e.g., “Rate these outputs on a scale of 1–5 for clarity and correctness.”) to automate evaluation.

Design a prompt- injection test (e.g., “Ignore previous instructions...”), observe the response, then refine system prompts to mitigate the vulnerability.

Text Books

1. Hands-On Large Language Models — Jay Alammar and Maarten Grootendorst, O’Reilly Media.
 - Covers prompt engineering, LLM fundamentals, and practical AI applications.
2. Generative AI on AWS — Chris Fregly and Antje Barth, O’Reilly Media.
 - Explains generative AI workflows, prompting methods, and deployment concepts.
3. Deep Learning — Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press.
 - Foundation concepts behind neural networks and generative AI systems.

Reference Books

1. Artificial Intelligence: A Modern Approach — Stuart Russell and Peter Norvig.
2. Natural Language Processing with Transformers — Lewis Tunstall, Leandro von Werra, Thomas Wolf.
3. Speech and Language Processing — Daniel Jurafsky and James H. Martin.

E-Resources

1. [OpenAI Prompt Engineering Guide](#)
 - Covers iterative prompting, structured prompting, and prompt design techniques.
2. [Prompt Engineering Guide](#)
 - Comprehensive tutorials on zero-shot, few-shot, chain-of-thought, and role prompting.
3. [Anthropic Prompt Engineering Overview](#)
 - Best practices for context building and structured prompting.

**Chairperson
Board of Studies (CSE)**

R24MC06

Technical Paper Writing and IPR
(Mandatory Course)

2 0 0 0

Course Objectives:

1. To develop the ability to write clear, structured, and professional technical documents.
2. To understand grammar, style and formatting required for technical writing.
3. To learn drafting, reviewing presenting technical papers.
4. To gain proficiency in using word processing tools for documentation.
5. To understand Intellectual Property Rights (IPR) and patenting process.

Course Outcomes

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

Mapping with POs and PSOs															Dok
Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
R24MC06.1	Demonstrate ability to write clear and structured technical documents	-	-	-	-	2	-	-	-	-	3	2	2	-	L1, L2
R24MC06.2	Apply drafting and editing techniques in report writing	-	2	-	-	3	-	-	-	-	3	2	2	-	L2, L3
R24MC06.3	Demonstrate proof reading and summarization skills	-	2	-	-	2	-	-	-	-	3	-	2	-	L3, L4
R24MC06.4	Use advanced MS Word tools for technical documentation.	-	-	-	-	3	-	-	-	-	2	-	2	2	L4, L5
R24MC06.5	Understand IPR concepts and patenting process	2	2	-	-	-	2	2	3	-	2	-	-	-	L5, L6

SYLLABUS**UNIT I:****10 Hours**

Introduction to Technical Writing: Introduction to technical writing, technical sentences formation, use of transitions, use of tenses in technical writing.

Planning and Structuring: Planning reports, identifying audience, voice, formatting and structuring reports, sections of a technical report, minutes of meeting writing. **CO's-CO1**

Self-Learning Topics: Case studies of technical reports.

UNIT II:**6 Hours**

Drafting and Editing Techniques: Drafting reports: Use of drafts, illustrations and graphics,

Final Editing: Grammar, spelling, readability, writing in plain English, jargon usage, punctuation, paragraph writing, avoiding ambiguity. **CO's-CO2**

Self-Learning Topics: Writing technical blogs.

UNIT III:**8 Hours**

Proofreading and Presentation Skills: Proof reading techniques, summarization methods, practice exercises.

Presentation Skills: Printed presentation, verbal presentation, introduction to proposals.

Self-Learning Topics: Poster and paper presentation techniques. **CO's–CO3**

UNIT IV: **8 Hours**

Word Processing Tools for Documentation: Using MSWord: Table of contents, indexing, outlines, comments, track changes. Footnotes, endnotes, citations, bibliography, Document comparison, document protection, macros, formatting tools. **COs–CO4**

Self-Learning Topics: Latex basics.

UNIT V: **9 Hours**

Intellectual Property Rights (IPR):

Nature of Intellectual Property: Patents, copyrights, trademarks, designs.

Process of Patenting: Research, innovation, patent filing, development.

International Scenario: Global IPR policies and cooperation. **CO's–CO5**

Self-Learning Topics: Case studies on patent disputes

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk (To be Delivered by SMEs from Industries) **CO's/PO's/ PSO's**

1. Research Paper Writing using IEEE Format CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Tools for plagiarism detection (Turnitin, Grammarly) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
3. Patent filing in startups and innovation ecosystems CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
4. Role of IPR in software industry, CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2, PSO3, PSO4

Text Books:

1. Technical Communication –Meenakshi Raman & Sangeeta Sharma
2. English for Writing Research Papers –Adrian Wallwork

Reference Books:

1. Day R – *How to Write and Publish a Scientific Paper*
2. ML A Handbook for Writers

Web References:

1. SWAYAM Courses on Technical Writing
2. NPTEL Technical Communication
3. IEEE Author Guidelines

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35

L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**Program: B. Tech Computer Science and Engineering****Regulation: R24****IV Year II Semester - Course Structure**

S. No	Category	Course Code	Course Title	Hours per Week			
				L	T	P	C
1	PE-5	R24CSPE5.1 R24CSPE5.2 R24CSPE5.3 R24CSPE5.4	1. Soft Computing 2. Mobile Ad-hoc Networks 3. Natural Language Processing 4. UI/UX Design 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
2	OE-4		OPEN ELECTIVE-4	3	0	0	3
3	PR	R24PR01	Internship & Project Work	0	0	20	12
Total				06	0	20	18

Category	Courses	Credits
PE- Professional Elective	1	3
OE-Open Elective	1	3
PJ- Internship & Project work	1	10
Total	03	16

Note: Student need to do at least ONE MOOC/NPTEL Course (of 3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R24 Regulations); they are allowed to register one semester in advance.

