AR25
PROGRAM STRUCTURE
and
FIRST AND SECOND YEAR SYLLABUS

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

For CBCS BASED B. TECH – FOUR YEAR PROGRAM
(Applicable for the batches admitted from AY 2025-26)

Geethanjali College of Engineering and Technology

(Autonomous)

(Approved by AICTE, Permanently Affiliated to JNTUH, Accredited by NBA and Accredited by NAAC with A+ Grade)

Sy. No: 33 & 34, Cheeryal (V), Keesara (M), Medchal District, Telangana – 501301

GEETHANJALI COLLEGE OF ENGINEERING AND TECHNOLOGY
(UGC Autonomous)
Cheeryal (V), Keesara (M), Medchal Dist., - 501 301

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE INSTITUTE

To be an epicenter, promoting scholarly activities, fostering innovation, research, and entrepreneurship, leading sustainable societal development.

MISSION OF THE INSTITUTE

1. To solve complex societal problems, inculcating critical thinking and problem-solving skills.
2. To inculcate creativity and innovation, developing a culture of research and entrepreneurship.
3. To preserve and promote cultural heritage, humanistic and spiritual values, promoting peace and harmony in society.

VISION OF THE DEPARTMENT

To produce globally competent and socially responsible computer science engineers contributing to the advancement of engineering and technology which involves creativity and innovation by providing excellent learning environment with world class facilities.

MISSION OF THE DEPARTMENT

1. To be a Centre of excellence in instruction, innovation in research and scholarship, and service to the stake holders, the profession, and the public.
2. To prepare graduates to enter a rapidly changing field as a competent computer science engineer.
3. To prepare graduate capable in all phases of software development, possess a firm understanding of hardware technologies, have the strong mathematical background necessary for scientific computing, and be sufficiently well versed in general theory to allow growth within the discipline as it advances.
4. To prepare graduates to assume leadership roles by possessing good communication skills, the ability to work effectively as team members, and an appreciation for their social and ethical responsibility in a global setting.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs):

Program Educational Objectives (PEOs) are broad statements that describe what graduates are expected to attain within a few years of graduation. The PEOs for Computer Science and Engineering graduates are:

PEO-I. To provide graduates with a good foundation in mathematics, sciences and engineering fundamentals required to solve engineering problems that will facilitate them to find employment in industry and / or to pursue postgraduate studies with an appreciation for lifelong learning.

PEO-II. To provide graduates with analytical and problem solving skills to design algorithms, other hardware / software systems, and inculcate professional ethics, inter-personal skills to work in a multi-cultural team.

PEO-III. To facilitate graduates get familiarized with state of the art software / hardware tools, imbibing creativity and Innovation that would enable them to develop cutting-edge technologies of multi-disciplinary nature for societal development.

Knowledge and Attitude Profile (WK)

WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

WK2: Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.

WK3: A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.

WK4: Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.

WK5: Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.

WK6: Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.

WK7: Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.

WK8: Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.

WK9: Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Program Outcomes (POs)

PO1. Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2. Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3. Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4. Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5. Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6. The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7. Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8. Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9. Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.

PO10. Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11. Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES (PSOs):

PSO1: Demonstrate competency in Programming and problem solving skills and apply these skills in solving real world problems.

PSO2: Select appropriate programming languages, Data structures and algorithms in combination with modern technologies and tools, apply them in developing creative and innovative solutions.

PSO3: Demonstrate adequate knowledge in emerging technologies.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B.TECH. COMPUTER SCIENCE AND ENGINEERING
AR25 STRUCTURE

S.No.	Category	Credits as per GCET AR25	Credits as per AICTE Model Curriculum(2022)
1	**Humanities and Social Sciences including Management Courses**	**13**	**16**
2	**Basic Sciences Courses**	**19**	**23**
3	**Engineering Sciences Courses** including workshop, drawing, basics of electrical/ mechanical/ computer etc.	**21**	**29**
4	**Program Core Courses**	**62**	**59**
5	**Program Elective Courses**: Subjects relevant to chosen specialization/branch	**18**	**12**
6	**Open Elective Courses**: Electives from other technical and/or emerging subjects	**06**	**09**
7	**Project work, Seminar and Internship** in industry or else where	**22**	**15**
8	**Value Added Courses**	**3**	
	Total	**164**	**163**

COURSE CODE AND DEFINITION

S.No.	Category Abbreviation	Description
1	HSMC	Humanities and Social Sciences including Management Course
2	BSC	Basic Science Course
3	ESC	Engineering Science Course
4	PCC	Program Core Course
5	PEC	Professional Elective Course
6	OEC	Open Elective Course
7	PROJ	Project, Seminar and Internship
8	VA	Value Added Course

DEFINITION OF CREDIT

S.No.	Abbreviation	Credits	Description
1	L	1	1 Hour Lecture (L) per week
2	T	1	1 Hour Tutorial (T) per week
3	P/D	0.5 1	1 Hour Practical (P)/ Drawing (D) per week 2 Hours Practical (P)/ Drawing (D) per week

SCHEME OF INSTRUCTION AND EXAMINATION

B.TECH. COMPUTER SCIENCE AND ENGINEERING

Academic Year 2025-26

PROGRAM STRUCTURE

FIRST YEAR SEMESTER-I

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25MA11001	Matrices and Calculus	BSC	3	1	0	40	60	100	4
2	25CH11001	Engineering Chemistry	BSC	3	0	0	40	60	100	3
3	25EN11001	English for skill Enhancement	HSMC	3	0	0	40	60	100	3
4	25PH11002	Semiconductor Devices and Circuits	ESC	3	0	0	40	60	100	3
5	25CS11001	Programming for Problem Solving	ESC	2	0	0	40	60	100	2
6	25CH11L01	Engineering Chemistry Lab	BSC	0	0	2	40	60	100	1
7	25CS11L01	Programming for Problem Solving Lab	ESC	0	0	2	40	60	100	1
8	25EN11L01	English Language and Communication Skills Lab	HSMC	0	0	2	40	60	100	1
9	25ME11L01	Engineering Workshop	ESC	0	0	2	40	60	100	1
10		Induction Program								
Total				**14**	**1**	**8**	**360**	**540**	**900**	**19**
Total Periods per Week				**23**						

B.Tech CSE **AR25**

FIRST YEAR SEMESTER-II

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25MA12001	Ordinary Differential Equations and Vector Calculus	BSC	3	-	-	40	60	100	3
2	25PH12001	Advanced Engineering Physics	BSC	3	-	-	40	60	100	3
3	25ME12001	Engineering Drawing and Computer Aided Drafting	ESC	2	-	2	40	60	100	3
4	25EE12001	Basic Electrical Engineering	ESC	3	-	-	40	60	100	3
5	25CS12001	Data Structures	ESC	3	-	-	40	60	100	3
6	25CS12003	Discrete Mathematics	PCC	3	-	-	40	60	100	3
7	25PH12L01	Advanced Engineering Physics Lab	BSC	-	-	2	40	60	100	1
8	25CS12L01	Data Structures Lab	ESC	-	-	2	40	60	100	1
9	25EE12L01	Basic Electrical Engineering Lab	ESC	-	-	2	40	60	100	1
			Total	**17**	**0**	**8**	**360**	**540**	**900**	**21**
			Total Periods per Week	**25**						

B.Tech CSE **AR25**

SECOND YEAR SEMESTER-I

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25CS21001	Advanced Data Structures	PCC	3	0	0	40	60	100	3
2	25CS21002	Object Oriented Programming through java	PCC	3	0	0	40	60	100	3
3	25CS21003	Data Base Management Systems	PCC	3	0	0	40	60	100	3
4	25EC21002	Digital Logic Design	ESC	3	0	0	40	60	100	3
5	25MS21002	Business Economics and Financial Analysis	HSMC	3	0	0	40	60	100	3
6	25CS21L01	Advanced Data Structures Lab	PCC	0	0	2	40	60	100	1
7	25CS21L02	Object Oriented Programming through java Lab	PCC	0	0	2	40	60	100	1
8	25CS21L03	Data Base Management Systems Lab	PCC	0	0	2	40	60	100	1
9	25CS21SD1	Skill Development Course 1-(Data Visualization using Python)	SDC	0	0	2	40	60	100	1
10	25CE21VA1	Environmental Science	VAC	1	0	0	40	60	100	1
Total				**16**	**0**	**8**	**400**	**600**	**1000**	**20**
Total Periods per Week				**24**						

SECOND YEAR SEMESTER-II

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25CS22001	Algorithm Design and Analysis	PCC	3	0	0	40	60	100	3
2	25CS22002	Computer Organization and Assembly Language Programming	PCC	3	0	0	40	60	100	3
3	25CS22003	Operating Systems	PCC	3	0	0	40	60	100	3
4	25CS22004	Web Technologies	PCC	2	0	0	40	60	100	2
5	25CS22005	Automata Theory and Compiler Design	PCC	3	0	0	40	60	100	3
6	25CS22L01	Algorithm Design and Analysis and Compiler Design Lab	PCC	0	0	2	40	60	100	1
7	25CS22L02	Operating Systems and Assembly Level Programming Lab	PCC	0	0	2	40	60	100	1
8	25CS22L03	Web Technologies Lab	PCC	0	0	2	40	60	100	1
9	25CS22SD2	Skill Development Course 2–Node Js /ReactJS	SDC	0	0	2	40	60	100	1
10	25MS22001	**Innovation and Entrepreneurship**	HSMC	2	0	0	40	60	100	2
Total				**16**	**0**	**8**	**400**	**600**	**1000**	**20**
Total Periods per Week				**24**						

*__Note:__ Students who wish to exit after II Year II Semester has to register for this optional course and acquire the credits allotted by doing 6 weeks Work-based Vocational Course/ Internship or Apprenticeship.

B.Tech CSE **AR25**

THIRD YEAR SEMESTER-I

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25MA31001	Statistics for Machine Learning	BSC	3	-	-	40	60	100	3
2	25CS31001	Artificial Intelligence	PCC	3	-	-	40	60	100	3
3	25CS31002	Computer Networks	PCC	3	-	-	40	60	100	3
4	**Open Elective I**		OEC	2	-	-	40	60	100	2
	25CE31101	Building Science and Technology								
	25AI31105	Agentic AI								
	25CY31107	Ethical Hacking Fundamentals								
	25DS31109	Data Engineering								
	25EE31110	Fundamentals of Electric Vehicles								
	25EC31111	Principles of Communication								
	25ME31112	Industrial Robotics								
	25MS31113	Intellectual Property Rights								
	25MA31114	Logical Reasoning-I								
5	**Professional Elective I**		PEC	3	-	-	40	60	100	3
	25CS31004	Computer Graphics								
	25CS31005	Introduction to Data Science								
	25 CS31006	Software Testing Methodologies								
	25 CS31007	Data Mining								
	25 CS31008	Web Programming								
	25 CS31009	Distributed Systems								
6	25CS31L01	Artificial Intelligence with Python Lab	PCC	-	-	2	40	60	100	1
7	25CS31L02	Computer Networks Lab	PCC	-	-	2	40	60	100	1
8	25MA31L01	Statistics for Machine Learning Lab	BSC	-	-	2	40	60	100	1
9	25CS31003	Field-Based Project/Internship	PROJ	-	-	4	-	100	100	2
10	25CS31SD3	Skill Development Course-3(UI Design/ Flutter/DevOps/Android Studio)	SDC	-	-	2	40	60	100	1
11	25EN31VA2	Indian Knowledge System	VAC	1	-	-	40	60	100	1
Total				**15**	**0**	**12**	**400**	**700**	**1100**	**21**
Total Periods per Week				**27**						

THIRD YEAR SEMESTER-II

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25CS32001	Software Engineering	PCC	2	-	-	40	60	100	2
2	25CS32002	Machine Learning	PCC	3	-	-	40	60	100	3
3	25CS32003	Cryptography and Network Security	PCC	3	-	-	40	60	100	3
4	**Professional Elective – II**		PEC	3	-	-	40	60	100	3
	25CS32004	Image Processing								
	25CS32005	Block Chain Technology								
	25CS32006	Software Project Management								
	25CS32007	Mining Massive Datasets								
	25CS32008	Full Stack Development								
	25CS32009	Generative AI								
5	**Open Elective II**		OEC	2	-	-	40	60	100	2
	25CE32201	Building Services Engineering								
	25AI32205	Fundamentals of Fuzzy Logic								
	25CY32206	Social Media Security								
	25DS32208	MERN Stack								
	25EE32210	Digital Energy								
	25EC32211	Electronics for Health Care								
	25ME32212	Non-Conventional Sources of Energy								
	25MS32213	Supply Chain Management								
	25MA32214	Logical Reasoning-II								
6	25CS32L01	Software Engineering Lab	PCC	-	-	2	40	60	100	1
7	25CS32L02	Machine Learning Lab	PCC	-	-	2	40	60	100	1
8	25CS32L03	Cryptography and Network Security Lab	PCC	-	-	2	40	60	100	1
9	25EN32L01	English for Employability Skills Laboratory	HSMC	-	-	2	40	60	100	1
10	25CS32SD4	Skill Development Course4 –Prompt Engineering	SDC	-	-	2	40	60	-	1
11	25MS32VA3	Gender Sensitization*/Human Values and Professional Ethics*	VAC	1	-	-	40	60	100	1
			Total	**14**	**0**	**10**	**440**	**660**	**1100**	**19**
			Total Periods per Week	**24**						

***Note: For the courses Gender Sensitization and Human Values and Professional Ethics -** one hour of instruction will be conducted on alternate weeks. For example, if a one-hour class for Gender Sensitization is conducted this week, then a one-hour class for

Human Values and Professional Ethics will be conducted in the following week.

FOURTH YEAR SEMESTER-I

S.No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/D	CIE	SEE	Tot	
1	25CS41001	Natural Language Processing	PCC	3	-	-	40	60	100	3
2	25CS41002	Deep Learning	PCC	3	-	-	40	60	100	3
3	25CS41003	Cyber Security and Digital Forensics	PCC	2	-	-	40	60	100	2
4	25MS41003	Fundamentals of Management For Engineers	HSMC	3	-	-	40	60	100	3
5	**Professional Elective – III**		PEC	3	-	-	40	60	100	3
	25CS41005	Computer Vision								
	25CS41006	Scripting Languages								
	25CS41007	Vulnerability and Penetration Testing								
	25CS41008	Data Stream Mining								
	25CS41009	Cloud Computing								
	25CS41010	Information Retrieval Systems								
6	**Professional Elective – IV**		PEC	3	-	-	40	60	100	3
	25CS41011	Augmented Reality and Virtual Reality								
	25CS41012	Agile Methodology								
	25CS41013	Big Data Technologies								
	25CS41014	Quantum Computing								
	25CS41015	Robotic Process Automation								
	25CS41016	Cyber Forensics								
7	25CS41L01	Natural Language Processing Lab	PCC	-	-	2	40	60	100	1
8	25CS41L02	Deep Learning Lab	PCC	-	-	2	40	60	100	1
9	25CS41L03	Cyber Security and Digital Forensics Lab	PCC	-	-	2	40	60	100	1
10	25CS41004	Industry Oriented Mini Project/ Internship	PROJ	-	-	4	-	100	100	2
Total				**17**	**0**	**10**	**360**	**640**	**1000**	**22**
Total Periods per Week				**27**						

B.Tech CSE **AR25**

FOURTH YEAR SEMESTER-II

S. No.	Course Code	Course Title	Category	Number of Periods per Week			Scheme of Examination with Maximum Marks			No of Credits
				L	T	P/ D	CIE	SEE	Tot	
1	**Professional Elective V**		PEC	3	-	-	40	60	100	3
	25CS42002	Social Media Mining								
	25CS42003	Nature Inspired Computing								
	25CS42004	Internet of Things								
	25CS42005	Game Theory								
	25CS42006	Mobile Application Development								
	25CS42007	Human Computer Interaction								
2	**Professional Elective VI**		PEC	3		-	40	60	00	3
	25CS42008	High Performance Computing								
	25CS42009	Edge Computing								
	25CS42010	Graph Theory								
	25CS42011	Adhoc and Sensor Networks								
	25CS42012	Sustainable Engineering								
	25CS42013	Distributed Databases								
3	**Open Elective III**		OEC	2	-	-	40	60	100	2
	25CE42301	Disaster Management								
	25AI42304	Chatbots								
	25CY42306	Data Privacy								
	25DS42308	Android Application Development								
	25EE42310	Sustainable Energy								
	25EC42311	Introduction to Sensors and Instrumentation								
	25ME42312	Digital Manufacturing								
	25MS42313	Project Management and Finance								
	25MA42314	Mathematics in India from Vedic Period to Modern Times								
4	25CS42001	Project Work	PROJ	-	-	20	40	60	100	14
			Total	**8**	**0**	**20**	**160**	**240**	**400**	**22**
			Total Periods per Week	**28**						

25MA11001 – MATRICES AND CALCULUS

B. Tech.CSE – I Year, I Sem.

Prerequisite(s): None

L	T	P/D	C
3	1	-/-	4

Course Objectives: Develop ability to

1. Understand various types of matrices, properties and rank of the matrix to find the solution for system of equations, if it exists.
2. Apply the knowledge of eigen values and eigenvectors of a matrix from quadratic form into a canonical form through linear and orthogonal transformations.
3. Familiarize students with the statements, geometrical interpretations, and applications of Mean Value Theorems such as Rolle's theorem, Lagrange's Mean Value Theorem, and Cauchy's Mean Value Theorem.
4. Compute partial derivatives, composite functions of several variables and apply the methods of differential calculus to optimize multivariable functions
5. Evaluate definite integrals to calculate surface and volume of revolutions of curves, multiple integrals and apply the same to solve engineering problems.