Course Objectives:

1. Understand the fundamental concepts of soft computing techniques such as neural networks, fuzzy logic, genetic algorithms, and hybrid systems.
2. Analyze different neural network architectures and learning algorithms for solving complex problems.
3. Apply fuzzy logic principles to model uncertainty and develop intelligent decision-making systems.
4. Understand and implement genetic algorithms for optimization and search problems.
5. Design and evaluate hybrid soft computing systems for real-world applications.

Course Outcomes:

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 5.1.1	Understand fundamentals of soft computing and distinguish it from hard computing.	3	2	1	1	1	-	-	-	-	-	-	2	1	L1 L2
R24CSPE 5.1.2	Analyze and apply neural network models for problem solving.	3	3	2	2	2	-	-	-	-	-	-	3	2	L2 L3
R24CSPE 5.1.3	Apply fuzzy logic principles to design intelligent systems.	3	3	3	2	2	-	-	-	-	-	-	3	2	L3 L4
R24CSPE 5.1.4	Understand and implement genetic algorithms for optimization problems.	3	3	3	2	3	-	-	-	-	-	-	3	3	L4 L5
R24CSPE 5.1.5	Analyze and design hybrid soft computing systems for real-world applications.	3	3	3	3	3	1	1	-	1	1	1	3	3	L5 L6

SYLLABUS

UNIT I:

15 Hours

Basic elements of Soft Computing: Difference between Hard Computing & Soft Computing- Introduction to Neural Networks - Fuzzy Logic - Genetic Algorithms - Hybrid Systems.

CO's-CO1

Self-Learning Topics: Need for soft computing in modern application, Components of soft computing.

UNIT II:

14Hours

Neural Networks: Neural Networks: What is Neural Network, Basic Models of Neural Networks - McCulloch-Pitts Neuron - Hebb Network, supervised learning network: perceptron networks – adaptive linear neuron, multiple adaptive linear neuron, BPN, RBF, TDNN-associative memory

network: auto-associative memory network, hetero-associative memory network, BAM, hop field networks, iterative auto associative memory network & iterative associative memory network – unsupervised learning networks: Kohonen self organizing feature maps. **CO's–CO2**

Self-Learning Topics: Mathematical model of a neuron, working principle of McCulloch-Pitts neuron, Bidirectional Associative Memory (BAM)

UNIT III: 13 Hours

Fuzzy Logic System: Introduction to fuzzy logic, classical sets and fuzzy sets, fuzzy set operations, fuzzy relations, fuzzy composition, natural language and fuzzy interpretations, fuzzy inference system, fuzzy controllers. **CO's–CO3**

Self-Learning Topics: Difference between crisp logic and fuzzy logic Natural Language & Fuzzy Interpretation- Membership functions for linguistic terms

UNIT IV: 14 Hours

Genetic algorithm: Fundamental, basic concepts, working principle, encoding, fitness function, reproduction,

Genetic modeling: Inheritance operator, cross over, inversion & deletion, mutation operator, Bitwise operator, Generational Cycle, Convergence of GA, Applications & advances in GA, Differences & similarities between GA & other traditional methods. **COs–CO4**

Self-Learning Topics: Basic terminology: population, chromosome, gene, Definition and role of fitness function.

UNIT V: 13 Hours

Hybrid Systems: Sequential Hybrid Systems, Auxiliary Hybrid Systems, Embedded Hybrid Systems, Neuro-Fuzzy Hybrid Systems, Neuro-Genetic Hybrid Systems, Fuzzy-Genetic Hybrid Systems.

GA based Back propagation Networks: GA based Weight Determination, K - factor determination in Columns. **CO's-CO5**

Self-Learning Topics: Real-world applications of hybrid systems, Use of GA to optimize weights Board of Studies : Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Books:

1. Soft Computing – Advances and Applications - Jan 2015 by B.K. Tripathy and J. Anuradha – Cengage Learning
2. S.N.Sivanandam, SNDeepa, Principles of Soft Computing, 2nd Edition, Wiley India, 2007
3. V.Kecman, Learning and Soft Computing, Pearson Education, India, 2006.
4. S.Rajasekaran & G.A.Vijayalakshmi Pai, Neural Networks, Fuzzy Logic & Genetic Algorithms, Synthesis & applications, PHI Publication, 1st Edition, 2009.

Reference Books:

1. S.Haykins, Neural networks: a comprehensive foundation Pearson Education, India, 2002
2. David E. Goldberg, “Genetic Algorithms-In Search, optimization and Machine learning”, Pearson Education
3. J.S.R.Jang, C.T.Sun and E.Mizutani, “Neuro-Fuzzy and Soft Computing”, Pearson Education 2004.
4. Timothy J. Ross, “Fuzzy Logic with Engineering Applications”, McGraw-Hill International

Editions, 1995

5. G.J.Klir&B.Yuan, “Fuzzy Sets & Fuzzy Logic”,PHI,1995.

6. X.YuandM.Gen, Introduction to EvolutionaryAlgorithms,SpringerVerlag,2010.

Web References:

1. <https://sci2s.ugr.es/links>

2. <https://pravin-hub-rgb.github.io/BCA/resources/sem5/sc/index.html>

3. <https://vtupulse.com/soft-computing/soft-computing-video-tutorial>.