Course outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply elementary transformations to solve a system of linear equations and reduce the quadratic form to the canonical form using linear and / or orthogonal transformation.	PO1,PO2, PO3 and PSO1	I & II	1,2,3,4	SDG4 SDG9
CO2	Apply Mean Value Theorems to analyze the behaviour of functions, interpret their geometrical meaning, and solve related problems in mathematical and engineering contexts.	PO1,PO2, PO3 and PSO1	III	1,2,3,4	
CO3	Apply the concept of partial differentiation to solve constrained optimization problems without graphical representation	PO1,PO2, PO3 and PSO 1	IV	1,2,3,4	
CO4	Apply the definite / multiple integrals to compute areas and volumes of any region / solids	PO1,PO2, PO3 and PSO1	V	1,2,3,4	

UNIT-I: Matrices 8 L

Rank of a matrix by Echelon form and Normal form, Inverse of Non-singular matrices by Gauss-Jordan method.

System of linear equations: Solving System of Homogeneous and Non-Homogeneous equations, Gauss Seidel Iteration Method.

UNIT-II: Eigenvalues and Eigenvectors **10 L**

Linear Transformation and Orthogonal Transformation: Eigenvalues and eigenvectors and their properties, 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 form by Orthogonal Transformation.

UNIT III: Single Variable Calculus: **10 L**

Limits and Continuity of Functions and their properties, Mean Value Theorems – Rolle's Theorem, Lagrange's Mean Value Theorem with their geometrical interpretation and applications, Cauchy's mean Value Theorem, Taylor's Series (All the theorems without proof)

UNIT IV: Multivariable Calculus (Partial Differentiation and applications) **10 L**

Definitions of Limit and Continuity, Partial Differentiation, Euler's Theorem, Total derivative, Jacobian, Functional dependence and independence.
Applications: Maxima and Minima of functions of two variables and three variables using method of Lagrange multiplier.
Improper Integrals: Beta and Gamma Functions and their applications without proofs.

UNIT V: Multivariable Calculus (Integration) **10 L**

Evaluation of Double Integrals (Cartesian and polar coordinates), Change of order of integration (only Cartesian form), Change of variables for double integrals (Cartesian to polar),
Evaluation of Triple Integrals: Change of Variables for triple integrals (Cartesian to Spherical and Cylindrical polar coordinates).
Applications: Areas by double integrals and volumes by double integrals and triple integrals.

TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 44th Edition, 2017.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 5th Edition, 2016.

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 10th Edition, 2011.
2. G.B. Thomos and R.L. Finney, Calculus and Analytic Geometry, 9th Edition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A Text book of Engineering Mathematics, Laxmi Publications, 10th Edition, 2015.
4. H.K. Das and Er. RajnishVerma, Higher Engineering Mathematics, S. Chand and Company Ltd, New Delhi.

25PH11002 -SEMICONDUCTOR DEVICES AND CIRCUITS (CSE)

B. Tech.CSE -I Year I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Understand the electrical characteristics, models of semiconductor diodes, applications of diodes as rectifiers and clipping circuits.
2. Understand the operation, configurations and characteristics of Bipolar Junction Transistor.
3. Understand the necessity of transistor biasing and types of biasing for faithful amplification.
4. Understand transistor amplifier circuits using h-parameter equivalent circuit.
5. Understand the working, characteristics and comparison of JFET, MOSFET, FinFET and CNTFET

Course Outcomes: At the end of the course, students would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Analyze the electrical characteristics and models of semiconductor diodes and apply them in rectifier and clipping circuits.	PO1,PO2, PO3,PO4, PO5,PO6	I	1,2,3	SDG4
CO2	Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.	PO1,PO2, PO3,PO4,PO5	II	1,2,3	
CO3	Design appropriate biasing networks for BJTs and determine the operating point for amplifier applications.	PO1,PO2, PO3,PO4,PO5	III	1,2,3	
CO4	Analyzetransistoramplifiercircuitsusingh-parametermodelsandassessperformancefor various configurations.	PO1,PO2, PO3,PO4,PO5	IV	1,2,3	
CO5	Analyze the structure, working, and characteristics of JFETs, MOSFETs, and advanced devices like FinFETs and CNTFETs, and compare modern device technologies.	PO1,PO2,PO3, PO4,PO5,PO6	V	1,2,3	

UNIT - I:

Diode Characteristics and Applications: PN junction diode – I-V characteristics, Diode resistance and capacitance, Diode models (Ideal, Simplified, Piecewise Linear), Rectifiers – Half-wave, Full-wave (Center-tap and bridge), Capacitor filter for rectifiers, Clippers and clampers, Zener diode – I-V characteristics and voltage regulation.

UNIT - II:

Bipolar Junction Transistor (BJT): Structure and working principle of BJT, Current components and transistor action, Configurations: Common Base (CB), Common Emitter

(CE), Common Collector (CC), Input and output characteristics, Determination of h-parameters from transistor characteristics.

UNIT - III:

BJT Biasing: Need for biasing and stabilization, Load line and operating point, Biasing techniques: Fixed bias, Collector-to-base bias, Voltage divider bias, Stability factors and thermal runaway

UNIT - IV:

Transistor Amplifiers: Transistor as a small-signal amplifier, h-parameter equivalent circuit, CE, CB, CC amplifier analysis using h-parameters, Approximate CE model – with and without emitter bypass capacitor.

UNIT - V:

Special Purpose Diodes: Principle of Operation of – SCR, Tunnel Diode, Varactor Diode, Photo Diode, Solar Cell, LED and Schottky Diode

Field Effect Transistors and Advanced Devices: JFET: Structure, operation, and characteristics, MOSFET: Enhancement and Depletion modes – Structure, operation, and characteristics, Advanced Devices: FinFETs - 3D structure, Scaling advantages, CNTFETs - Structure, ballistic transport, fabrication, Comparison: CMOS vs. FinFET vs. CNTFET.

TEXTBOOKS:

1. Millman, Jacob, and Christos C. Halkias. *Electronic Devices and Circuits*. Tata McGraw-Hill, 1991.
2. Boylestad, Robert L., and Louis Nashelsky. *Electronic Devices and Circuit Theory*. Pearson, 11th ed., 2013.
3. Sedra, Adel S., and Kenneth C. Smith. *Microelectronic Circuits*. Oxford University Press, 7th ed., 2014.

REFERENCEBOOKS:

1. Bell,DavidA.*ElectronicDevicesandCircuits*.OxfordUniversityPress,5thed.,2008.
2. Neamen,DonaldA.*ElectronicCircuitAnalysisandDesign*.McGraw-Hill,2nded.,2001.
3. Salivahanan,S.,andN.SureshKumar.*ElectronicDevicesandCircuits*.McGraw-Hill Education, 4th ed., 2017.
4. Razavi,Behzad.*FundamentalsofMicroelectronics*.Wiley,2nded.,2013.
5. Taur, Yuan, and Tak H. Ning. *Fundamentals of Modern VLSI Devices*. Cambridge University Press, 2nd ed., 2009.

25EN11001–ENGLISH FOR SKILL ENHANCEMENT

B. Tech. CSE - I Year I Sem

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Improve vocabulary.
2. Use appropriate sentence structures in oral and written communication.
3. Strengthen reading comprehension and independent study skills.
4. Write paragraphs, essays, précis and draft letters.
5. Write technical reports

Course Outcomes: At the end of the course, students would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Infer and use appropriate vocabulary in oral and written communication.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	SDG: 4
CO2	Apply the rules of functional grammar and sentence structures in communication.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	
CO3	Comprehend any given text and respond precisely.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	
CO4	Construct meaningful and explicit sentences in written form befitting the context.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	

UNIT –I

Theme: **Perspectives Lesson on 'The Generation Gap' by Benjamin M. Spock from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan Pvt. Ltd.**

Vocabulary: The Concept of Word Formation-The Use of Prefixes and Suffixes-Words Often Misspelt - Synonyms and Antonyms

Grammar: Identifying Common Errors in Writing with Reference to Parts of Speech particularly Articles and Prepositions—Degrees of Comparison

Reading: Reading and Its Importance-Sub Skills of Reading–Skimming and Scanning.

Writing: Sentence Structures and Types -Use of Phrases and Clauses in Sentences- Importance of Proper Punctuation-Techniques for Writing Precisely–Nature and Style of Formal Writing.

UNIT–II

Theme: Digital Transformation Lesson on '*Emerging Technologies*' from the prescribed text book titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.

Vocabulary: Homophones, Homonyms and Homographs

Grammar: Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement.

Reading: Reading Strategies-Guessing Meaning from Context – Identifying Main Ideas – Exercises for Practice

Writing: Paragraph Writing —Types, Structures and Features of a Paragraph - Creating Coherence —Linkers and Connectives - Organizing Principles in a Paragraph — Defining- Describing People, Objects, Places and Events — Classifying- Providing Examples or Evidence - Essay Writing - Writing Introduction and Conclusion.

UNIT–III

Theme: Attitude and Gratitude Poems on '*Leisure*' by William Henry Davies and '*Be Thankful*'-Unknown Author from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.

Vocabulary: Words Often Confused-Words from Foreign Languages and their Use in English.

Grammar: Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses.

Reading: Sub-Skills of Reading – Identifying Topic Sentence and Providing Supporting Ideas - Exercises for Practice.

Writing: Format of a Formal Letter-Writing Formal Letters E.g.., Letter of Complaint, Letter of Requisition, Job Application with CV/Resume–Difference between Writing a Letter and an Email - Email Etiquette.

UNIT–IV

Theme: Entrepreneurship Lesson on '*Why a Start-Up Needs to Find its Customers First*' by Pranav Jain from the prescribed text book titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.

Vocabulary: StandardAbbreviationsinEnglish– InferringMeaningsofWordsthroughContext– Phrasal Verbs—Idioms.

Grammar: Redundancies and Clichés in Written Communication – Converting Passive to Active Voice and Vice-Versa.

Reading: Prompt Engineering Techniques–Comprehending and Generating Appropriate Prompts - Exercises for Practice

Writing: Writing Practices-Note Making-Précis Writing.

UNIT–V

Theme: Integrity and Professionalism Lesson on '*Professional Ethics*' from the prescribed text book titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.

Vocabulary: Technical Vocabulary and their Usage–One Word Substitutes–Collocations.

Grammar: Direct and Indirect Speech-Common Errors in English(Covering all the other aspects of grammar which were not covered in the previous units)

Reading: Survey, Question, Read, Recite and Review (SQ3R Method) – Inferring the Meaning and Evaluating a Text-Exercises for Practice

Writing: ***Report Writing-Technical Reports-Introduction–Characteristics of a Report – Categories of Reports Formats- Structure of Reports (Manuscript Format)-Types of Reports-Writing a Technical Report.***

TEXTBOOK:

1.Board of Editors.*2025.English for the Young in the Digital World.* Orient BlackSwan Pvt.Ltd.

REFERENCEBOOK(S):

1. Swan, Michael.(2016).*Practical English Usage.* Oxford University Press. New Edition.
2. *Karal,Rajeevan.2023.English Grammar Just for You.* Oxford University Press. New Delhi
3. 2024.*Empowering with Language: Communicative English for Undergraduates.* Cengage Learning India Pvt. Ltd. New Delhi
4. Sanjay Kumar &PushpaLata. 2022. *Communication Skills – A Workbook.* Oxford University Press. New Delhi
5. Wood,F.T.(2007).*Remedial English Grammar.* Macmillan.
6. Vishwamohan, Aysha.(2013).*English for Technical Communication for Engineering Students.* McGraw-Hill Education India Pvt.Ltd.

25CH11001– ENGINEERING CHEMISTRY

B. Tech. CSE - I Year I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None.

Course Objectives: Develop an ability to

1. Acquire knowledge of various water treatment methods and their industrial significance in resolving the problem of water hardness.
2. Understand fundamental principles of electrochemistry and corrosion with a perspective of their industrial applications.
3. Impart fundamental knowledge of various energy sources and their practical applications in engineering.
4. Understand the various aspects of polymers, including conducting and biodegradable polymers, and their applications in diverse fields.
5. Acquire knowledge of materials such as cement, lubricants, and biosensors, as well as spectroscopic techniques applicable in engineering, industrial and biomedical fields.

Course Outcomes: At the end of the course, students would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Predict problems associated with hardness of water and identify appropriate method to treat hardness.	PO1, PO2	I	1,2,3	SDG 3,6,7,8, 12
CO2	Analyze different electrodes and corrosion control methods for interpreting their applications in various sectors.	PO1, PO2	II	1,2,3,4	
CO3	Comprehend the usage of batteries, fuel cells and various energy sources, enhancing their potential as future engineers and entrepreneurs.	PO1, PO2	III	1,2,3	
CO4	Categorize polymers and their applications in different fields.	PO1, PO2	IV	1,2,3	
CO5	Apply knowledge of engineering materials and principles of spectroscopic techniques to support industrial and biomedical applications	PO1, PO2	V	1,2,3	

UNIT-I: Water and its treatment

Introduction-Hardness, types, degree of hardness and units. Estimation of hardness of water by complexometric method - Numerical problems. Potable water and its specifications (WHO) – Steps involved in the treatment of potable water - Disinfection of potable water

bychlorination and break-point chlorination. Defluoridation - Nalgonda technique.

Boiler troubles: Scales, Sludges and Caustic embrittlement. Internal treatment of boiler feed water - Calgon conditioning and Phosphate conditioning. External treatment methods - Softening of water by ion- exchange processes. Desalination of brackish water – Reverse osmosis.

Unit-II: Electro chemistry and Corrosion

Introduction- Electrode potential, standard electrode potential, Nernst equation (no derivation), electrochemical cell - Galvanic cell, cell representation, EMF of cell - Numerical problems. Types of reference electrodes – Quinhydrone and Calomel electrode.Construction, working and determination of pH of an unknown solution using Quinhydrone and Calomel electrode.

Corrosion: Introduction**-** Definition**,** causes and effects of corrosion – Theories of corrosion, chemical and electrochemical theories of corrosion, Factors affecting rate of corrosion - Nature of the metal, Nature of the corroding environment. Corrosion control methods - Cathodic protection Methods - Sacrificial anode and impressed current methods. Metallic coatings-Methods of application - Galvanizing and Tinning.

UNIT-III: Energy sources

Batteries: Introduction – Classification of batteries - Primary, secondary and reserve batteries with examples. Construction, working and applications of Lead – acid storage battery and Lithium-ion battery. Fuel Cells – Differences between a battery and a fuel cell, Construction and applications of Direct Methanol Fuel Cell (DMFC).

Fuels: Introduction and characteristics of a good fuel,Calorificvalue–Units-HCV,LCV-Dulong's formula - Numerical problems.

Fossil fuels: Introduction, Classification, Petroleum - Refining of Crude oil, LPG and CNG composition and uses.

Synthetic Fuels: Fischer-Tropschprocess. Introduction and applications of Hythane and Green Hydrogen.

UNIT-IV: Polymers

Definition-Classification of polymers: Based on origin and tacticity with examples –Types of polymerizations - Addition (free radical addition mechanism) and condensation polymerization. Plastics,ElastomersandFibers:Definitionandapplications(PVC,Teflon,Nylon-6,6). Differences between thermoplastics and thermo setting plastics.

Conducting polymers: Definition and Classification with examples-Mechanism of conduction in trans-poly-acetylene and applications of conducting polymers.

Biodegradable polymers: Poly lactic acid and its applications.

UNIT-V: Engineering Materials and their applications

Cement: Portland cement, its composition, setting and hardening.

Lubricants: Definition and characteristics of a good lubricant.

Properties of lubricants- viscosity, cloud and pour point, flash and fire point.

Biosensor-Definition, Amperometric Glucose monitor sensor.

Spectroscopic techniques and applications: UV-Visible spectroscopy- Principle, Selection rules, Types of electronic transitions and applications (Analysis of pollutants in

dye industry); IR spectroscopy-Principle- Mode of vibrations, Applications in night vision-security, Pollution under Control- CO sensor (Passive Infrared detection),

SUGGESTEDTEXT BOOKS:

1. Engineering Chemistry by P.C.Jain and M. Jain, Dhanpatrai PublishingCompany, 2010.
2. Engineering Chemistry by Rama Devi,Dr.P. Aparna and Rath,Cengagelearning, 2025.

REFERENCETEXT BOOKS:

1. Engineering Chemistry by ThirumalaChary,Laxminarayana&Shashikala,Pearson Publications (2020).
2. Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company(P)Ltd. Delhi 2011.
3. Engineering Chemistry by ShikhaAgarwal,CambridgeUniversityPress, Delhi 2015.
4. Engineering Analysis of Smart Material Systems byDonaldJ.Leo,Wiley,2007.
5. Challenges and Opportunities in Green Hydrogen by **Editors:** ParamvirSingh,Avinash Kumar Agarwal, Anupma Thakur, R.K Sinha.
6. Raman Spectroscopy in Human Health and Biomedicine,
7. E-Content-https://doi.org/10.1142/13094|October2023
8. E-books:
   https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2up
   https://www.worldscientific.com/ doi/epdf/10.1142/13094

25CS11001: PROGRAMMING FOR PROBLEM SOLVING

B. Tech.CSE – I Year, I Sem.
Prerequisite(s): None

L	T	P/D	C
2	-	-/-	2

Course Objectives: Develop ability to

1. To learn the fundamentals of computers.
2. To understand the various steps in program development.
3. To learn the syntax and semantics of the C programming language.
4. To learn the usage of structured programming approaches in solving problems.

Course Outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Develop algorithms and flowcharts to solve problems and implement them using C programs.	PO1, PO2, PO3, PO4, PSO1, PSO2	Unit I	BTL 3	SDG 4,9
CO2	Apply control structures and iterative statements to solve real-world problems using C.	PO1, PO2, PO3, PO4,PO10, PSO1, PSO2	Unit II	BTL 3	SDG 4,8
CO3	Design modular programs using functions, recursion, and preprocessor directives.	PO1, PO2, PO3, PSO1, PSO2	Unit III	BTL 4	SDG 9,12
CO4	Implement and analyze searching and sorting algorithms using arrays in C.	PO1, PO2, PO3, PO4, PSO1, PSO2	Unit IV	BTL 4	SDG 4,9
CO5	Utilize pointers and strings for dynamic memory management and efficient program design.	PO1, PO2, PO3, PO4, PSO1, PSO2	Unit V	BTL 4	SDG 8,9

UNIT - I: Logic Building: Flow chart, Algorithm, Pseudo code. Introduction to Programming Computer Languages, Creating and running programs, Program Development. Introduction to the C Language Background, C Programs, Identifiers, Data Types, Variables, Constants, Input/output functions. Operators Arithmetic, relational, logical, bitwise, conditional, increment/decrement, assignment. Expressions, Precedence and Associativity, Expression Evaluation, Type conversions.

UNIT– II: **Control statements**: Selection Statements (decision making) – if and switch statements. Repetition statements (loops) while, for, do-while statements. Break, continue, goto statements.

UNIT - III: Functions and Program structure: User defined functions, inter function communication, Scope and Lifetime of variables, Storage classes-auto, register, static, extern, type qualifiers. The C preprocessor. Recursive functions.

UNIT - IV: **Array**s: Declaring and Referencing Arrays, Array Subscripts, Using Array Elements as Function Arguments, Array Arguments, Basic searching in an array of elements (linear and binary search techniques), Basic algorithms to sort array of elements (Bubble, Insertion and Selection sort algorithms two – dimensional arrays matrix addition and matrix multiplication, Declaration of Multidimensional arrays.

UNIT - V:

Pointers: Introduction, Pointers and addresses, Pointer types, Pointers and function arguments, Pointers and arrays, address arithmetic, Array of Pointers, Pointers to Pointers, Pointer to Function, pointers and multi-dimensional arrays. Dynamic Memory Allocation.

Strings: String Basics, String Library Functions: Assignment and Substrings, Longer Strings: Concatenation and Whole-Line Input, String Comparison, character pointers and functions

TEXT BOOKS:

1. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson.

2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill.
3. YashvantKanetkar, Let Us C, 18th Edition, BPB.
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression).
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition.
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

25CH11L01– ENGINEERING CHEMISTRY LAB

L	T	P/D	C
-	-	2/-	1

B. Tech. CSE - I Year I Sem.

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Estimate the hardness content in water and check its suitability for drinking purpose.
2. Acquire ability to perform acid-base titrations using instrumental methods such as conductometry, potentiometry, and pH metry.
3. Gain hands-on experience in synthesizing polymers like Bakelite and Nylon – 6, 6 in the laboratory.
4. Measure physical properties like acid value and viscosity.
5. Gain conceptual understanding of experiments involving core chemical principles through virtual platforms, with relevance to engineering applications.

Course outcomes: At the end of the course, the student would be able to:

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Estimate hardness in water to verify its suitability for drinking purpose.	PO1, PO2	1	3,4	SDG 3,6,7,8, 12
CO2	Apply instrumental techniques like conductometry, potentiometry, and pH metry.	PO1, PO2	2,3,4,5,6	3,4	
CO3	Use fundamental preparatory techniques for the synthesis of polymers such as Bakelite and Nylon-6,6.	PO1, PO2	7,8	3,4	
CO4	Determine physical properties, namely acid value and viscosity of a given fluid.	PO1, PO2	10, 11	2	
CO5	Demonstrate the ability to analyze and interpret virtual experiments based on fundamental chemical principles applicable to engineering systems.	PO1, PO2	12,13,14,15	1,2	

List of Experiments: (A minimum of **TEN** Experiments are to be conducted using hardware)

1. Estimation of Hardness of water by EDTA Complexometric method.
2. Estimation of the concentration of strong acid by Conductometry.
3. Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.
4. Estimation of concentration ofFe^{+2} ionbyPotentiometryusing$KMnO_4$.
5. Estimation of concentration of strong acid with strong base by Potentiometry using quinhydrone.
6. Determination of an acid concentration using pH meter.

7. Preparation of Bakelite.
8. Preparation Nylon–6,6.
9. Determination of rate of corrosion of mild steel in the presence and absence of inhibitor.
10. Estimation of acid value of given lubricant oil.
11. Estimation of viscosity of lubricant oil using Ostwald's Viscometer.
12. Construction of Fuel cell and it's working.
13. Smart materials for Biomedical applications
14. Batteries for electrical vehicles.
15. Functioning of solar cell and its applications.

Equipment required:

1. Potentiometer cum pH meter
2. Conductometer
3. Ostwald's viscometer
4. Electric water bath
5. Glassware (Burette, Pipette, conical flask, volumetric flask, beaker)

25EN11L01–ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB

B. Tech. CSE - I Year I Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): Nil

Course Objectives: Develop an ability to

1. Enhance active listening skills
2. Listen and comprehend the speech of people from different linguistic backgrounds
3. Improve pronunciation and neutralize accent
4. Express ideas fluently and appropriately
5. Speak in social and professional contexts

Course Outcomes: At the end of the course, the student would be able to:

CO	Course Outcomes	Related POs and PSOs	Related Exercises	BTL	Related SDGs
CO1	Listen actively and identify important information in spoken texts.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	SDG: 4
CO2	Use Phonetics to neutralize accent and speak intelligibly.	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	
CO3	Articulate ideas explicitly both verbally and non- verbally	POs: 8 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2, 3,4 and 5	

Syllabus: English Language and Communication Skills Lab (ELCS) shall have two parts:

a. **Computer Assisted Language Learning (CALL) Lab which focusses on listening skills**
b. **Interactive Communication Skills (ICS) Lab which focusses on speaking skills**

The following course content is prescribed for the **English Language and Communication Skills Lab**.

Exercise – I

CALL Lab:

Instruction: Speech Sounds-Listening Skill - Importance – Purpose - Types- Barriers- Active Listening

Practice: Listening to Distinguish Speech Sounds (Minimal Pairs) **-** *Testing Exercises*

ICS Lab:

- **Diagnostic Test – Activity titled '*Express Your View'***

Instruction: Spoken and Written language - Formal and Informal English - Greetings - Introducing Oneself and Others

Practice: Any Ice-Breaking Activity

Exercise – II

CALL Lab:

Instruction: Listening vs. Hearing - Barriers to Listening

Practice: Listening for General Information - Multiple Choice Questions - *Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)*

ICS Lab:

Instruction: Features of Good Conversation – Strategies for Effective Communication

Practice: Role Play Activity - Situational Dialogues –Expressions used in Various Situations –Making Requests and Seeking Permissions — Taking Leave - Telephone Etiquette

Exercise - III

CALL Lab:

Instruction: Errors in Pronunciation – Tips for Neutralizing Mother Tongue Influence (MTI)

Practice: Differences between British and American Pronunciation –*Listening Comprehension Exercises*

ICS Lab:

Instruction: How to make Formal Presentations, Describing Objects, Situations, Process, Places, People and Events

Practice: Picture Description Activity – Looking at a Picture and Describing Objects, Situations, Places, People and Events *(A wide range of Materials / Handouts are to be made available in the lab.)* Oral Presentations.

Exercise – IV

CALL Lab:

Instruction: Techniques for *Effective* Listening

Practice: Listening for Specific Details - Listening - Gap Fill Exercises - *Listening Comprehension Exercises*

(It is essential to identify a suitable passage with exercises for practice.)

ICS Lab:

Instruction: How to Tell a Good Story - Story Star- Sequencing-Creativity

Practice: Activity on Telling and Retelling Stories - Collage

Exercise – V

CALL Lab:

Instruction: Identifying the literal and implied meaning

Practice: Listening for Evaluation - Write the Summary – Listening Comprehension Exercises

(It is essential to identify a suitable passage with exercises for practice.)

ICS Lab:

Instruction: Understanding Non-Verbal Communication

Practice: Silent Speech - Dumb Charades Activity

- **Post-Assessment Test on 'Express Your View'**

REFERENCE BOOKS:

1. Shobha, KN & Rayen, J. Lourdes. (2019). *Communicative English – A workbook.* Cambridge University Press
2. Board of Editors. (2016). *ELCS Lab Manual: A Workbook for CALL and ICS Lab Activities.* Orient BlackSwan Pvt. Ltd.
3. Mishra, Veerendra et al. (2020). *English Language Skills: A Practical Approach.* Cambridge University Press
4. (2022). *English Language Communication Skills – Lab Manual cum Workbook.* Cengage Learning India Pvt. Ltd.
5. Ur, Penny and Wright, Andrew. 2022. *Five Minute Activities – A Resource Book for Language Teachers.* Cambridge University Press.s

25CS11L01: PROGRAMMING FOR PROBLEM SOLVING LAB

B. Tech.CSE – I Year, I Sem.
Prerequisite(s): None

L	T	P/D	C
-	-	2/-	1

[Note: The programs may be executed using any available Open Source/ Freely available IDE Some of the Tools available are:
CodeLite: https://codelite.org/
Code::Blocks: http://www.codeblocks.org/
DevCpp :http://www.bloodshed.net/devcpp.html
Eclipse: http://www.eclipse.org
This list is not exhaustive and is NOT in any order of preference]

Course Objectives: The students will learn the following:

1. To work with an IDE to create, edit, compile, run and debug programs
2. To analyze the various steps in program development.
3. To develop programs to solve basic problems by understanding basic concepts in C like operators, control statements etc.
4. To develop modular, reusable and readable C Programs using the concepts like functions, arrays etc.
5. To Write programs using the Dynamic Memory Allocation concept.

Course outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Formulate the algorithms for simple problems	PO1, PO2, PO3, PO5, PSO1, PSO2	Unit I	BTL 3	SDG 4,9
CO2	Translate given algorithms to a working and correct program	PO1, PO2, PO3, PO5,PSO1, PSO2	Unit I	BTL 3	SDG 4,8
CO3	Correct syntax errors as reported by the compilers.	PO1, PO2, PO5, PSO1, PSO2	Unit I,II,III,IV,V	BTL 4	SDG 4
CO4	Identify and correct logical errors encountered during execution	PO1, PO2, PO3, PO5, PO10, PSO1, PSO2	Unit I,II,III,IV,V	BTL 3	SDG 4,9
CO5	Represent and manipulate data with arrays, strings	PO1, PO2, PO3, PO5, PSO1, PSO2	Unit IV,V	BTL 3	SDG 4,9

CO6	Use pointers of different types	PO1, PO2, PO3, PO5, PSO1, PSO2	Unit V	BTL 3	
CO7	Modularize the code with functions so that they can be reused	PO1, PO2, PO3, PO5, PSO1, PSO2	Unit III	BTL 4	

PRACTICE SESSIONS:

Simple numeric problems:

a) Write a program for finding the max and min from the three numbers.

b) Write the program for the simple, compound interest.

c) Write a program that prints a multiplication table for a given number and the number of rows in the table. For example, for a number 5 and rows = 3, the output should be:

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

d) Write a program that shows the binary equivalent of a given positive number between 0 to 255.

Expression Evaluation:

a) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement).

b) Write a program that finds if a given number is a prime number.

c) Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.

d) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

Arrays, Pointers and Functions:

a) Write a C program to find the minimum, maximum and average in an array of integers.

b) Write a C program that uses functions to perform the following: I. Addition of Two Matrices II. Multiplication of Two Matrices

c) Write a program for reading elements using a pointer into an array and display the values using the array.

d) Write a program for display values reverse order from an array using a pointer.

Strings:

a) Write a C program that uses functions to perform the following operations: I. To insert a sub-string into a given main string from a given position. II. To delete n Characters from a given position in a given string

b) Write a C program to determine if the given string is a palindrome or not (Spelled same in both directions with or without a meaning like madam, civic, noon, abcba, etc.)

c) Write a C program that displays the position of a character ch in the string S or – 1 if S doesn't contain ch.

d) Write a C program to count the lines, words and characters in a given text.

Sorting and Searching:

a) Write a C program that uses non-recursive function to search for a Key value in a given list of integers using linear search method.

b) Write a C program that uses non-recursive function to search for a Key value in a given sorted list of integers using binary search method.

c) Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.

d) Write a C program that sorts the given array of integers using selection sort in descending order

e) Write a C program that sorts the given array of integers using insertion sort in ascending order

f) Write a C program that sorts a given array of names.

TEXT BOOKS:

1. Jeri R. Hanly and Elliot B.Koffman, Problem solving and Program Design in C 7th Edition, Pearson.
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill
3. YashavantKanetkar, Let Us C, 18th Edition, BPB
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression)
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill

B.Tech CSE **AR25**

25ME11L01-ENGINEERING WORKSHOP

L	T	P/D	C
0	0	2/-	1

B.Tech. I Year, I Sem.

Prerequisites: Practical skill

Course Objectives:
1. To introduce students to basic manufacturing processes and workshop practices.
2. To provide hands-on training in carpentry, fitting, welding, sheet metal, and machining
3. To develop skills in using hand tools and measuring instruments.
4. To enhance safety awareness and proper handling of workshop equipment.
5. To build a foundational understanding of industrial production and fabrication.