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSPE5.2|

**Mobile Ad-hoc Networks
(PROFESSIONAL ELECTIVE-V)**

3 0 0 3 (Common to CSE)

Course Objectives:

1. Understanding of the current topics in MANETs and WSNs, both from an industry and research point of views.
2. Understanding of the principles of mobile ad hoc networks (MANETs) and what distinguishes them from infrastructure-based networks. .
3. Understand how proactive routing protocols function and their implications on data transmission delay and bandwidth consumption.
4. Know about routing protocol
5. Analyze various routing algorithms

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE5.2.1	Understanding of the current topics in MANETs and WSNs, both from an industry and research point of views.	3	2	-	-	-	2	3	-	L2
R24CSPE5.2.2	Understanding of the principles of mobile ad hoc networks (MANETs) and what distinguishes them from infrastructure-based networks. .	2	3	-	3	-	2	3	-	L2
R24CSPE5.2.3	Understand how proactive routing protocols function and their implications on data transmission delay and bandwidth consumption.	2	2	3	2	3	2	3	-	L2
R24CSPE5.2.4	Know about routing protocol	1	2	3	2	3	2	3	-	L2
R24CSPE5.2.5	Analyze various routing algorithms	1	2	2	2	3	2	3	-	L3

SYLLABUS

UNIT-I: INTRODUCTION: Introduction to adhoc networks-definition, characteristics features, applications. Characteristics of Wireless channel, Adhoc Mobility Models:- Indoor and outdoor models.

COs–CO1

UNIT-II: MEDIUM ACCESS PROTOCOLS: MAC Protocols: design issues, goals and classification. Contention based protocols- with reservation, scheduling algorithms, protocols using directional antennas. IEEE standards: 802.11a, 802.11b, 802.11g, 802.15. HIPERLAN.

COs–CO2

UNIT-III: NETWORK PROTOCOLS: Routing Protocols: Design issues, goals and classification. Proactive Vs reactive routing, Unicast routing algorithms, Multicast routing algorithms, hybrid routing algorithm, Energy aware routing algorithm, Hierarchical Routing, QoS aware routing.

COs–CO3

UNIT-IV: END-END DELIVERY AND SECURITY: Transport layer Security Issues issues in designing- Transport layer classification, adhoc transport adhoc networks: issues and challenges, network security attacks, secure routing protocols

COs–CO4

UNIT-V: CROSS LAYER DESIGN AND INTEGRATION: Cross layer Design: Need for cross layer design, cross layer optimization, parameter optimization techniques, Cross layer cautionary perspective. Integration of adhoc with Mobile IP networks.

COs–CO5

Text Books:

1. C.Siva Ram Murthy and B.S.Manoj, Ad hoc Wireless Networks Architectures and protocols, 2nd edition, Pearson Education. 2007
2. Charles E. Perkins, Ad hoc Networking, Addison – Wesley, 2000

References Books:

1. Stefano Basagni, Marco Conti, Silvia Giordano and Ivan stojmenovic, Mobilead hoc networking, Wiley IEEE press, 2004.
2. Mohammad Ilyas, The handbook of adhoc wireless networks, CRC press, 2002.
3. T. Camp, J. Boleng, and V. Davies “A Survey of Mobility Models for Ad Hoc Network

4. Research,” Wireless Commun. and Mobile Comp., Special Issue on Mobile Ad Hoc Networking Research, Trends and Applications, vol. 2, no. 5, 2002, pp. 483–502.
5. A survey of integrating IP mobility protocols and Mobile Ad hoc networks, Fekri M. Abduljalil and Shrikant K. Bodhe, IEEE communication Survey and tutorials, v no.1 2007
6. V.T. Raisinhani and S.Iyer “Cross layer design optimization in wireless protocol stacks”Comp. communication, vol 27 no. 8, 2004.
7. V.T.Raisinhani and S.Iyer,”ÉCLAIR; An Efficient Cross-Layer Architecture for wireless protocol stacks”,World Wireless cong., San francisco,CA,May 2004.
8. V.Kawadia and P.P.Kumar,”A cautionary perspective on Cross-Layer design,”IEEE Wireless commn., vol 12, no 1,2005.

MOOC Course:

https://onlinecourses.nptel.ac.in/noc18_cs09/preview

**Chairperson
Board of Studies (CSE)**

Course Objectives:

1. Introduce to some of the problems and solutions of NLP and their relation to linguistics and statistics.
2. To understand linguistic phenomena and learn to model them with formal grammars.
3. To understand and carry out proper experimental methodology for training and evaluating empirical NLP systems.
4. To learn how to manipulate probabilities, construct statistical models over strings and trees
5. To design, implement, and analyze NLP algorithms. Able to design different language modeling.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 5.3.1	Show sensitivity to linguistic phenomena and an ability to model them with formal grammars	3	2	1	-	1	-	-	-	-	-	2	3	1	L1, L2
R24CSPE 5.3.2	Understand and carry out proper experimental methodology for training and evaluating empirical NLP systems.	3	3	2	2	2	-	-	-	1	-	2	3	2	L2, L3
R24CSPE 5.3.3	Able to manipulate probabilities, construct statistical models over strings and trees	3	3	2	2	2	-	-	-	-	-	2	3	2	L3, L4
R24CSPE 5.3.4	Analyze semantic representations and meaning extraction techniques	3	3	2	2	2	-	-	-	1	-	2	3	2	L4, L5
R24CSPE 5.3.5	Able to design, implement and analyze NLP algorithms. Able to design different language modeling Techniques	3	3	3	2	2	1	-	1	1	-	3	3	3	L5, L6

SYLLABUS**UNIT I:****15Hours**

Natural Language processing (NLP): Introduction, Applications or Use cases of NLP, Components of NLP, Steps in NLP, Finding the Structure of Words: Words and Their Components, Lexemes, Morphemes, Morphology, Problems in morphological processing,

Typology, Morphological Typology

CO's–CO1

Self-Learning Topics: Tokenization techniques in NLP, NLTK library, Real-world NLP applications

UNIT II:

15Hours

Syntax Analysis: Parsing Natural Language, Tree banks: A Data-Driven Approach to Syntax, **Representation of Syntactic Structure:** Syntax Analysis using Dependency Graph, Syntax Analysis using Phrase Structure Trees,

Parsing Algorithms: Shift Reduce Parsing, Hyper Graphs and Chart Parsing (CYK Parsing), **Models for ambiguity Resolution in Parsing:** Probabilistic Context Free Grammar, Generative Models, Discriminative models for Parsing. **CO's–CO2**

Self-Learning Topics: SpaCy parsing toolkit, Transition-based vs graph-based parsing

UNITIII:

13Hours

Language Modeling: Introduction, N-Gram Models, Language Model Evaluation, Parameter Estimation, Language Model Adaptation, Types of Language Models, Language-Specific Modeling Problems. **CO's–CO3**

Self-Learning Topics: Smoothing techniques (Laplace, Good-Turing), Wordembeddings

UNIT IV:

15Hours

Semantic Parsing: Introduction, Semantic Interpretation, System Paradigms, Word Sense Systems, Software. **COs–CO4**

Self-Learning Topics: Named Entity Recognition (NER), Word Sense Disambiguation (WSD)

UNIT V:

14Hours

Predicate-Argument Structure: Meaning Representation Systems, Software.