Course Outcomes: At the end of the course, students would be able to

CO	Course Outcomes	Related POs and PSOs	Related Trades	BTL	Related SDGs
CO1	Understand the basic manufacturing processes and operations	POs: 1,3, and 9 PSOs: 3	1, 2, 3, 4, 5 and 7	2 and 4	SDG 4 SDG 9 SDG 12
CO2	Use hand tools and equipment safely and efficiently.	POs: 1,3, and 9 PSOs: 3	1, 2, 3, 4,5,6 and 7	3	
CO3	Perform basic operations in carpentry, fitting, welding, sheet metal work, and machining	POs: 1,3, and 9 PSOs: 3	1, 2, 3, 4, 5 and 7	2	
CO4	Read and interpret workshop drawings	POs: 1,3, and 9 PSOs: 3	1, 2, 3, 4, 5,6 and 7	2	
CO5	Develop teamwork, time management, and quality awareness in a workshop environment.	POs: 1,3, and 9 PSOs: 3	1, 2, 3, 4, 5,6 and 7	2	

1. TRADES FOR EXERCISES:

At least two exercises from each trade:

i. **Carpentry:** T- Lap Joint, Dovetail Joint, Mortise and Tenon Joint

ii. **Fitting:** V- Fit, Dovetail Fit and Semi- circular fit

iii. **Tin Smithy**: Square Tin, Rectangular Tray and Conical Funnel

iv. **Foundry**: Preparation of Green Sand Mould using Single Piece and Split Pattern

v. **Welding Practice:** Arc Welding and Gas Welding

vi. **House wiring**: Parallel and Series, Two-way Switch and Tube Light

vii. **Black Smithy**: Round to Square, Fan Hook and S- Hook

2. TRADES FOR DEMONSTRATION AND EXPOSURE:

Plumbing, Machine Shop, Metal Cutting (Water Plasma), Power tools in construction and Wood Working

TEXT BOOKS:

1. Workshop Practice, B. L. Juneja, Cengage Learning India, 1st edition, 2015.
2. Workshop Practice Manual, K. Venkata Reddy, BS Publication,6th Edition, Rpt.2025.

REFERENCE BOOKS:

1. Workshop Manual, K. Venugopal, Anuradha Publications, 2012th edition, 2012

25MA12001 – ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS

B. Tech. CSE– I Year, II Sem.
Prerequisite(s): 25MA11001- Matrices and Calculus

L	T	P/D	C
3	-	-/-	3

Course Objectives: Develop ability to

1. Solve first and higher order differential equations of various types.
2. Analyze properties of Laplace Transform, and Inverse Laplace Transform.
3. Solve Ordinary Differential Equations using Laplace Transform techniques.
4. Explain properties of vector operators to determine solenoidal and irrotational vectors, directional derivatives of vectors.
5. Determine the length of a curve, area between the surfaces and volumes of solids using vector integration.

Course Outcomes (COs): At the end of the course, students would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Form first order differential equations for Growth and Decay and apply appropriate methods for solving them	PO1,PO2, PO3 and PSO1	I	1,2,3,4	SDG4 SDG9
CO2	Form higher order differential equations for Electrical circuits and apply appropriate methods for solving them.	PO1,PO2, PO3 and PSO1	II	1,2,3,4	
CO3	Apply Laplace transform techniques to evaluate integrals and solve ordinary differential equations with initial conditions.	PO1,PO2, PO3 and PSO1	III	1,2,3,4	
CO4	Analyze and compute vector derivatives and relate vector integrals to physical and engineering applications	PO1,PO2, PO3 and PSO1	IV & V	1,2,3,4	

UNIT-I: First Order Ordinary Differential Equations 8 L

Exact Differential Equations, Equations reducible to Exact Differential Equations, Linear Differential Equations and Bernoulli's Equations orthogonal Trajectories (only in Cartesian Coordinates)

Applications: Newton's law of cooling, Law of Natural growth and decay

UNIT II: Ordinary Differential Equations of Higher Order 10 L

Higher Order Linear Differential Equations with Constant Coefficients: Non-homogeneous of the type e^{ax}, sin ax, cos ax, x^n, $e^{ax}V(x)$ and $xV(x)$**,** Method of variation of parameters.

Applications: Electrical Circuits.

UNIT III: Laplace Transforms **10 L**

Definition of Laplace transform, Existence of Laplace transforms, Laplace transform of standard functions, first shifting theorem, Laplace transform of functions when they are multiplied and divided by "t", Laplace transforms of derivatives and integrals of functions, Laplace Transform of Periodic function, Inverse Laplace transform by different methods, Convolution theorem(without proof).

Applications: Evaluation of integrals using Laplace Transforms, Solving Initial Value Problems by using Laplace Transform method.

UNIT IV: Vector Differentiation **10 L**

Vector point functions and Scalar point functions, Gradient, Divergence and Curl, Directional derivatives, Vector Identities, Scalar potential function, Solenoidal and Irrotational vectors.

UNITV: Vector Integration **10 L**

Line, Surface and Volume Integrals. Theorems of Green's Gauss and Stokes (without proofs) and their applications.

TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 44th Edition, 2017.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 5th Edition, 2016.

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 10th Edition, 2011.
2. G.B. Thomos and R.L. Finney, Calculus and Analytic Geometry, 9thEdition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A Text book of Engineering Mathematics, Laxmi Publications, 10th Edition, 2015.
4. H.K. Das and Er. RajnishVerma, Higher Engineering Mathematics, S. Chand and Company Ltd, New Delhi.

B.Tech CSE **AR25**

25PH12001- ADVANCED ENGINEERING PHYSICS

B. Tech.CSE I Year II Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Understand the fundamental concepts of quantum behavior of matter in its micro state and experimental evidence to dual nature of matter, and physical significance and application of wave function.
2. Understand the characteristics of intrinsic and extrinsic semiconductors, and applications of Hall effect.
3. Understand the concepts of quantum computing principles, quantum gates, and basic quantum algorithms.
4. Understand the properties and applications of magnetic and dielectric materials.
5. Understand the working and applications of lasers and fibreoptics in modern technology.

Course Outcomes: At the end of the course, students would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply quantum mechanical principles to explain particle behavior and energy band formation in solids.	PO1,PO2	I	1,2,3	SDG4
CO2	Classify semiconductors, interpret Fermi level variations, and apply Hall effect concept to determine the type of semiconductor.	PO1,PO2	II	1,2,3	
CO3	Explain quantum computing concepts, quantum gates, and describe basic quantum algorithms.	PO1,PO2	III	1,2,3	
CO4	Classify magnetic and dielectric materials, assess their characteristics, and apply them in technological applications.	PO1,PO2	IV	1,2,3	
CO5	Explain principles of lasers and opticalfibres, their operation and application in communication and sensing technologies.	PO1,PO2	V	1,2,3	

UNIT-I: Quantum Mechanics

Introduction, de-Broglie hypothesis, Heisenberg uncertainty principle, physical significance of wave function, postulates of quantum mechanics: operators in quantum mechanics, eigen values and eigen functions, expectation value; Schrödinger's time independent wave equation, particle in a 1D box, Bloch's theorem (qualitative), Kronig-Penney model (qualitative): E-k diagram, effective mass of electron, formation of energy bands, origin of bandgap, classification of solids, concept of discrete energy levels and quantum confinement in nanomaterials.

UNIT-II: Semiconductors

Classification of semiconductors: n-type, p-type, carrier concentration in intrinsic and extrinsic semiconductors, Fermi level in intrinsic and extrinsic semiconductors, variation of Fermi level with temperature and concentration of dopants in extrinsic semiconductors, direct and indirect band gap semiconductors, Hall effect and its applications.

UNIT-III: Quantum Computing

Introduction, linear algebra for quantum computation, Dirac's Bra and Ket notation and their properties, Hilbert space, Bloch's sphere, concept of quantum computer, classical bits, Qubits, multiple Qubit system, quantum computing system for information processing, evolution of quantum systems, quantum measurements, entanglement, quantum gates, challenges and advantages of quantum computing over classical computation, Introduction to quantum algorithms: Deutsch-Jozsa, Shor, Grover (Qualitative).

UNIT-IV: Magnetic and Dielectric Materials

Introduction to magnetic materials, origin of magnetic moment-classification of magnetic materials, hysteresis, Weiss domain theory of ferromagnetism, soft and hard magnetic materials, synthesis of ferromagnetic materials using sol-gel method, applications: magnetic hyperthermia for cancer treatment, magnets for EV, Giant Magneto Resistance (GMR) device.

Introduction to dielectric materials, types of polarization (qualitative): electronics, ionic & orientation; ferroelectric, piezoelectric, pyroelectric materials and their applications: Ferroelectric Random-Access Memory (Fe-RAM), load cell and fire sensor.

UNIT-V: Laser and Fibre Optics

Introduction to laser, characteristics of laser, Einstein coefficients and their relations, meta stable state, population inversion, pumping, lasing action, Ruby laser, He-Ne laser, CO2 laser, semiconductor diode laser, applications: Bar code scanner, LIDAR for autonomous vehicle.

Introduction to fibre optics, total internal reflection, construction of optical fibre, acceptance angle, numerical aperture, classification of optical fibres, losses in optical fibre, applications: optical fibre for communication system, sensor for structural health monitoring.

TEXT BOOKS:

1. Walter Borchardt-Ott,*Crystallography:An Introduction*, Springer.
2. Charles Kittel, *Introduction to Solid State Physics*,JohnWiley&Sons,Inc.
3. ThomasG.Wong, *Introduction to Classical and Quantum Computing*,RootedGrove
4. Physics, Halliday, Resnick and Krane, Wiley Publishers, 5th edition, 2018.
5. Engineering Physics, B.K. Pandey, S. Chaturvedi, Cengage Learing.2012.

REFERENCE BOOKS:

1. Jozef Gruska, *Quantum Computing*, McGrawHill
2. MichaelA.Nielsen&IsaacL.Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press.
3. JohnM.Senior, *Optical Fiber Communications Principles and Practice*, Pearson Education Limited.
4. A Textbook of Engineering Physics, Dr. M. N. Avadhanulu, Dr. P.G. Kshirsagar, S. Chand, Revised edition, 2018.

25EE12001– BASIC ELECTRICAL ENGINEERING

B.Tech. CSE - I Year, II Sem.
Prerequisite(s): None

L	T	P/D	C
3	-	-/-	3

Course Objectives: Develop an ability to

1. Provide a strong foundation in electrical engineering concepts to enable students to analyze and solve DC and AC electrical circuits.
2. Understand the construction, working principles, and characteristics of electrical machines.
3. Evaluate the performance of electrical machines through experimental, analytical, and simulation-based approaches.
4. Familiarize with electrical installation practices, wiring systems, and safety measures essential for professional engineering applications.
5. Promote teamwork, critical thinking, and problem-solving skills through problem/project-based learning activities relevant to real-world electrical systems.

Course Outcomes (COs): On completion of the course, the student would be able to:

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Analyze and solve DC and AC circuit problems using fundamental laws, theorems, and phasor concepts by applying electrical engineering concepts, while demonstrating teamwork and problem-solving skills through problem/project-based learning	PO1, PO2 PO4, PO6, PO8, PO11 & PSO1 and 2	Unit I & Unit II	BTL-4	SDG 7 SDG 9
CO2	Explain the construction, operation, and performance of transformers and electrical machines.	PO1, PO2, PO4, PO8 and PO11 & PSO1	Unit III & Unit IV	BTL-2	SDG 7 SDG 9
CO3	Evaluate performance of various DC and AC machines.	PO1, PO2, PO4, and PO11 & PSO1 and 2	Unit IV	BTL-5	SDG 7 SDG 12
CO4	Apply principles of electrical installations, wiring systems, and safe practices in engineering applications.	PO1, PO2, PO3, PO6, PO8 and PO11 & PSO1	Unit V	BTL-3	SDG 3 SDG7 SDG 11

UNIT-I: D.C. Circuit Analysis: Electrical circuit elements (R, L and C), voltage and current sources, KVL&KCL, analysis of simple circuits with dc excitation. Superposition, Thevenin and Norton Theorems. Transient response of first-order RL and RC circuits (with DC excitation).

UNIT-II: A.C. Circuit Analysis: Representation of sinusoidal waveforms; determination of average and RMS values; phasor representation. Computation of real, reactive, and apparent power; analysis of power factor. Analysis of single-phase AC circuits and Resonance in series R–L–C Circuit.

UNIT-III: Transformer: Ideal and practical transformer, equivalent circuit, losses and efficiency in transformers. Auto-transformer and Fundamentals of three-phase transformer connections. Applications of transformers.

UNIT-IV: Electrical Machines: DC machines – construction, working principle and Applications; Three-Phase Induction Motor – generation of rotating magnetic field, construction, operation and its applications. Single-Phase Induction Motor – construction and operation.

UNIT-V: Electrical Installations: Components of LT switchgear – Types of Fuses, MCB, ELCB, MCCB; types of wires and cables; earthing methods. Batteries – types and key characteristics. Basic energy consumption calculations. Applications of installations.

TEXT BOOKS:

1. D.P. Kothari and I. J. Nagrath, “Basic Electrical Engineering”, Tata McGraw Hill, 4th Edition,2019.
2. MS Naidu and S Kamakshaiah, “Basic Electrical Engineering”, Tata McGraw Hill, 2nd Edition,2008.

REFERENCE BOOKS:

1. P. Ramana, M. Suryakalavathi, G.T. Chandrasheker, “Basic Electrical Engineering”, S. Chand, 2nd Edition, 2019.
2. D. C. Kulshreshtha, “Basic Electrical Engineering”, McGraw Hill, 2009
3. Abhijit Chakrabarthi, Sudipta Debnath, Chandan Kumar Chanda, “Basic Electrical Engineering”, 2nd Edition, McGraw Hill, 2021.
4. L. S. Bobrow, “Fundamentals of Electrical Engineering”, Oxford University Press, 2011.
5. V. D. Toro, “Electrical Engineering Fundamentals”, Prentice Hall India, 1989.

25CS12001: DATA STRUCTURES

B. Tech.CSE – I Year, II Sem.
Prerequisite(s): Programming for Problem Solving

L	T	P/D	C
3	-	-/-	3

Course Objectives: Develop ability to

1. Introduce students to advanced data representation techniques in C using structures, unions, enumerations, and typedef to effectively organize and manipulate complex data types.
2. Proficiency in file handling and data storage concepts, including text and binary file operations, database searching, file positioning, and multifile program design.
3. Build foundation in abstract data types and linear data structures, enabling students to implement and manage linked lists, circular lists, and doubly linked lists for efficient data organization.
4. Train students in the use of stacks and queues for solving computational problems such as expression conversion, evaluation, and balancing of symbols through algorithmic thinking.
5. Equip students with knowledge of hierarchical and network data structures, including trees and graphs, and their associated algorithms for searching, traversal, and application in problem-solving.

Course outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply user-defined data types such as structures, unions, and enumerations to represent complex data.	PO1, PO2, PSO1	Unit I	BTL 2,3	SDG4
CO2	Implement file operations on text and binary files for data storage, retrieval, and maintenance.	PO1,PO2, PO3, PSO2	Unit II	BTL 3,4	SDG 9
CO3	Develop and manipulate linear data structures including linked lists, stacks, and queues.	PO1,PO3, PO4, PSO1	Unit III,IV	BTL3, 4	SDG 8
CO4	Design and execute algorithms for trees and graphs including traversal, searching, and updating.	PO1,PO4, PO5, PSO2	Unit V	BTL 4,5	SDG 9

UNIT – I:

Structure and Union Types: Introduction, User-Defined Structure Types, Structure Type Data as Input and Output Parameters, Functions with Structured Result Values, Complex Structures, Self-Referential Structures, Bit Fields, Union Types, typedef, Enumeration.

UNIT – II:

Text and Binary File Pointers: Files Introduction, Modes of File, Input/ Output Files - Review and Further Study, Binary Files, Searching a Database, File status functions, File positioning functions, Command Line Arguments, Multifile Programming.

UNIT–III:

Introduction to Data Structures: Abstract data types, selecting a Data Structure, Linear list —Introduction, singly linked list, Circular Linked Lists, Doubly Linked List.

UNIT – IV:

Stacks: Stack ADT, Stack applications -Infix Expression to Postfix Expression Conversion, Postfix Expression Evaluation, Balancing Symbols, Expression Tree, Queues- Queue ADT

UNIT – V:

Trees: Introduction, Types of Trees, creating a Binary Tree from a General Tree, traversing a Binary Tree, Binary Search Trees (BST),BST Operations- Searching, Insertion and Deletion, BST ADT.

Graphs: Introduction to types of Graphs, Representation of Graphs, Graph Traversal Algorithms – Depth First Search, Breadth First Search, Graph ADT, Applications of Graphs.

TEXTBOOKS:

1. DataStructures:APseudocodeApproachwithC,2ndEdition,R.F.GilbergandB.A.Forouzan, Cengage Learning
2. DataStructureusingC–ReemaThareja,3rd Edition, Oxford University Press.
3. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

REFERENCE:

1. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.
2. BrianW.Kernighan and Dennis M.Ritchie, The C Programming Language, Prentice Hall of India.

25CS12003-DISCRETE MATHEMATICS

B.Tech. CSE - I Year, II Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives

Enable student to

1. Understand concepts of Mathematical Logic, mechanisms of inference rules for propositional and predicate logic and their applications.
2. Understand the concepts of Sets, Relations, Functions and their applications.
3. Learn the concepts of Algebraic Structures, and Boolean algebra
4. Understand basics of counting, Principles of Inclusion-Exclusion.
5. Understand basic definitions and properties of graphs and their applications in computer science and engineering.

Course Outcomes (COs)

After completion of the course, student would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Analyze and construct valid logical arguments using propositional and predicate calculus.	PO1, PO2,PO3, PSO1	Unit I	BTL 4	SDG 4 SDG 9
CO2	Apply concepts of set theory, relations, and functions to represent and manipulate discrete structures.	PO1, PO2,PO3, PSO1	Unit II	BTL 3	SDG 4 SDG 9
CO3	Analyze and apply algebraic structures to solve computational and logical problems	PO1, PO2,PO3, PSO1	Unit III	BTL 4	SDG 8 SDG 9
CO4	Apply combinatorial methods to solve counting, permutation, and combination problems in discrete structures.	PO1, PO2,PO3, PSO1	Unit IV	BTL 3	SDG 4 SDG 9
CO5	Apply graph theory concepts to model and solve computing problems.	PO1, PO2,PO3, PSO1	Unit V	BTL 3	SDG 9 SDG 11

UNIT-I

Mathematical Logic: Mathematical logic: Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

UNIT-II

Set Theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions.

UNIT-III

Algebraic structures: Introduction, Algebraic Systems, Semi groups and Monoids, Lattices as Partially Ordered Sets, Boolean algebra.

UNIT-IV

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutation with Constrained Repetitions, Binomial Coefficient, The Binomial and Multinomial Theorems, The Principle of Inclusion -Exclusion.

UNIT-V

Graph Theory: Basic Concepts, Isomorphism and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi-graphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

TEXT BOOK(S)

1. Discrete Mathematical Structures with Applications to Computer Science, J.P.Tremblay, R.Manohar, 1st Edition, Tata McGraw Hill, 2001. (Unit 1,Unit 2, Unit 3 - Algebraic structures)
2. Discrete Mathematics for Computer Scientists & Mathematicians, J.L. Mott, A. Kandel, T.P. Baker, 2nd Edition, PHI, 2009. (Unit 3-Elementary Combinatorics, Unit 4,Unit 5)

REFERENCE BOOK(S)

1. Discrete Mathematics and its Applications, Kenneth H. Rosen, Fifth Edition. TMH.
2. Discrete Mathematical structures Theory and application-Malik & Sen, Cengage.
3. Discrete Mathematics with Applications, Thomas Koshy, Elsevier.
4. Logic and Discrete Mathematics, Grass Man &Trembley, Pearson Education.

25ME12001-ENGINEERING DRAWING AND COMPUTER AIDED DRAFTING

L	T	P/D	C
2	0	2	3

B.Tech. CSE - I Year, II Sem.

Prerequisite(s): None

Course Objectives:
1. To introduce the fundamentals of engineering drawing and projection systems.
2. To develop skills in constructing orthographic, isometric, and sectional views.
3. To train students in interpreting and creating technical drawings using CAD tools.
4. To familiarize students with dimensioning standards and drafting conventions.
5. To bridge manual drafting techniques with computer-aided drafting practices.

Course Outcomes: At the end of the course, the student will be able to:

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
1	Understand and apply the concepts of Auto-CAD commands to practice Engineering Drawing.	POs: 1,3 and 9 PSOs: Nil	1, 2, 3, 4, and 5	2,3,4 and 5	SDG 4 SDG 9 SDG 11
2	Construct scales, Geometric curves (Conic sections & Cylindrical curves) by using Auto- CAD.	POs: 1,3 and 9 PSOs: Nil	1	2,3,4 and 5	
3	Apply the principles of Orthographic projections to draw points , Straight lines , Planes and regular solids by using Auto-CAD.	POs: 1,3 and 9 PSOs: Nil	2, 3	2,3,4 and 5	
4	Develop the sectional views and surfaces of a solid Geometries by using Auto-CAD.	POs: 1,3 and 9 PSOs: Nil	4	2,3,4 and 5	
5	Demonstrate drafting skills for Isometric and Orthographic views.	POs: 1,3 and 9 PSOs: Nil	5	2,3,4 and 5	

UNIT – I: Introduction to Engineering Graphics (Conventional)
Principles of Engineering Graphics and their Significance, Geometrical Constructions, Scales, Plain and Diagonal, Conic Sections including the Rectangular Hyperbola, General method only. Cycloid, Epicycloid and Hypocycloid.

UNIT - II: Orthographic Projections (Conventional and Computer Aided)
Introduction to Computer aided drafting, views, commands.
Principles of Orthographic Projections, Conventions, Projections of Points and Lines (Lines Inclined to both the Planes).

UNIT – III: Projections of Regular Planes and Solids (Conventional and Computer Aided)
Projections of Plane regular geometric figures. Computer aided orthographic projections of planes (Planes inclined to both the planes).
Right Regular Solids (Axis inclined to one plane)-Prism, Cylinder, Pyramid, Cone, Computer aided projections of planes & solids.

UNIT – IV: Sections of Solids and Development of Surfaces (Conventional)
Sectional views and development surfaces of Prism, Cylinder, Pyramid and Cone.

UNIT – V: Isometric Projections (Conventional and Computer Aided)
Principles of Isometric Projection, Isometric Scale, Isometric Views, Conventions, Isometric Views of Lines, Plane Figures, Simple and Compound Solids. Conversion of Isometric Views to Orthographic Views and Vice- versa.

Note:
1. The End Semester Examination will be in computer mode.
2. CIE – I will be in conventional/ computer mode.
3. CIE – II will be in computer mode.

TEXT BOOKS:
1. Engineering Drawing, N. D. Bhatt, Charotar, 54th Edition, 2023.
2. Engineering Drawing and graphics Using AutoCAD, T. Jeyapoovan and Vikas, S. Chand and company Ltd., 3rdEdition,2010.

REFERENCE BOOKS:
1. Engineering Drawing, Basant Agrawal and C.M. Agrawal, McGraw Hill, 3rd Edition, 2019.
2. Engineering Graphics and Design, WILEY, John Wiley and Sons Inc, 3rdEdition, 2020.
3. Engineering Drawing, M. B. Shah and B.C. Rane, Pearson, 2nd Edition, 2009.
4. Engineering Drawing, N. S. Parthasarathy and Vela Murali, Oxford, 1st Edition, 2015.
5.Computer Aided Engineering Drawing, K. Balaveera Reddy, CBS Publishers, 2nd Edition, 2015.

25EE12L01- BASIC ELECTRICAL ENGINEERING LAB

B.Tech. CSE - I Year, II Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): None.

Course Objectives: Develop an ability to

1. Understand basic electrical laws and network theorems to enable students to analyze DC and AC electrical circuits effectively.
2. Analyze transient behaviour of RLC circuits under different excitation conditions and understand their practical significance in electrical systems.
3. Familiarize students with the construction, operation, and testing of electrical machines and transformers, and enable them to evaluate their performance through suitable experimental methods.

CO	Course Outcomes	Related POs and PSOs	Related Experiments	BTL	Related SDGs
CO1	Analyze and solve DC and AC circuit problems using fundamental laws and theorems by applying electrical engineering concepts, while demonstrating teamwork and problem-solving skills through problem/project-based learning	PO1, PO2 PO4, PO6, PO8 and PO11 & PSO1 and 2	Exp 1,2 & 3	BTL-4	SDG 7 SDG 9
CO2	Analyze the transient response of various combinations of R, L and C circuits for different input conditions.	PO1, PO2 PO4, PO8 and PO11 & PSO1	Exp 4 & 6	BTL-4	SDG3 SDG7 SDG 11
CO3	Evaluate the performance of Electrical Machines and Transformers through various testing methods.	PO1, PO2 PO4, PO8 and PO11 & PSO1	Exp 7,8,9 and 10	BTL-5	SDG 7 SDG 12

LIST OF EXPERIMENTS:

1. Verification of KVL and KCL
2. Verification of Thevenin's and Norton's theorem
3. Verification of Superposition theorem
4. Transient response of series RL and RC circuits for DC excitation
5. Resonance in Series RLC circuit

6. Calculations and verification of Impedance and Current of RLC series circuits
7. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single-Phase Transformer
8. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
9. Performance Characteristics of a DC Shunt Motor
10. Torque-Speed Characteristics of a Three-phase Induction Motor.

ADDITIONAL EXPERIMENTS:

1. Three Phase Transformer: Verification of Relationship between Voltages and Currents (Star-Delta, Delta-Delta, Delta-star, Star-Star)
2. Measurement of Active and Reactive Power in a balanced Three-phase circuit

B.Tech CSE **AR25**

25PH12L01 - ADVANCED ENGINEERING PHYSICS LABORATORY

B. Tech.CSE -I Year II Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): None

Course Objectives: Develop an ability to

1. To provide practical exposure to advanced concepts in solid-state and modern physics.
2. To study the physical properties of materials like semiconductors, ferromagnetic, and ferroelectric substances.
3. To perform semiconductor characterization using Hall effect and band gap experiments.
4. To explore the working principles of lasers and optical fibers through hands-on experiments.
5. To develop skills in data analysis, interpretation, and scientific reporting.

Course outcomes: At the end of the course, the student would be able to:

CO	Course Outcomes	Related POs and PSOs	Related Experiments	BTL	Related SDGs
CO1	Analyze the characteristics of Solar cell and LED	PO1,PO2	3,4	3,4	SDG4
CO2	Determine key electrical, magnetic, and optical properties of semiconductors and other functional materials.	PO1,PO2	6,7,8,9	3,4	
CO3	Characterize semiconductors using Hall effect and energy gap measurement techniques.	PO1,PO2	1,2	3,4	
CO4	Demonstrate working knowledge of laser systems and optical fiber parameters through experimental study.	PO1,PO2	5,10	3,4	
CO5	Apply scientific methods for accurate data collection, analysis, and technical report writing.	PO1,PO2	All	3,4	

List of Experiments: (A minimum of **Eight** Experiments)

1. Determination of energy gap of a semiconductor.
2. Determination of Hall coefficient and carrier concentration of a given semiconductor.
3. Plot the V-I characteristics of a Solar cell.
4. Determination of Planck's constant using the V-I characteristics of the LED.
5. a. Determination of wavelength of a laser using a diffraction grating.
   b. Study of V-I & L-I characteristics of a given laser diode.

6. Determination of the magnetic moment of a bar magnet and the horizontal Earth's magnetic field.
7. Study of the B-H curve of a ferromagnetic material.
8. Study of the P-E loop of a given ferroelectric crystal.
9. Determination of the dielectric constant of a given material.
10. a. Determination of the numerical aperture of a given optical fibre.
    b. Determination of bending losses of a given optical fibre.

Equipment required:

1.Energy gap apparatus with thermometer.
2. Hall Effect Apparatus.
3. Solar cell arrangement with light source.
4. Characteristics of LED and LASER diode circuit board.
5. Bar magnet and other apparatus.
6. B-H Curve kit.
7. P-E loop kit.
8. RC circuit board and stop clock.
9. Optical fiber kit.

25CS12L01: DATA STRUCTURES LAB

B. Tech.CSE – I Year, II Sem.
Prerequisite(s): Programming for Problem Solving

L	T	P/D	C
-	-	2/-	1

Course Objectives: Develop ability to

1. Introduce students to advanced data representation techniques in C using structures, unions, enumerations, and typedef to effectively organize and manipulate complex data types.
2. Proficiency in file handling and data storage concepts, including text and binary file operations, database searching, file positioning, and multifile program design.
3. Build foundation in abstract data types and linear data structures, enabling students to implement and manage linked lists, circular lists, and doubly linked lists for efficient data organization.
4. Train students in the use of stacks and queues for solving computational problems such as expression conversion, evaluation, and balancing of symbols through algorithmic thinking.
5. Equip students with knowledge of hierarchical and network data structures, including trees and graphs, and their associated algorithms for searching, traversal, and application in problem-solving.

Course outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.	PO1, PO2, PO3, PO4, PO5,PSO1, PSO2	Unit I,II,III,IV	BTL 3,4	SDG 4,8,9
CO2	Ability to implement the concepts of Trees and Graphs	PO1, PO2, PO3, PO4, PO5,PSO1, PSO2	Unit V	BTL 4	SDG 4,9

List of Experiments

1. Write a program that uses functions to perform the following operations on singly linked list.:

   i) Creation ii) Insertion iii) Deletion iv) Traversal

2. Write a program that uses functions to perform the following operations on doubly linked list.:

i) Creation ii) Insertion iii) Deletion iv) Traversal

3. Write a program that uses functions to perform the following operations on circular linked list.:

i) Creation ii) Insertion iii) Deletion iv) Traversal

4. Write a program that implement stack (its operations) using

i) Arrays ii) ADT

5. Write a program that implement Queue (its operations) using

i) Arrays ii) ADT

6. Write a program to implement the tree traversal methods (Recursive and Non-Recursive).

7. Write a program to implement Binary Search tree

8. Write a program to implement the Graph traversal methods.

i)DFS ii)BFS

TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.

2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCE BOOK:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

25CS21001–ADVANCED DATA STRUCTURES

B. Tech.CSE -II Year I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): 25CS12001: Data Structures

Course Objectives: Develop an ability to

1. Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
2. Introduces sorting and pattern matching algorithms.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Use asymptotic notation in representing space and time complexities of algorithms.	PO1, PO2, PO3, PO4, PO5, PSO2	I	3	4
CO2	Construct different linear and non-linear data structures using array and linked representations and use them in appropriate applications.	PO1, PO2, PO3, PO4, PO5, PO9, P10, PO11, PSO2	I,III,IV	4	4,9
CO3	Implement, use and compare different sorting, searching, Hashing and pattern-matching algorithms for solving various problems	PO1, PO2, PO3, PO4, PO5, PO9, P10, PO11, PSO2	II, V	5	4,9

UNIT – I:

Performance Analysis: Time and Space Complexity, Asymptotic Notation.
Row and Column major order memory mappings, Disjoint Sets: representation and operations, Sparse Matrices: Array representation and operations

UNIT – II: Sorting and Searching

Comparison based Sorting: Merge Sort, Quick Sort, Shell Sort, Tree Sort
Non-Comparison based sorting: Counting sort, Radix sort, bucket sort
Searching: Interpolation Search, Jump Search, Fibonacci Search and Skip list search.

UNIT – III: Search Trees

Binary Search Trees: AVL Tree, Red-Black Tree, Splay Tree
Digital Search Trees: Binary Tries, Patricia Tries, Suffix Tries and Standard Tries.

UNIT – IV:
Multi way Search Trees and Heaps:
Multi way Search Tree: B-Trees, B-Trees ADT, 2-3 Trees, B* Trees, B+ Trees.
Heaps: Binary Heaps, Binomial Heaps, Fibonacci Heaps, Comparisons of various Heaps, Applications.

UNIT – V: Hashing and Pattern Matching:
Hashing: Introduction, Hash tables, Hash Functions:Division Method, Multiplication Method, Mid-Square Method, Folding Method
Collision Resolution: Open Addressing(linear, quadratic probing and double hashing) and Closed addressing(Chaining).
String Pattern matching: Brute Force, Knuth-Morris-Pratt, Boyer-Moore, Rabin-Karp.

TEXTBOOKS:
1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A.Forouzan, Cengage Learning.
2. Data Structure using C– Reema Thareja, 3rd Edition, Oxford University Press.

REFERENCEBOOKS:
1. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, 1st Edition, PHI/Pearson Education.

25CS21002–OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B. Tech.CSE -II Year I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Understand the principles of object-oriented programming and Java basics, including data types, operators, control statements, and class/object concepts.

2. Learn and apply inheritance, packages, interfaces, and polymorphism for building modular and reusable applications.

3. Develop robust programs by implementing exception handling and multithreading concepts in Java.

4. Utilize advanced Java features such as generics, lambda expressions, I/O streams, and collection framework for efficient data management.