Discourse Processing: Cohesion, Reference Resolution, Discourse Cohesion and Structure

Self-Learning Topics: Text summarization techniques, Chatbot design basics **CO's-CO5**

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Text Book:

1. Multilingual Natural Language Processing Applications: From Theory to Practice–Daniel M.Bikel and Imed Zitouni, Pearson Publication.
2. Speech and Natural Language Processing-DanielJurafsky&JamesHMartin,Pearson Publications.

Reference Books:

1. Natural Language Processing and Information Retrieval: TanvierSiddiqui,U.S.Tiwary

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25

AIETM|R24|CSE|R24CSPE5.3|Natural Language Processing

L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

R24CSPE5.4**UX/UI Design****3 0 0 3**

(Professional Elective-V)

Course Objectives:

The objectives of this course for the student are to:

1. To introduce the fundamental concepts of UX/UI design, design thinking, and user-centered design principles.
2. To enable students to understand and apply user research techniques and visual design principles for identifying user needs.
3. To develop skills in structuring information architecture and creating wire frames for effective user experience.
4. To provide knowledge of interaction design principles and hands-on experience with prototyping tools.
5. To equip students with skills to perform usability testing, design responsive interfaces, and build professional design portfolios.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSPE 5.4.1	Recall and explain UX/UI design concepts, design thinking stages, and user-centered design principles.	2	1	1	-	1	-	-	1	1	1	-	1	1	L1, L2
R24CSPE 5.4.2	Apply user research methods and visual design principles to identify user requirements and design elements.	2	3	2	2	1	2	1	2	2	1	-	2	1	L2, L3
R24CSPE 5.4.3	Analyze and design information architecture, user flows, and wireframes for applications and websites	2	2	3	2	2	1	1	1	3	2	1	3	2	L3, L4
R24CSPE 5.4.4	Develop interactive prototypes using design tools and apply interaction design principles.	2	2	3	2	3	1	1	1	3	2	1	3	3	L4, L5
R24CSPE 5.4.5	Evaluate usability of interfaces and create responsive UI designs and professional design portfolios.	2	3	3	3	2	2	2	2	3	2	1	3	3	L5, L6

SYLLABUS**UNITI:****14Hours****Introduction to UX/UI Design Tools**

Definition of UX and UI Design, The design process, User-centered design principles, Importance of UX/UI design in the digital world. **CO's–CO1**

Self-Learning Topics: Case studies of good and bad UX, Understand design fundamentals

UNITII: 15Hours

User Research in Practice

Conducting user interviews and surveys for a chosen design project. Analyzing user data to identify key insights and pain points. Visual Design: Design principles (balance, hierarchy, contrast, etc.), Typography and color theory, Visual identity and branding. **CO's–CO2**

Self-Learning Topics: User research methods, Understand user needs & visual aesthetics

UNITIII: 15 Hours

Information Architecture: Web site and application structure, Content organization and navigation systems, Card sorting and site maps, and developing user person as based on user research data. Creating user scenarios to illustrate user needs and interactions. Creating low-fidelity wireframes to illustrate the basic layout and functionality. Applying design principles for usability and user flow. Visual Design Principles, implementing design principles (e.g., typography, color theory) to create visually appealing interfaces. **CO's–CO3**

Self-Learning Topics: Usability principles & user flow, Structure user-friendly systems

UNITIV: 15 Hours

Interaction Design and Site Mapping: User interface elements and their functionality, Interaction design principles (affordance, feedback, etc.), Wire framing and prototyping techniques developing sitemaps and user flows for a chosen application/website. Evaluating the effectiveness of information architecture for intuitive navigation. Design Tools and Resources: Introduction to industry-standard design software (e.g., Figma, Sketch, and Adobe XD). **COs–CO4**

Self-Learning Topics: User flows & sitemap creation, Develop interactive designs

UNITV: 13Hours

Prototyping and User Testing: Building interactive prototypes using design software functionalities. Conducting usability testing with users and iterating on designs based on feedback. Developing a basic style guide for visual consistency. Design for Different Platforms (Apply, Create) Exploring design considerations for responsive web and mobile interfaces. Adapting UI design for touch-based interactions. Design Presentation and Portfolio (Communicate, Create) Learning effective techniques for presenting design ideas and prototypes. Building a design portfolio show casing the completed projects from the course. **COs–CO5**

Self-Learning Topics: Portfolio development & presentation, Build complete UX solutions

Board of Studies :Computer Science and Engineering

Approved in BoS No : 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries) CO's/PO's/PSO's

1. UX in Product-Based Companies(like Google, Amazon)
2. Design Systems and Scalable UI Development
3. Accessibility in UX Design (Inclusive Design)

Text Books:

1. DonaldNorman,The Design of Everyday Things, Basic Books.
2. SteveKrug, Don't Make Me Think, Pearson Education.
3. JesseJamesGarrett,The Elements of User Experience, New Riders.

Reference Books:

1. Alan Cooper, About Face: The Essentials of Interaction Design, Wiley.
2. Jeff Gothelf, Lean UX , O'Reilly.
3. William Lidwell, Universal Principles of Design, Rockport.
4. Bill Buxton, Sketching User Experiences, Morgan Kaufmann.
5. Susan Weinschenk, 100 Things Every Designer Needs to Know About People, New Riders.

Web References:

1. www.interaction-design.org
2. www.nngroup.com
3. www.figma.com
4. www.coursera.org
5. www.dribbble.com

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. To impart basic concepts of computer architecture and organization.
2. To explain key skills of constructing cost-effective computer systems.
3. To familiarize students with basic CPU organization.
4. To help students understand various memory devices.
5. To facilitate students in learning I/O communication and multiprocessors.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 4.1.1	Understand computer structure and arithmetic operations.	3	-	2	1	2	-	-	-	-	1	-	3	1	L1, L2
R24CSOE 4.1.2	Describe instruction formats and CPU organization.	3	3	2	2	2	-	-	2		1	-	3	2	L2, L3
R24CSOE 4.1.3	Apply register transfer and micro-operations.	3	3	3	2	3	-	-	-	-	1	-	3	2	L3, L4
R24CSOE 4.1.4	Analyze memory hierarchy and memory management techniques.	3	-	3	3	3	-	-	-	-	1	-	3	2	L4, L5
R24CSOE 4.1.5	Explain I/O organization and multiprocessor systems.	3	2	2	2	3	2	3	-	1	1	1	2	2	L5, L6

SYLLABUS**UNITI:****15Hours**

Structure of computers & computer arithmetic: Computer types, Functional units, Basic operational concepts, Von Neumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multi computer, Data representation, Fixed and Floating point, Error detection and correction codes. Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations.

CO's-CO1

Self-Learning Topics: Floating point representation, Error correction techniques, Number systems

UNITII:**13Hours**

Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer Instructions and Instruction cycle, Timing and Control, Memory-Reference Instructions, Input-Output and interrupt, Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, CISC and RISC architecture.

CO's-CO2

Self-Learning Topics: Modern processors, Instruction pipelining basics, addressing mode examples

UNIT III:

14Hours

Register Transfer and Micro-Operations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit. Micro-programmed Control, Control Memory, Address Sequencing, Design of Control Unit.

CO's–CO3

Self-Learning Topics: RTL examples, Control unit design, Micro program sequencing

UNITIV:

13 Hours

Memory System: Memory Hierarchy, Semi conductor Memories, RAM, ROM, Types of ROM, Cache Memory, Virtual memory, Paging, Secondary Storage, RAID.

CO's–CO4

Self-Learning Topics: Cache mapping techniques, Virtual memory implementation, RAID levels

UNITV:

13Hours

Input Output and Multiprocessors: I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Communication and Synchronization, Cache Coherence.

CO's–CO5

Self-Learning Topics: DMA controllers, Cache coherence protocols, Parallel processing concepts

Board of Studies : Computer Science and Engineering

Approved in BoS No:03

Approved in ACM No:03

Expert Talks:

1. Modern Processor Architectures and CPU Design CO's/PO's/PSO'sMapped:CO2,CO3,CO5, PO1, PO2, PO3, PO5
2. Memory Systems and High-Performance Computing CO's/PO's/PSO'sMapped:CO4,CO5, PO1, PO2, PO3, PO5
3. I/O Systems and Embedded Computing Applications CO's/PO's/PSO's Mapped: CO1, CO5,PO1, PO3, PO5

Text Books:

1. M.MorrisMano, Computer System Architecture,3rdEdition,Pearson/PHI.
2. CarlHamacher,ZvonkoVranesic,Safwat Zaky, Computer Organization,McGrawHill.

Reference Books:

1. WilliamStallings,Computer Organization and Architecture, PrenticeHall.
2. AndrewS.Tanenbaum, Structured Computer Organization, PearsonEducation.
3. JohnP.Hayes, Computer Architecture and Organization, TataMcGrawHill.

Internal Assessment Pattern

Cognitive Level	InternalAssessment–1(%)	InternalAssessment–2(%)
L1(Remember)	35	-
L2(Understand)	40	-
L3(Apply)	25	25
L4(Analyze)	-	35

L5(Evaluate)	-	25
L6(Create)	-	15

Chairperson
Board of Studies(CSE)

R24CSOE4.2 FUNDAMENTAL OF INFORMATION SECURITY 3 0 0 3
(OPEN ELECTIVE-IV)

Course Objectives:

1. To understand the fundamentals of information security concepts and principles.
2. To analyze threats, vulnerabilities and cyber-attacks in computer systems.
3. To understand cryptography, authentication and access control mechanisms.
4. To illustrate network security, web security and secure communication techniques.
5. To understand cyber laws, ethical hacking concepts and security management practices.

Course Outcomes:

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 4.2.1	Describe fundamentals of information security principles, threats, and vulnerabilities	3	-	-	-	2	-	-	-	-	-	2	3	2	L1 L2
R24CSOE 4.2.2	Analyze cryptographic techniques and authentication mechanisms	3	3	-	2	3	-	-	2	-	-	2	3	2	L2 L3
R24CSOE 4.2.3	Analyze process synchronization mechanisms and dead lock handling techniques.	3	3	2	2	3	-	-	-	2	-	2	3	3	L3 L4
R24CSOE 4.2.4	Analyze system security, web security, and malware protection techniques	3	2	2	2	3	-	-	-	-	-	2	3	2	L4 L5
R24CSOE 4.2.5	Evaluate cyber laws, ethical hacking practices, and information security management	3	2	3	2	3	2	2	-	-	2	3	-	-	L5 L6

SYLLABUS0

UNIT I:

15 Hours

Introduction to Information Security: Overview of Information Security, Need for Security, Security Goals – Confidentiality, Integrity, Availability, Threats and Vulnerabilities, Types of Attacks, Security Services and Mechanisms. Security Models and Standards: OSI Security Architecture, Security Policies, Risk Management, Security Standards and Frameworks, Cyber Security Overview. **CO’s–CO1**

Self-Learning Topics: Evolution of Cyber Security and Recent Security Incidents.