5. Design and implement interactive GUI-based applications using AWT, Swing, event handling, and layout managers.

Course Outcomes Mapping

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Explain and apply OOP concepts, Java syntax, control structures, classes, objects, constructors, methods, recursion, and String operations.	PO1, PO2, PO4	I	2	4, 8, 9, 11
CO2	Demonstrate inheritance, polymorphism, packages, and interface usage to create well-structured and maintainable Java programs.	PO1, PO2, PO3, PO4, PO11	II	3	
CO3	Implement exception handling and multithreading mechanisms to develop efficient, safe, and concurrent Java applications.	PO1, PO2, PO4, PO5	III	4	
CO4	Apply generics, lambda expressions, Java I/O streams, and collections to handle and process data effectively in applications.	PO1, PO2, PO3, PO5, PO11	IV	3	
CO5	Design and develop GUI-based applications using AWT/Swing components, event handling, and layout managers for interactive interfaces.	PO1, PO2, PO3, PO5, PO6, PO7, PO11	V	5,6	

UNIT - I

Object oriented thinking and Java Basics- Need for oop paradigm, summary of oop concepts, coping with complexity, abstraction mechanisms. History of Java, Java buzzwords, data types, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring String class.

UNIT - II

Inheritance, Packages and Interfaces – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super keyword uses, using final keyword with inheritance, polymorphism- method overriding, abstract classes, the Object class. Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

UNIT - III

Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception subclasses. Differences between multithreading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads.

UNIT - IV

Enumerations, auto boxing and annotations, generics, lambda expressions.
Introduction to java I/O: byte streams, character streams, serialization.
Introduction to java Collections: Collection classes- ArrayList, LinkedList, Stack, PriorityQueue, TreeSet, HashMap, TreeMap

UNIT - V

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

Swing – Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JFrame and JComponent, JLabel, ImageIcon, JTextField, JButton, JCheckBox, JRadioButton, JList, JComboBox, Tabbed Panes, Scroll Panes, Trees, and Tables. Menu Basics, Menu related classes - JMenuBar, JMenu, JMenuItem, JCheckBoxMenuItem, JRadioButtonMenuItem, JSeperator. creating a popup menu.

TEXTBOOKS:

1. **Schildt, H. (2024). Java: The Complete Reference (13th ed.). McGraw-Hill**

REFERENCEBOOKS:

1. Horstmann, C. S. (2021). Core Java, Volume I – Fundamentals (12th ed.). Pearson Education.
2. Somashekara, M. T., Guru, D. S., & Manjunatha, K. S. (2017). Object Oriented Programming with Java (2nd ed.). PHI Learning Private Limited..
3. Malhotra, S., & Choudhary, S. (2018). Programming in Java (2nd ed.). Oxford University Press India.

25CS21003-DATABASE MANAGEMENT SYSTEMS

B. Tech.CSE -II Year I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): 25CS12001:DATA STRUCTURES

Course Objectives

Develop ability to

1. Learn and practice data modeling using entity-relationship and develop database design.
2. Understand the features of database management systems, relational databases and schema refinement.
3. Understand Structured Query Language (SQL) and NoSQL databases.
4. Understand needs of database processing and learn techniques for controlling the consequences of concurrent data access.
5. Understand various indexing techniques for the data stored in devices.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Design simple database using ER modelling and apply additional features of ER model.	PO1, PO2, PO3, PO5, PO6, PO11, PSO1	Unit I	2	4,5,8,9,12
CO2	Apply constraints on a relational model, querying using relational algebra and normalization techniques to arrive at a minimally redundant database.	PO1, PO2, PO3, PO4, PO5, PO11, PSO1	Unit II	3	
CO3	Apply SQL querying and distinguish SQL and NoSQL databases.	PO1, PO2, PO3, PO5, PSO1, PSO3	Unit III	4	
CO4	Apply concepts of concurrency control and data recovery in database transactions.	PO1, PO2, PO3, PO4, PO5, PO6, PO11, PSO1	Unit IV	4	
CO5	Apply indexing techniques to organize the data on the secondary storage devices enabling efficient data retrieval.	PO1, PO2, PO3, PO5, PO11, PSO1, PSO3	Unit V	5	

UNIT-I

Database System Applications: A Historical Perspective, File Systems versus a DBMS, the DataModel, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model

UNIT-II

Introduction to the Relational Model: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views.

Relational Algebra, Tuple relational Calculus, Domain relational calculus.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued dependencies, FOURTH normal form, FIFTH normal form.

UNIT-III

SQL languages: DDL, DML, TCL, DCL.
SQLQUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.

NoSQL Databases: Introduction, Overview, and History of NoSQL Databases. The Definition of the Four Types of NoSQL Databases. Comparison of relational databases to NoSQL, SQL Vs NoSQL, basic CRUD operations using Mongo DB.

UNIT-IV

Transaction Management: Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, testing for serializability, Lock-Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity.

Recovery System: Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions. Buffer Management – Failure with loss of non-volatile storage -Advance Recovery systems- Remote Backup systems.

UNIT-V

Storage and Indexing: File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Comparison of File Organizations, Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM),B+ Trees: A Dynamic Index Structure Storing and retrieving images, storing and retrieving files.

TEXT BOOK(S)

1. Data base System Concepts, Silberschatz, Korth, McGraw hill, VI edition.
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill, 3rd Edition.
3. Seven NoSQL Databases in a Week, Aaron Ploetz, Devram Kandhare, Sudarshan Kadambi, Xun (Brian) Wu, O'reillyPackt Publishing Ltd, 2018.

REFERENCE BOOK(S)

1. Fundamentals of Database Systems, Elmasri, Navathe, 7th Edition, Pearson Education, 2016.
2. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
3. Introduction to Database Systems, C.J.Date, Pearson Education.

25EC21002–DIGITAL DESIGN

B.Tech. CSE - II Year, I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None

Course Objectives: Develop an ability to

1. Understand basic concepts of various number systems used in digital systems.
2. Understand Boolean algebra and various Boolean simplification theorems.
3. Understand simplification of Boolean functions using k-map and tabular method.
4. Understand design and analysis of combinational and sequential logic circuits.
5. Understand symmetric functions and design the same using relay contacts.
6. Understand Threshold logic and design switching functions using threshold elements

Course Outcomes Mapping

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply knowledge of number systems, codes and Boolean algebra to the analysis and design of digital logic circuits	PO1, PO2,PO3, PO4, PSO1	1	3	4, 8, 9, 12
CO2	Apply the knowledge of logic gates to design and implement various digital circuits	PO1, PO2, PO3, PO4, PSO1	II	3	
CO3	Identify, formulate, and solve simple problems in the area of digital logic circuit design.	PO1, PO2, PO3, PO4, PSO1	III	4	
CO4	Apply the concepts of symmetric functions, Threshold logic to design logic circuits.	PO1, PO2, PO3, PO4, PSO1	IV	3	
CO5	Design digital circuits, component(s) or process to meet desired needs within realistic constraints	PO1, PO2, PO3, PO4, PSO1	V	3	

UNIT–I:

Number Systems: Binary, Octal, Decimal, Hexadecimal, Fixed-point and Floating-point Number Representations, Complements of Numbers: 1's and 2's Complement, Error Detection and Correction Codes: Parity Check, Hamming Code.

Boolean Algebra and Logic Gates: Axiomatic definitions, basic theorems and properties, Boolean Functions: Canonical and standard forms, Digital Logic Gates Overview.

UNIT-II:

Gate-Level Minimization Techniques: Karnaugh maps: 2, 3, and 4 variables, Sum-of-products (SOP) and product-of-sums (POS) simplification, Don't care conditions, Implementation using NAND and NOR gates.

UNIT-III:

Combinational Logic Circuits: Analysis and design procedures, Binary adder-subtractor and BCD adder, magnitude comparator, decoders, encoders, multiplexers and demultiplexers.

UNIT IV: Synthesis of Symmetric Networks

Relay Contacts, Analysis and Synthesis of Contact Networks, Symmetric Networks, Identification of Symmetric Functions and realization of the same.

Threshold Logic: Threshold Element, Capabilities and Limitations of Threshold logic, Elementary Properties, Synthesis of threshold networks (Unate function, Linear separability, Identification and realization of threshold functions, Map based synthesis of two-level Threshold networks).

UNIT V: Sequential Machines Fundamentals

Sequential Logic Circuits: Gated latches, Flip-flops: Clocked S-R, D, T, JK, Master-Slave JK,Design of synchronous and asynchronous counters, Shift registers: types and applications

Synchronous Sequential Logic Moore and Mealystate machines, State diagrams, SDtate tables, and state reduction, Case studies: sequence detector, traffic light controller, vending machine.

TEXT BOOKS:

1. Switching and Finite Automata Theory, Zvi Kohavi & Niraj K. Jha, 2nd Edition, 2009, Cambridge University Press.
2. M.Morris Mano, Michael D.Ciletti, Digital Design with an Introduction to the Verilog HDL,6th Edition, Pearson Education/PHI, 2017.

REFERENCE BOOKS:

1. Digital Fundamentals - A Systems Approach", Thomas L. Floyd, Pearson, 2013.
2. Fundamentals of Logic Design, Charles H. Roth, Cengage Learning, 5th Edition, 2004.
3. Digital Design, Morris Mano, PHl, 3rd Edition

25MS21002–BUSINESS ECONOMICS AND FINANCIAL ANALYSIS

B.Tech. CSE - II Year, I Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): None.

Course Objectives

Develop ability to

1. Learn the basic business types
2. Understand the impact of the economy on business and firms specifically.
3. Analyze the business firms from the managerial perspective.
4. Learn fundamental concepts of Accounting.
5. Learn financial perspective of an organization.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply micro and macroeconomics concepts of business entities.	4,6,7,10,11	I, II, III	3	8,9,16
CO2	Assess the elasticity of demand and types of market structures in business operations.	3,4,7,10	II	5	
CO3	Apply the concepts of theories of production and demand forecasting in decision-making.	3,4,6,7,10,11	II, III	3	
CO4	Evaluate and interpret the financial statements.	3,4,8,10,11	IV	5	
CO5	Evaluate and interpret the financial performance of an organization.	3,4,6,7,8,9,10,11	IV, V	5	

UNIT I: Introduction to Business and Economics Business: Types of Business Firm, Conventional and Non-Conventional Sources of Capital for a Company. **Economics:** Significance of Economics, Micro and Macro Economic Concepts, Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics. Concepts and Importance of National Income, Inflation, Money Supply and Inflation, Business Cycle, Features and Phases of Business Cycle.

UNIT II: Demand and Supply Analysis: Demand Function and Law of Demand, Determinants of demand. Elasticity of Demand: Types of Elasticity, Measurement and Significance of Elasticity, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making. Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting. Supply Analysis: Determinants of Supply, Supply Function and Law of Supply.

UNIT III: Production, Cost, Pricing & Market Structures: Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions. **Cost analysis:** Types of Costs, Short run and Long run Cost Functions. **Pricing:** Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis. **Market Structures:** Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, and Monopolistic Competition.

UNIT IV: Financial Accounting: Accounting concepts and Conventions, Accounting cycle, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts (Simple Problems).

UNIT V: **Financial Ratios Analysis:** Concept of Ratio Analysis, Importance and Types of Ratios, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios – Analysis and Interpretation (simple problems).

TEXT BOOKS:

1. D. D. Chaturvedi, S. L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata Mc –Graw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.

REFERENCE BOOKS:

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press, 2015.
2. S. N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, VikasPublications, 2013.

25CS21L01–ADVANCED DATA STRUCTURES LAB

B.Tech. CSE - II Year, I Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): 25CS12001: Data Structures

Course Objectives

Develop ability to

1. Learn the Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
2. Introduces sorting and pattern matching algorithms.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Mapped Experiments	BTL	Related SDGs
CO1	Implement disjoint sets and perform Find and Union operations.	PO1, PO2, PO3, PO4, PO5, PSO2	1	2,3	4,9
CO2	Construct different linear and non-linear data structures using array and linked representations and use them in appropriate applications.	PO1, PO2,PO3, PO4, PO5, PO9, P10, PO11, PSO2	6,7, 8, 9, 10	4	
CO3	Implement, use and compare different sorting, searching and pattern-matching algorithms for solving various problems	PO1, PO2,PO3, PO4, PO5, PO9, P10, PO11, PSO2	2,3,4,5, 11, 12	5	

List of Experiments:

WEEK 1: Write a program to represent disjoint subsets and perform Find and Union operations on the sets.

WEEK 2: **Write a program to implement the following string pattern-matching algorithms:**

i) Knuth–Morris–Pratt (KMP) ii) Boyer–Moore

WEEK 3: Write a program that implements the following Comparison based Sorting methods to sort a given list of integers in ascending order.

i) Merge Sort ii) Quick Sort

WEEK 4: Write a program that implements the following Non-Comparison based Sorting methods to sort a given list of integers in ascending order.
i) Radix sort ii) bucket sort

WEEK 5: **Write a program to implement the following searching techniques:**
i) Jump Search ii) Fibonacci Search

WEEK 6: Write a program to implement AVL trees

WEEK 7: Write a program to implement B Trees (2,3)

WEEK 8: Write programs to implement Binary Heap data structures and analyze their operations

WEEK 9: Write programs to implement the Binomial Heap data structures and analyze their operations

WEEK 10: Write a program to implement the following types of tries:
i) Binary Trie ii) Standard Trie

WEEK 11: Write a program to implement the following Hash Functions:
i) Division Method ii) Multiplication Method iii) Mid-square Method iv) Folding Method

WEEK 12: Write a program to implement the following collision resolution techniques and display the hash table after each insertion:
i) Linear Probing ii) Quadratic Probing iii) Double Hashing iv) Separate Chaining

TEXTBOOKS:

1. Data Structures: A Pseudocode Approach with C, R. F. Gilberg and B.A.Forouzan2nd Edition, Cengage Learning
2. Data Structure using C– Reema Thareja, 3rd Edition, Oxford University Press.

REFERENCE BOOKS:

1. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, 1st Edition, PHI/Pearson Education.

25CS21L02- OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

B-TECH- CSE: II Year –I Sem

Prerequisite(s): None

L	T	P/D	C
-	-	2/-	1

Course Objectives

CO1: Understand the principles of object-oriented programming and Java basics, including data types, operators, control statements, and class/object concepts.

CO2: Learn and apply inheritance, packages, interfaces, and polymorphism for building modular and reusable applications.

CO3: Develop robust programs by implementing exception handling and multithreading concepts in Java.

CO4: Utilize advanced Java features such as generics, lambda expressions, I/O streams, and collection framework for efficient data management.

CO5: Design and implement interactive GUI-based applications using AWT, Swing, event handling, and layout managers.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Mapped Experiments	BTL	Related SDGs
CO1	Explain and apply OOP concepts, Java syntax, control structures, classes, objects, constructors, methods, recursion, and String operations.	1,2,3,4,5	Week1, Week2, Week3	2	4,8,9,10
CO2	Demonstrate inheritance, polymorphism, packages, and interface usage to create well-structured and maintainable Java programs.	1,2,3,4,5	Week4, Week5, Week6	3	
CO3	Implement exception handling and multithreading mechanisms to develop efficient, safe, and concurrent Java applications.	1,2,3,4,5	Week7, Week8, Week9	4	
CO4	Apply generics, lambda expressions, Java I/O streams, and collections to handle and process data effectively in applications.	1,2,3,4,5	Week10, Week11, Week12	3	
CO5	Design and develop GUI-based applications using AWT/Swing components, event handling, and layout managers for interactive interfaces.	1,2,3,4,5	Week13, Week14	5/6	

Week 1

1.Design and implement a Library Book Catalog System using OOP concepts in Java.
You must create a class **Book** with the following specifications:

- **Attributes:** title, author, publisher, year, isbn, price
- **Constructors Required:** Default constructor, Parameterized constructor, Copy constructor
- **Methods to Define:**
  displayDetails(), matches(String keyword) → return true if keyword matches title/author, applyDiscount(double percentage)

Write a main program to Create an array of Book objects, Search for books by author/title, Display books

2.Develop a BankAccount Management application in Java using classes and objects.
Create class **BankAccount** with:

- **Attributes:**accountHolder, accountNumber, balance, accountType
- **Constructor(s):** Parameterized constructor initializing account details
- **Methods:**deposit(double amount), withdraw(double amount) (deny withdrawal if insufficient funds), checkBalance(), calculateInterest(float rate) → return interest

Write a driver program to simulate multiple account operations.