UNIT II:

15Hours

Cryptography: Classical Encryption Techniques, Symmetric Key Cryptography, Asymmetric Key

AIETM|R24|CSE|R24CSOE4.2|Fundamental of Information Security

Cryptography, Public Key Infrastructure, Hash Functions, Digital Signatures. Authentication Mechanisms: User Authentication, Password Security, Multifactor Authentication, Biometric Authentication, Key Management Techniques, **CO's–CO2**

Self-Learning Topics: Modern Encryption Standards and Applications.

UNIT III: **15 Hours**

Network Security: Network Security Concepts, Firewalls, Intrusion Detection Systems, Intrusion Prevention Systems, VPNs, Secure Socket Layer (SSL), Transport Layer Security (TLS). Wireless and Communication Security: Wireless Security Protocols, Email Security, Secure Communication Techniques, Network Monitoring and Packet Analysis. **CO's–CO3**

Self-Learning Topics: Cloud Security and IoT Security Challenges

UNIT IV: **15 Hours**

System Security: Operating System Security, Malware, Viruses, Worms, Trojans, Ransomware, Antivirus Mechanisms, System Hardening Techniques. **Web Security:** Web Application Security, SQL Injection, Cross Site Scripting (XSS), Session Hijacking, Secure Coding Practices, Security Testing. **COs–CO4**

Self-Learning Topics: Mobile Security and Secure Software Development Practices.

UNIT V: **10 Hours**

Cyber Laws and Ethics: Cyber Crimes, Cyber Laws, Intellectual Property Rights, Ethical Hacking Concepts, Digital Forensics Basics. **Security Management:** Information Security Management System (ISMS), Disaster Recovery, Business Continuity Planning, Incident Response, Security Auditing and Compliance. **CO's-CO5**

Self-Learning Topics: Cyber Security Governance and Security Awareness Programs

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Expert Talk (To be Delivered by SMEs from Industries) **CO's/PO's/PSO's**

1. Information Security in Cloud Computing (Cloud Threats, Security Models, Identity and Access Management in Cloud Platforms) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Ethical Hacking and Penetration Testing (Security Assessment Tools, Vulnerability Analysis, Real-Time Attack Demonstrations) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
3. Cyber Security in Industry Applications (Banking Security, Healthcare Data Protection, Enterprise Security Management) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Cryptography and Network Security– William Stallings, 7th Edition, Pearson.
2. Information Security Principles and Practices–Mark Stamp, 3rd Edition, Wiley.
3. Cyber Security Essentials –James Graham, Richard Howard, Ryan Olson.

Reference Books:

1. Network Security Essentials– William Stallings.
2. Information Security: Principles and Practice –Deven Shah.

3. Cyber Security and Cyber Laws –AlfredBasta.
4. Principles of Information Security–MichaelE.WhitmanandHerbertJ. Mattord.

Web References:

1. <https://nptel.ac.in/>
2. <https://www.coursera.org/>
3. https://www.tutorialspoint.com/computer_security/
4. <https://owasp.org/>

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	100	100

**Chairperson
Board of Studies(CSE)**

Course Objectives:

1. Learn aspects of security, importance of data gathering, foot printing and system hacking.
2. Learn tools and techniques to carry out penetration testing.
3. How intruders escalate privileges?
4. Explain Intrusion Detection, Policy Creation, Social Engineering, DDoS Attacks, Buffer Overflows and Virus Creation.
5. Compare different types of hacking tools.

Course Outcomes

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE4.3.1	Describe the aspects of Ethical Hacking and implement foot printing.	3	3	2	2	2	-	-	-	-	-	-	3	2	L1, L2
R24CSOE4.3.2	Exploit the scanning and sniffing tools in order to generate signatures of transactions.	3	3	2	2	2	-	-	-	-	-	-	3	2	L2, L3
R24CSOE4.3.3	Identify worms and phishing attacks.	3	3	2	2	2	-	-	-	-	-	-	3	2	L3, L4
R24CSOE4.3.4	Analyse the steps involved in Hijacking of web servers and SQL injection	3	2	2	2	2	-	-	-	-	-	-	3	2	L4, L5
R24CSOE4.3.5	Classify wireless hacking techniques to secure cloud and outline the phases of penetration testing.	3	3	3	2	2	-	-	-	-	-	-	3	2	L5, L6

SYLLABUS**UNIT–I: Introduction to ethical Hacking and purpose of hacking****13Hours**

Types of attack sand attack vector types, Prevention from hackers, Phases of hacking. Foot printing and Reconnaissance. What is foot printing? Active and passive foot printing, purpose of foot printing, objectives of foot printing, foot printing threats, Types of foot printing, foot printing countermeasures Web Application (In) security: The Evolution of Web Applications, Common Web.

COs–CO1**Self-Learning Topics:** Sparse vs Dense Graphs**UNIT–II: Scanning networks, Enumeration and sniffing****15Hours**

Scanning networks: Network scanning and its types, objectives of network scanning, scanning live systems, scanning techniques-TCP Connect/ Full Open Scan, Types of Stealth scans, port scanning countermeasures, IDS evasion techniques, Banner grabbing and its tools, vulnerability scanning, proxy servers, anonymizers, IP spoofing and its counter measures. Enumeration and Sniffing:What

is Enumeration? Enumeration techniques, Enumeration types, Enumeration counter measures. What is sniffing? Packet sniffing: process of packet sniffing, sniffing methods, ARP spoofing and MAC flooding, active and passive sniffing, types of sniffing attacks, sniffing detection technique.