Week 2

3.Create a Java program to maintain hospital patient records.
Define a class **Patient** with:

- **Attributes:**patientId, name, age, problem, doctorAssigned, feePaid
- **Constructor:** Parameterized (use this keyword to initialize)
- **Methods:**displayReport(), updateDoctor(String doctor), addFee(double amount)

Write a menu-driven main program to register patients, update records, display patients.

4. Write a Java program to compute total, average & grade of students.
Class **Student** must contain:

- **Attributes:** name, rollNo, marks[], total, average
- **Constructor:** Accept rollNo, name and marks list dynamically
- **Methods:**computeTotal(), computeAverage(), grade() → return letter grade, displayResult()

Generate results of 10 students using array of objects.

Week 3

5.Design and implement a Java program to generate monthly salary slips using **inner classes**.You must define an **Outer Class Employee** and an **Inner Class SalaryBreakup**, where the inner class accesses and calculates salary components using the outer class data.

Outer Class: Employee

Attributes:intid, String name, double baseSalary, String department

Constructors Required: Default constructor, Parameterized constructor using this(id, name, baseSalary, department)

Methods to Implement:displayEmployeeDetails(), calculateSalary(), generateSlip()

Inner Class: SalaryBreakup

Attributes:doublehra, double da, double pf, double tax, double netPay

Methods to Implement:computeBreakup(), printBreakup()

6. Design and implement a Java program to simulate an ATM PIN verification system using string comparison techniques.
Class: ATMPinChecker
Attributes:String username, String pinStored
Methods to Implement:booleanverifyPIN(String enteredPIN), void authenticate()
Write a driver program to display a message "PIN ENTERED CORRECT" if successful pin entered within 3 attempts else should display "ACCOUNT LOCKED".

Week4
7. Write a Java program for a vehicle rental company using hierarchical inheritance.
Base Class: Vehicle
Derived Classes: Car, Bike, SUV
Attributes:

- **Vehicle**: vehicleNo, brand, baseFare
- **Car**: noOfSeats, acAvailable
- **Bike**: helmetRequired, geared
- **SUV**: fourWheelDrive, luxuryLevel

Methods:

- calculateFare(int days) → overridden in each subclass(Different fare per day for each type of vehicle)
- displayBill().

8.Create a payroll system using inheritance & method overriding.
Classes: Employee → Manager, Developer, Intern
Attributes:

- **Employee**: id, name, baseSalary
- **Manager**: bonus
- **Developer**: projectAllowance
- **Intern**: stipend

Methods:

- calculateSalary() → overridden for each subclass
- displayPaySlip()→ displays no.of days present and net salary

Week5
9.Develop a Java program that implements a digital payment system using interfaces and runtime polymorphism. You must create an **interface Payment** containing void pay (double amount), void refund (double amount) methods. Implement the interface in the following classes with given attributes and appropriate validation logic:
UPI Payment: upiId, upiPin, balance (Payment allowed only if PIN matches and balance ≥ amount)
CardPayment: cardNumber, cardHolder, cvv, limit (Payment allowed only if limit ≥ amount)
WalletPayment: walletId, walletBalance (Payment allowed only if wallet balance ≥ amount)
All classes must override pay(double amount) to deduct amount and refund() to add back amount. Write a Test Class to create objects for all payment types and display payment success or failure messages accordingly.

10. Write a program implementing a multi-package modular scientific calculator.
Classes Required:

1. calc.basic → add(), subtract(), multiply(), divide()
2. calc.advanced → power(), sqrt(), log()
3. print.display → Result formatting class

Create a **menu-driven Test class** in the default package to display operation list to user, call the appropriate method from the corresponding package and print results using class "display".

Week6
11. Create a pluggable media player.
Abstract Class: MediaFile
Attributes:fileName, duration, fileSize
Abstract Methods:play(), pause(), stop()
Derived Classes: AudioFile, VideoFile, LiveStream
Implement overridden methods with messages. Demonstrate polymorphism using MediaFile reference.

12. Develop an area calculator app using interface and runtime binding.
Interface: Shape → methods: area(), display()
Classes with attributes: Circle-radius, Rectangle-length, width, Triangle-base, height
Create various objects using Shape references for the shapes and perform operations dynamically.
Week7
13. Write a program that accepts two integers from user and performs division.
Use **try & multiple catch** to handle:

- ArithmeticException (division by zero)

InputMismatchException (invalid number input)
Display appropriate error message for each.
14. Create two threads, where:

- Thread-A prints **even numbers** up to 10

Thread-B prints **odd numbers** up to 10
Both must run concurrently and display results alternately. (Use sleep()to visualize the threads execution)

Week8
15. Develop an ATM withdrawal simulation with balance validation.
Class:ATMAccount
Attributes:accountNumber, holderName, balance
Methods Required:

- withdraw(double amount) throws InsufficientFundsException
- deposit(double amount)
- checkBalance()

Generate **InsufficientFundsException** and handle in a try-catch block.
16. Create a program to process student results from user input.
Exception Conditions:

- Marks < 0 or >100 → throw **InvalidMarksException**
- If total subjects entered = 0 → throw **DivideByZeroException**

Compute **total, average and grade** only when no exception occurs.

Week9

17. Implement a continuous chat exchange between two threads.
Classes: Sender, Receiver
Use wait(), notify(), or notifyAll() for communication.
Simulate message sending/receiving in alternate fashion like WhatsApp chat.
18. Build a **producer-consumer** order system.
Producer: Adds orders to queue
Consumer: Processes orders one-by-one
Use synchronized queue and thread communication to prevent overfill/underflow.

Week10

19. Build a **Course Feedback Form UI**.
Components Required:

- Rating from 1 to 5 for different questions using **JRadioButton**
- Suggestions/Commentsfield using **JTextArea**
- Submit**Button** → print feedback summary either in the console/the window itself
- Clear**Button** → reset fields

20. Create a Java Swing application to design a **Student Registration Form**.
The form should contain the following components:
JTextField: Name, Email, Contact, Address
JComboBox: Gender (Male/Female/Other)
JCheckBox: Course Selection (e.g., Java/Python/Data Science/AI)
JRadioButton:Mode of Learning (Online/Offline)
JButton: Submit, Reset, Exit

Week11

21. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts.
22. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the+, -,*, /, % operations. Add a text field to display the result.

Week12

23. Write a java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired.
24. Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. 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 Display the exception in a message dialog box.

Week13

25. Create a billing system with generic menu items.
Generic Class: MenuItem<T1,T2> (itemName, price, category)
Task:Use ArrayList<MenuItem> to store items and Compute bill dynamically.

Method:generateBill(List<MenuItem>)
Apply discount using lambda if total > 1500.
26. Create a quiz scoring leaderboard.
Class: Player with attributes name, score, timeTaken
Data Structure: ArrayList<Player> players
Task: Sort players using **Comparator**-Highest score first and if tie → lesser timeTakencomes first
Methods

- displayTop() - Display Top-5 performers
- search() - Search by player name &return their score if found else return "No such Player".
- delete() –Delete a player by name &return "SUCCESS" if found else return "No such Player".

Week14
27. Design a job scheduling system.
Class: **Task**with attributes as taskId, runTime, priority
Requirements:

- Higher priority jobs execute first
- Use **Comparator** to rank tasks
- Print execution order with runTime

28. Develop a Student Marks Database using Collections.
Specifications:

- Use HashMap<Integer, Double> → rollNo → percentage.
- **Methods**:
  - addRecord(int roll,doublepercentage)
  - displayAllSorted() using **TreeMap**
  - searchByRoll(int roll)
- Display topper list & fail list (< 40%).

25CS21L03-DATABASE MANAGEMENT SYSTEMS LAB

B-TECH- CSE: II Year –I Sem

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): 25CS12001: DATA STRUCTURES

Course Objectives

Develop ability to

1. Learn and practice data modeling using entity-relationship and develop database design.
2. Understand the features of data base management systems and Relational database.
3. Understand Structured Query Language(SQL) and learn SQL syntax.
4. Understand normalization process of a logical data model and correct any anomalies.
5. Understand needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Design simple database using ER modeling and apply additional features of ER model.	PO1, PO2, PO3, PO5, PO6, PO11, PSO1	I	2	4,5,8,9,12
CO2	Apply constraints on a relational model, querying using relational algebra and normalization techniques to arrive at a minimally redundant database.	PO1, PO2, PO3, PO4, PO5, PO11, PSO1	II	3	
CO3	Apply SQL querying and distinguish SQL and NoSQL databases.	PO1, PO2, PO3,PO5, PSO1, PSO3	III	4	
CO4	Apply concepts of concurrency control and data recovery in database transactions.	PO1, PO2, PO3, PO4, PO5, PO6, PO11, PSO1	IV	4	
CO5	Apply indexing techniques to organize the data on the secondary storage devices enabling efficient data retrieval.	PO1, PO2, PO3, PO5, PO11, PSO1, PSO3	V	5	

LIST OF EXPERIMENTS

Week-1
E-R Model. Analyze the problem with the entities which identify data persisted in the database which contains entities, attributes.

Week-2
Concept design with E-R Model. Apply cardinalities for each relationship; identify strong entities and weak entities for relationships like generalization, aggregation, specialization.

Week-3
Relation Model. Represent attributes as columns in tables and different types of attributes like Composite, Multi-valued, and Derived. Apply Normalization.

Week-4
Installation of MySQL and Queries using DATA DEFINITION LANGUAGE(DDL) COMMANDS- Create, Alter, Drop, Truncate

Week-5
Data Manipulation Language (DML) COMMANDS- SELECT, INSERT, UPDATE, DELETE

Week-6
Data Control Language (DCL)- GRANT, REVOKE Transaction Control Language (TCL) COMMANDS - COMMIT, ROLL BACK SAVE POINT

Week-7
In Built Functions - DATE FUNCTION, NUMERICAL FUNCTIONS, CHARACTER FUNCTIONS, CONVERSION FUNCTION.
Querying Using aggregate functions COUNT, SUM, MIN, MAX, AVG, GROUPBY and HAVING.

Week-8
Querying. Queries using ANY, ALL, IN, INTERSECT, UNION

Week-9
Querying. NESTED QUERIES AND JOIN QUERIES. Nested Queries, Correlated sub queries, Simple Join, a) Equi-join b) Non Equi-join, Self join , Outer Join

Week-10
Set Operators. Union, Union all, Intersect, Minus.
Views. Creating and dropping view

Week-11
Triggers. Creation of INSERT TRIGGER, DELETE TRIGGER, UPDATE TRIGGER

Week-12
Procedures. Creation, Execution and Modification of stored Procedure

Week-13

a) Write a program in MongoDB to create a database and a collection named students.
b) Insert single and multiple documents into the Students collection containing Roll Number, Name, Branch, and CGPA.
c) Write queries to display all documents and filter records using conditions (Example: CGPA > 8).
d) Perform update and delete operations on a document in the collection.
e) Write a MongoDB program to sort records and count the number of documents.

Week-14

a) Create aKeyspace named college and switch to it.
b) Create a table,students with the fields: *roll (Primary Key), name, branch,* and *CGPA*.
c) Insert multiple records into the table and display all the rows.
d) Write CQL queries to update and delete a particular student record.
e) Alter the table to add a new column email, and drop the table afterward.

25CS21SD1- DATA VISUALIZATION USING PYTHON

B-TECH- CSE: II Year –I Sem

L	T	P/D	C
-	-	2/-	1

Prerequisite(s): NIL

Course Objectives:

- Effective use of Python to generate Data Visualization.
- To discern patterns and relationships in the data.
- To build Dashboard applications.
- To communicate the results clearly and concisely.
- To be able to work with different formats of data sets.

Course Outcomes: At the end of the course a student would be able to

- Analyze different types of data and data sources, apply visualization fundamentals, and create basic visualizations using Python.
- Connect and process datasets of various formats using Python, and apply aggregate functions and custom calculations for data analysis.
- Perform data manipulation, sorting, filtering, pivoting, and formatting operations using NumPy and Pandas to improve data representation.
- Develop advanced and customized visualizations using Matplotlib and Seaborn, and demonstrate sharing, printing, and exporting of results.
- Create interactive and spatial visualizations by designing custom charts and implementing map-based visualizations using the Folium library.

Lab Problems:

Week 1: Identify suitable data sources, load the data into Python, and generate a basic visualization by applying fundamental visualization principles.

Week 2: Getting started with Python Software using Data file formats, connecting your Data to Python IDE, creating basic charts like line, bar charts, Tree maps etc.,.

Week 3: Using Python , Overview of SUM, AVG, and Aggregate features, Creating custom calculations and fields.

Week 4: Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.

Week 5: Data Manipulation, Axis Formatting, and Pivot Operations Using Python.

Week 6: Structuring Data and Performing Sorting and Filtering Using NumPy and Pandas.

Week 7: Advanced Data Visualization Techniques Using Matplotlib.

Week 8: Creating Waffle Charts and Word Clouds Using Python.

Week 9: Data analysis using Seaborn visualizations with sharing, printing, exporting, and integration of spatial visualization.

Week 10: Creating custom charts and working with Folium Library for interactive map-based Visualization.

Text Books:

1. Core Python Programming by Dr R. Nageswara Rao, Dreamtech Press.
2. Python for Data Analysis, Data Wrangling with Pandas and Mumpy and Juypter by Wes McKinney - OREILLY Publications

25CE21VA1-ENVIRONMENTAL SCIENCE

B-TECH- CSE: II Year –I Sem

L	T	P/D	C
1	-	-/-	1

Prerequisite(s): NIL

Course Objectives

1. Understanding the importance of ecological balance for sustainable development.
2. Understanding the impacts of developmental activities and mitigation measures.
3. Understanding the environmental policies and regulations

CO	Course Outcomes	Related POs & PSOs	Related Units	BTL	Related SDGs
CO1	Explain the structure, functions, and significance of ecosystems, energy flow, biogeochemical cycles, and ecological balance.	**POs:** 1, 2 **PSOs:** 1	I	2	SDG 4, SDG 15
CO2	Analyze the utilization and conservation of natural resources, including water, land, forest, mineral, and energy resources, with reference to sustainability.	**POs:** 1, 2, 7 **PSOs:** 1	II	3, 4	SDG 6 , SDG 7, SDG 12
CO3	Explain the importance of biodiversity, identify threats to biodiversity, and describe conservation strategies including in-situ and ex-situ methods.	**POs:** 1, 2, 7 **PSOs:** 1	III	2, 3	SDG 15, SDG 14
CO4	Analyze different types of environmental pollution and evaluate pollution control technologies and global environmental initiatives.	**POs:** 1, 2, 4, 7 **PSOs:** 1,3	IV	3, 4	SDG 3, SDG 11, SDG 13
CO5	Explain environmental policies, legislation, EIA processes, and sustainable development concepts to address environmental challenges.	**POs:** 1, 6, 7, 8 **PSOs:** 1	V	2, 3	SDG 11, SDG 12, SDG 13

UNIT - I

Ecosystems: Definition, Scope, and Importance of ecosystem. Classification, structure, and function of an ecosystem, Food chains, food webs, and ecological pyramids. Flow of energy, Biogeochemical cycles, Bioaccumulation, Biomagnification, ecosystem value, services and carrying capacity, Field visits.

UNIT - II

Natural Resources: Classification of Resources: Living and Non-Living resources, water resources: use and over utilization of surface and ground water, floods and droughts, Dams: benefits and problems. Mineral resources: use and exploitation, environmental effects of extracting and using mineral resources, Land resources: Forest resources, Energy resources: growing energy needs, renewable and non-renewable energy sources, use of alternate energy source, case studies.

UNIT - III

Biodiversity and Biotic Resources: Introduction, Definition, genetic, species and ecosystem diversity.Value of biodiversity; consumptive use, productive use, social, ethical, aesthetic and optional values. India as a mega diversity nation, Hot spots of biodiversity. Field visit. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; conservation of biodiversity: In-Situ and Ex-situ conservation. National Biodiversity act.