Self-Learning Topics: Time & Space complexity analysis in large graphs **COs–CO2**

UNIT-III: Trojans and other Attacks: **13Hours**

Introduction, Types of worms, Difference between DoS and DDoS attack, Water hole attack, brute force attack, Types of phishing attack, Eavesdropping, Man-in-the-middle attack, buffer overflow Attack, BOTNETS, Steganography: Overview, techniques, Social Engineering, Example of physical engineering, Methods of avoiding physical social engineering attacks and Hybrid social engineering attack. **COs-CO3**

Self-Learning Topics: Properties of trees in graphs

UNIT-IV: Hacking web servers, web applications and SQL Injection: **12Hours**

Session hijacking: What is session hijacking? How does session hijacking work? What Do Attackers Gain from Session Hijacking? How to prevent Session Hijacking, Web Servers, Web Server Attacks, Stages of Web Server attacks, Counter measures, web Application, Common Threats to web applications and how to avoid them.

SQL Injection: SQL injection, Types of SQL injections, Methods of detecting SQL injection vulnerabilities, and their prevention. **COs–CO4**

Self-Learning Topics: Complexity optimization using heaps

UNIT-V: **13Hours**

Wireless network Hacking, Cloud computing security, Cryptography, Pen testing: Types of wireless Architecture, wireless encryption techniques, Breaking WEP/WPA and defending WPA encryption, wireless Sniffing, wireless hacking techniques, Identification of the security vulnerabilities of a cloud computing system, cloud computing architecture. Types of cloud computing services, Models of cloud computing security. Threats and attacks. **COs–CO5**

Self-Learning Topics: GPS navigation systems, Routing protocols (Internet)

Board of Studies: Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No:03

Expert Talk (To be Delivered by SMEs from Industries) **COsPOs/PSOs**

1. “Modern Cyber Security Threats and Ethical Hacking Techniques”CO5PO1,PO2,PO3

Text Books:

1. Stuart McClure, Joel Scambray and GoergeKurtz,Hacking Exposed7: Network Security Secrets& Solutions, Tata Mc Graw Hill Publishers, 2010.
2. Bensmith, and Brian Komer, Microsoft Windows Security Resource Kit, Prentice Hall of India, 2010.

Reference Books:

1. RafayBaloch, “A Beginners Guide to Ethical Hacking”.
2. AllenHarper,ShonHarris,JonathanNess,ChrisEagle, “Gray Hat Hacking the Ethical Hackers Handbook”,3rdEdition,McGraw-HillOsborneMedia paperback(January27,2011)

Web References:

1. <https://www.coursera.org>

2. <https://www.edx.org>
3. <https://www.udemy.com>
4. <https://krebsonsecurity.com>
5. <https://www.cybrary.it>

Internal Assessment Pattern

CognitiveLevel	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

Chairperson
Board of Studies(CSE)

R24CSOE4.4**DATA MINING****3 0 0 3**

(Open Elective-IV)

Course Objectives:

1. To introduce fundamental concepts of data mining, data preprocessing and data exploration techniques.
2. To understand and apply classification methods for predictive data analysis.
3. To learn association rule mining techniques for discovering patterns in large datasets.
4. To develop knowledge of clustering techniques for grouping and analyzing data.
5. To explore anomaly detection methods for identifying unusual patterns and outliers in data.

Course Outcomes:

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

Course Code	Course Outcomes(COs)	Mapping with Pos and PSOs												Dok	
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
R24CSOE 4.4.1	Understand the fundamental concepts of data mining, data preprocessing, and data exploration techniques.	3	2	1	1	2	-	-	-	-	-	2	3	2	L1, L2
R24CSOE 4.4.2	Apply classification algorithms to build predictive models and evaluate their performance.	3	3	2	2	3	-	-	1	-	-	2	3	3	L2, L3,
R24CSOE 4.4.3	Analyze and generate association rules to discover patterns in large datasets.	3	2	2	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSOE 4.4.4	Implement clustering techniques to group data and interpret meaningful patterns.	3	2	2	2	3	-	-	1	-	-	2	3	3	L4, L5
R24CSOE 4.4.5	Identify and analyze anomalies using statistical and machine learning approaches.	3	3	2	3	3	1	1	-	-	-	2	3	3	L5, L6

SYLLABUS**UNITI: INTRODUCTION****15Hours**

Data Mining, Data Mining tasks, Types of Data, Data Quality, Data Processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Summary Statistics, Visualization, OLAP and Multidimensional Data Analysis.

CO's-CO1

Self-Learning Topics: Real-world applications of data mining, Introduction to big data and data warehousing, Data visualization using Tableau/Power BI, Data preprocessing using Python (NumPy, Pandas), Case study on recommendation systems

UNITII: CLASSIFICATION**15Hours**

Basic Concepts, Decision Trees, Model Evaluation, General approach for solving classification problems, Decision Tree Induction, Overfitting issues, Performance evaluation, Rule-based

classifier, Nearest Neighbor classifier, Bayesian Classifier.

CO's-CO2

Self-Learning Topics: Implementation of classifiers using Scikit-learn, Comparison of classifiers (Decision Tree, KNN, Naive Bayes), Advanced methods (Random Forest, SVM), Performance metrics (precision, recall, F1-score), Case study on spam email classification

UNITIII: ASSOCIATION ANALYSIS

13Hours

Problem Definition, Frequent Item set Generation, Rule Generation, Confidence-based pruning, Apriori Algorithm, Compact Representation, FP-Growth Algorithm.

CO's-CO3

Self-Learning Topics: Market basket analysis, Implementation of Apriori and FP-Growth in Python, Measures like lift and conviction, Applications in retail analytics, Case study on product recommendation systems

UNITIV: CLUSTERING ANALYSIS

13 Hours

Overview, Basics and Importance of Clustering, Clustering Techniques, Types of Clusters, K-means Algorithm, Issues in K-means, Bisecting K-means.

CO's-CO4

Self-Learning Topics: Implementation of K-means in Python, Hierarchical clustering methods, DBSCAN algorithm, Cluster validation techniques (Silhouette score), Case study on customer segmentation

UNITV: ANOMALY DETECTION

12Hours

Preliminaries, Causes of Anomalies, Detection Approaches, Statistical Methods, Outliers in univariate and multivariate distributions, Mixture models, Proximity-based, Density-based, and Clustering-based methods.

CO's-CO5

Self-Learning Topics: Anomaly detection using Isolation Forest and LOF, Fraud detection in banking, Network intrusion detection, Time-series anomaly detection, Case study on credit card fraud detection

Board of Studies :Computer Science and Engineering

Approved in BoS No: 03

Approved in ACM No: 03

Text Books:

1. Data Mining: Concepts and Techniques–JiaweiHan,MichelineKamber,Jian Pei
2. Data Mining Techniques –ArunK.Pujari(UniversitiesPress,India).
3. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach, Vipin Kumar (PearsonIndia)

Reference Books:

1. Data Mining and Knowledge Discovery Handbook–OdedMaimon, Lior Rokach
2. Principles of Data Mining–D.J. Hand, HeikkiMannila, Padhraic Smyth
3. Data Mining :*KnowItAll*–SoumenChakrabartietal.

Web References:

1. [NPTELDataMiningCourse](#)–FreecoursebyIITcoveringclassification,clustering, association rules, and preprocessing.
2. [SWAYAM Data Mining Course \(IIT Kharagpur\)](#)– 8-week Indian government course with full syllabus and certification.
3. DataMining Tutorial–GeeksforGeeks

Internal Assessment Pattern

Cognitive Level	InternalAssessment#1(%)	InternalAssessment#2(%)
L1	35	--

AIETM|R24|CSE|R24CSOE4.4|Data Mining

L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total(%)	100	100

.

**Chairperson
Board of Studies(CSE)**

Biometrics**(Open Elective IV)**

3 0 0 3 (Common to CSE)

Course Objectives:

1. To understand the technologies of fingerprint, iris, face and speech recognition
2. To understand the general principles of design of biometric systems and the underlying trade-offs.
3. To recognize personal privacy and security implications of biometrics based identification technology.

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PS01	PS02	PS03	
R24CSPE4.5.1	Understand basic concepts of biometric technology	3	2	-	-	-	2	3	-	L1,L2,L4
R24CSPE4.5.2	Analyze fingerprint technology	2	3	-	3	-	2	3	-	L2,L3
R24CSPE4.5.3	Analyze face recognition systems	2	2	3	2	3	2	3	-	L3,L4
R24CSPE4.5.4	Understand voice based biometric recognition	1	2	3	2	3	2	3	-	L3,L4
R24CSPE4.5.5	Understand Multi-biometric systems	1	2	2	2	3	2	3	-	L4,L5

SYLLABUS**UNIT-I: INTRODUCTION TO BIOMETRICS**

Introduction and background – biometric technologies – passive biometrics – active biometrics - Biometrics Vs traditional techniques – Benefits of biometrics - Operation of a biometric system– Key biometric processes: verification, identification and biometric matching – Performance measures in biometric systems: FAR, FRR, FTE rate, FTA rate and rate- Need for strong authentication – Protecting privacy and biometrics and policy –Biometric applications. **COs–CO1**

UNIT-II: FINGERPRINT IDENTIFICATION TECHNOLOGY

Fingerprint Patterns, Fingerprint Features, Fingerprint Image, width between two ridges -Fingerprint Image Processing - Minutiae Determination - Fingerprint Matching: Fingerprint Classification, Matching policies. **COs–CO2**

UNIT-III: FACE RECOGNITION

AIETM | R24| CSE| R24CSPE4.5| **Biometrics**

Introduction, components, Facial Scan Technologies, Face Detection, Face Recognition, Representation and Classification, Kernel- based Methods and 3D Models, Learning the FaceSpace, Facial Scan Strengths and Weaknesses, Methods for assessing progress in Face Recognition. **COs–CO3**

UNIT-IV: VOICE SCAN

Introduction, Components, Features and Models, Addition Method for managing Variability, Measuring Performance, Alternative Approaches, Voice Scan Strengths and Weaknesses, NIST Speaker Recognition Evaluation Program, Biometric System Integration. **COs–CO4**

UNIT-V: FUSION IN BIOMETRICS

Introduction to Multi biometric - Advantages of multimodal system, Information Fusion in Biometrics - Issues in Designing a Multi biometric System - Sources of Multiple Evidence -Levels of Fusion in Biometrics – Sensor level, Feature level, Rank level, Decision level fusion - Score level Fusion. Examples –gait based biometric systems. **COs–CO5**

Text Books:

1. James Wayman, Anil Jain, Davide Maltoni, Dario Maio, —Biometric Systems, Technology Design and Performance Evaluation, Springer.
2. David D. Zhang, —Automated Biometrics: Technologies and Systems, Kluwer Academic Publishers, New Delhi.
3. Arun A. Ross, Karthik Nandakumar, A.K.Jain, —Handbook of Multibiometrics, Springer, New Delhi.

References Books:

1. Paul Reid, —Biometrics for Network Security, Pearson Education, 2004.
2. Nalini K Ratha, Ruud Bolle, —Automatic fingerprint Recognition System, Springer.
3. L C Jain, I Hayashi, S B Lee, U Halici, —Intelligent Biometric Techniques in Fingerprint and Face Recognition, CRC Press, 1999.
4. John Chirillo, Scott Blaul, —Implementing Biometric Security, John Wiley, 2003.
5. S.Y. Kung, S.H. Lin, M.W.Mak, —Biometric Authentication: A Machine Learning Approach, Prentice Hall, 2005

Web References:

1. NPTEL / SWAYAM: Biometrics (IIT Kanpur), Link: nptel.ac.in/courses/106104119
2. Virtual Cyber Labs: Biometric Authentication Frameworks, Link: virtualcyberlabs.com/biometric-authentication-in-cybersecurity/
3. Springer Practical Guide to Biometrics, Link: springer.com/gp/book/9781852335960

Chairperson
Board of Studies (CSE)