UNIT - IV

Environmental Pollution and Control Technologies: Environmental Pollution: Classification of pollution, Air Pollution: Primary and secondary pollutants, Automobile and Industrial pollution, Ambient air quality standards. Water pollution: Sources and types of pollution, drinking water quality standards. Soil Pollution: Sources and types, Impacts of modern agriculture, degradation of soil. Noise Pollution: Sources and Health hazards, standards, Solid waste: Municipal Solid Waste management, composition and characteristics of e-Waste and its management. Pollution control technologies: Wastewater Treatment methods: Primary, secondary and Tertiary. Overview of air pollution control technologies, Concepts of bioremediation. Global Environmental Issues and Global Efforts: Climate change and impacts on human environment. Ozone depletion and Ozone depleting substances (ODS). Deforestation and desertification. International conventions / Protocols: Earth summit, Kyoto protocol, and Montréal Protocol. NAPCC-GoI Initiatives.

UNIT - V

Environmental Policy, Legislation & EIA: Environmental Protection act, Legal aspects Air Act-1981, Water Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, biomedical waste management and handling rules, hazardous waste management and handling rules. EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socioeconomical aspects. Strategies for risk assessment, Concepts of Environmental Management Plan (EMP). Towards Sustainable Future: Concept of Sustainable Development Goals, Population and its explosion, Crazy Consumerism, Environmental Education, Urban Sprawl, Human health, Environmental Ethics, Concept of Green Building, Ecological Foot Print, Life Cycle assessment (LCA), Low carbon life style.

TEXT BOOKS:

1. Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission.
2. Environmental Studies by R. Rajagopalan, Oxford University Press.

REFERENCE BOOKS:

1. Environmental Science: towards a sustainable future by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
2. Environmental Engineering and science by Gilbert M. Masters and Wendell P. Ela. 2008 PHI Learning Pvt. Ltd.
3. Environmental Science by Daniel B. Botkin & Edward A. Keller, Wiley INDIA edition.
4. Environmental Studies by Anubha Kaushik, 4th Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications.
6. Introduction to Environmental Science by Y. Anjaneyulu, BS. Publication.

25CS22001–ALGORITHM DESIGN AND ANALYSIS

B. Tech.CSE -II Year II Sem.

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): 25CS12001: Data Structures, 25CS21001:Advanced Data Structures

Course Objectives: Develop an ability to

1. Develop proficiency in evaluating algorithms using asymptotic notations, including best-, average-, and worst-case time/space complexities, and solving related recurrence relations.
2. Master various algorithmic strategies—divide-and-conquer, greedy, dynamic programming, backtracking, and branch-and-bound—identifying suitable use cases and demonstrating their application.
3. Critically assess and contrast different algorithms in terms of efficiency, scalability, and correctness through rigorous analytical reasoning and empirical evaluation.
4. Differentiate between tractable (polynomial-time) and intractable (super-polynomial or exponential-time) problems;
5. Identify and classify problems as P, NP, NP-hard, or NP-complete, and assess their relationships through polynomial-time reductions and Cook's theorem. Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Analyze time and space complexity of a given algorithm and use asymptotic notation to represent the complexities.	PO1, PO2, PO11, PSO2	I	4	4,9
CO2	Apply divide and conquer, greedy, dynamic programming, backtracking and branch and bound algorithmic approaches to a given problem.	PO1, PO2,PO3, PO4, P10, PO11,PSO2	I,II,III,IV	3	
CO3	Prove that a given problem is NP-Complete by using the concepts of NP-Completeness.	PO1, PO2, PO4, PO10, PO11, PSO2	V	5	
CO4	Apply lower-bound theory using comparison trees for ordered searching, sorting, and selection	PO1, PO2, PO4, PO11, PSO2	V	3	

UNIT - I

Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation, and Little oh notation.

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.

UNIT - II

Greedy method: General method, applications- knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem (Dijkstra's), Huffman coding

Non-recursive binary tree traversals, Connected components, Biconnected components, Topological sort, strongly connected components

UNIT - III

Dynamic Programming: General method, applications- Optimal binary search tree, 0/1 knapsackproblem, matrix-chain multiplication, Longest common subsequence, Single source shortest path problem(Bellman-Ford), All pairs shortest path problem(Floyd-Warshall), Traveling salesperson problem.

UNIT - IV

Backtracking: General method, applications, n-queens problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

Branch and Bound: General method, applications - Travelling salesperson problem, 0/1 knapsackproblem - LC Branch and Bound solution, FIFO Branch and Bound solution.

UNIT – V

Lower Bound Theory: Comparison trees for ordered searching, sorting and selection

NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem, NP-hard graph problems: Clique Decision Problem, Node-Cover Decision Problem

TEXT BOOKS:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni, and Rajasekaran, University Press.

REFERENCE BOOKS:

1. Design and Analysis of algorithms, Aho, Ullman, and Hopcroft, Pearson education.

2. Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C.Stein, PHI Pvt. Ltd./ Pearson Education.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley and Sons.

25CS22002-COMPUTER ORGANIZATION AND ASSEMBLY LANGUAGE PROGRAMMING

II Year. B.Tech. CSE – II Sem

L	T	P/D	C
3	-	-/-	3

Prerequisite(s): 20EC21002-DIGITAL DESIGN

Course Objectives

Develop ability to

1. To introduce principles of computer organization and the basic architectural concepts.
2. Recommend instruction formats, addressing modes, micro instructions for design of control unit
3. Write assembly level programs using 8086 microprocessor.
4. Understand the I/O and memory organizations of a Computer system
5. Recognize different parallel processing architectures

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Explain various aspects of the stored program concept of computer architecture and organization.	PO1, PO2, PO3, PO4, PO11, PSO1, PSO2	I	2	4,7,8,9,12
CO2	Develop micro programs towards the design of Microprogrammed Control Unit.	PO1, PO2, PO3, PO4, PO8, PO11, PSO1, PSO2	II	3	
CO3	Use 8086 microprocessor architecture, develop assembly language programs.	PO1, PO2, PO3, PO4 PO5, PO11, PSO1, PSO2	III	3	
CO4	Compare various input-output and memory Organization Techniques.	PO1, PO2, PO3, PO4, PO11, PSO1, PSO2	IV	4	
CO5	Differentiate various parallel processing architectures.	PO1, PO2, PO3, PO4, PO11, PSO1, PSO2	V	4	

UNIT - I

Introduction: Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language 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.
Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer Instructions, Instruction cycle.

UNIT - II
Micro programmed Control: Control memory, Address sequencing, micro program example, design of control unit.
Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Program Control, Program interrupts.

UNIT-III
Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms.
8086 Architecture: Register Organization of 8086, 8086 Architecture, Signal Description of 8086, Physical Memory Organization, Pipelining in 8086, 8086 Flag Registers, 8086 Instruction Set and Assembler Directives.

UNIT – IV
Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access.
Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Hit and Miss ratio, associative mapping, direct mapping, set associative mapping, Writing into cache.

UNIT - V
Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, CISC Characteristics, RISC Characteristics.
Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Inter processor arbitration, Inter processor communication and synchronization, Cache Coherence.

ARM Architecture, Advanced Processor Architectures (introductory Overview)- Superscalar Architecture, Pipelined Architecture, Multicore Architecture, VLIW (Very Long Instruction Word), GPU Architecture

TEXT BOOK(S)

1. Computer System Architecture, Morris Mano, 3rd Edition (Revised), Pearson Education, 2017, M. Morris Mano, 3/e, Pearson Education. (UNIT-1,2,4,5).
2. Advanced Micro Processor and Peripherals, Hall/A K Ray, McGraw Hill Education, 2006. (UNIT-3).

REFERENCE BOOK(S)

1. Computer Organization and Design: The Hardware/Software Interface (ARM Edition) John L. Hennessy and David A. Patterson, 1st Edition, Morgan Kaufmann (Elsevier), 2017

2. Structured Computer Organization, Andrew S. Tanenbaum and Todd Austin 6th Edition, Pearson Education, 2013Computer Organization – Carl Hamacher, ZvonksVranesic, SafeaZaky, Vth Edition, McGraw Hill.
3. Computer Organization and Architecture: Designing for Performance, William Stallings, 10th Edition, Pearson Education, 2016.

25CS22003: OPERATING SYSTEMS

B. Tech. CSE - II Year II Sem.

Prerequisite(s): 25CS11001:Programming for Problem Solving25CS12001:DATA STRUCTURES

L	T	P/D	C
3	-	-/-	3

Course Objectives: Develop ability to

1. Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
2. Introduce the issues to be considered in the design and development of operating system
3. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Course outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Explain operating system structures, services, system calls, and analyze basic process and thread management.	PO1, PO2, PO5, PO11, PSO1, PSO2	I	3	4,8,9, 12, 16
CO2	Apply CPU scheduling algorithms and evaluate deadlock conditions with suitable handling techniques.	PO1, PO2, PO3, PO4, PO5, PO11, PSO1, PSO2	II	3	
CO3	Analyze process synchronization issues and apply IPC mechanisms for coordinated process communication.	PO1, PO2, PO3, PO4, PO5, PSO1, PSO2	III	4	
CO4	Evaluate memory management and virtual memory techniques, including page replacement strategies.	PO1, PO2, PO3, PO4, PO5, PO11, PSO1, PSO2	IV	4	
CO5	Explain file system structures and perform file operations using system calls.	PO1, PO2, PO3, PO5, PO11 PSO1, PSO2	V	3	

UNIT-I

Operating System - Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls

Process - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

UNIT-II

CPU Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec

Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

UNIT-III

Process Management and Synchronization-The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors

Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT-IV

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

UNIT-V

File System Interface and Operations-Access methods, Directory Structure, Protection, FileSystem Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

TEXTBOOKS:

1. Operating System Principles-Abraham Silberchatz, Peter B.Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the UNIX environment, W.R.Stevens, Pearsoneducation.

REFERENCEBOOKS:

1. Operating Systems-Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach-Crowley,TMH.
3. Modern Operating Systems, Andrew S.Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/PearsonEducation
5. UNIX Internals-The New Frontiers, U.Vahalia, Pearson Education.

25CS22004-WEB TECHNOLOGIES

L	T	P/D	C
2	-	-/-	2

B. Tech. CSE - II Year II Sem.

Prerequisite(s): 25CS21002:Object Oriented Programming through Java

Course Objectives: Develop ability to

1. To understand web fundamentals including client-server architecture, HTML5 semantic structure, and CSS3 responsive styling.
2. To master modern JavaScript programming including JSON data representation, parsing, and exchange for dynamic client-side interactions.
3. To develop single-page applications using React framework with hooks and state management techniques.
4. To implement server-side logic using Java Servlets for request handling and database integration via JDBC.
5. To develop full-stack web applications using JSON as the standard data interchange format between client and server.

Course outcomes Mapping

CO	Course Outcomes	RelatedPOs and PSOs	Related Units	BTL	Related SDGs
CO1	Explain web architecture and develop responsive UIs using HTML5 semantics, CSS3 Flexbox/Grid, and client-side styling	PO1,2,3,4,11 PSO1,2	1	1,2,3	4,9
CO2	Apply ES6+ JavaScript features for DOM manipulation, event handling, asynchronous operations, and JSON-based client-side data handling.	PO1,2,3,4,5,11 PSO1,2	2	3,4	9
CO3	Design and implement React applications using components, hooks, routing, and state management	PO1,2,3,4,5,11 PSO1,2	3	3,4	9
CO4	Develop server-side applications with Java Servlets for lifecycle management, request processing, JSON-based client–server communication, session management, and JDBC CRUD operations.	PO1,2,3,4,5,11 PSO1,2	4	3,4	9
CO5	Build scalable Node.js/Express servers implementing middleware chains, REST CRUD routes using JSON and MongoDB, templating engines, static file serving, and error management.	PO1,2,3,4,5,8,11 PSO1,2	5	3,4,5	9

UNIT – I: Web Fundamentals & Client-side Styling

Internet basics: HTTP/HTTPS protocols, client-server architecture, DNS resolution, browsers and rendering engines.

HTML5: Semantic elements, forms, advanced input types, validation attributes, multimedia.

CSS3: Selectors, box model, positioning, Flexbox layout, CSS Grid, animations/transitions.

UNIT – II: JavaScript Programming

Core JavaScript (ES6+): Variables, data types, operators, control structures, functions, template literals, destructuring, spread/rest operators.

DOM manipulation: Selecting elements, modifying content/styles, creating/removing elements, event handling.

JSON Concepts: JSON syntax and structure, JSON objects and arrays, converting JavaScript objects to JSON and vice versa using JSON.stringify() and JSON.parse(), using JSON for client-side data handling.

Client-side storage: localStorage/sessionStorage, form validation.

UNIT – III: React Framework

React fundamentals: JSX syntax, creating functional/class components, props drilling, state management.

React Hooks: useEffect , useContext, useReducer, custom hooks, useRef.

Advanced React: Conditional rendering, lists/keys, forms, React Router, Handling and rendering **JSON data** in React components, mapping JSON responses to UI elements.

State management: Context API, basic Redux principles, performance optimization.

UNIT – IV: Server-side Development

Java Servlets: Servlet lifecycle, Servlet Deployment, HttpServlet, request/response objects, parameters, Servlet filters, session management, cookies.

Data Exchange: Using JSON format for client-server communication in servlets, generating JSON responses, handling JSON request data.

JDBC fundamentals: CRUD operations and converting database results into JSON responses for web applications.

UNIT – V: Server-Side Web Development with Node.js & Express.js

Node.js Basics: Architecture, event loop, modules, npm, simple HTTP servers. **Express.js Fundamentals**: Application structure, routing, request/response handling. **Middleware:** JSON body parsing, custom middleware for validating JSON requests, CORS handling.

REST API Development: Designing RESTful CRUD APIs using JSON as the data exchange format, MongoDB integration as the backend database, error handling, and testing using Postman / Thunder Client with JSON requests and responses.

TEXT BOOKS

1. Web Design with HTML, CSS, JavaScript and jQuery, *Jon Duckett*, Wiley Publications. *(UNIT – I, II)*

2. Eloquent JavaScript: A Modern Introduction to Programming, *Marijn Haverbeke*, No Starch Press. *(UNIT – II)*
3. Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, *Kirupa Chinnathambi*, Addison-Wesley. *(UNIT – III)*
4. Java for Web Development, *S. Dwarkadas*, BPB Publications. *(UNIT – IV)*
5. Web Development with Node.js and MongoDB, *Valentino Gagliardi / Packt Publishing*. *(UNIT – V)*

REFERENCE BOOKS

1. You Don't Know JS (Yet) **–** *Kyle Simpson*, O'Reilly Media.
2. Learning Web Design**,** *Jennifer Niederst Robbins*, O'Reilly.
3. Pro MERN Stack: Mongo, Express, React, Node**,** *Vasan Subramanian*, Apress**.**
4. Head First Servlets and JSP**,** *Bryan Basham, Kathy Sierra, Bert Bates*, O'Reilly.
5. Node.js Design Patterns – Mario Casciaro & Luciano Mammino, Packt Publishing.
6. Learning Node.js Development – Andrew Mead, Packt Publishing.
7. Express in Action: Writing, Building, and Testing Node.js Applications – Evan Hahn, Manning Publications.
8. Professional Node.js: Building JavaScript-Based Scalable Software – Pedro Teixeira, Wrox.
9. Beginning Node.js – Basarat Ali Syed, Apress.

25CS22005:AUTOMATA THEORY AND COMPILER DESIGN

B. Tech.CSE – II Year, II Sem.

- **Prerequisite(s):**
  25MA11001-Matrices and Calculus
  25CS12003-DISCRETE MATHEMATICS

L	T	P/D	C
3	-	-/-	3

Course Objectives: Develop ability to

1. Understand mathematical methods of computing devices called abstract machines namely finite automata, pushdown automata and turing machines.
2. Explain deterministic and non-deterministic machines.
3. Identify different formal language classes and their relationships.
4. Design grammars and recognizers for different formal languages.
5. Determine the decidability and intractability of computational problems and comprehend the hierarchy of problems arising in computer science.
6. Illustrate all the major phases in process of compilation.

Course Outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Build and analyze deterministic and nondeterministic finite automata, evaluate language acceptance.	PO1, PO2	I,II	3,4	4,9
CO2	Construct regular expressions and context-free grammars and categorize languages within the Chomsky hierarchy.	PO1, PO2	II,III	3,4	4,9
CO3	Design Pushdown automata and Turing Machines for various languages, and relate these abstract machines to the early phases of compilation.	PO1, PO2	III	2	4,9
CO4	Perform lexical and syntax analysis for high level programs.	PO1, PO2	IV	3	4,8,9
CO5	Apply code optimization techniques, construct DAGs, handle register allocation, and generate efficient machine-level or object-level code.	PO1, PO2	V	3	7,9,12

UNIT - I Fundamentals : Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automaton (DFA) and non deterministic finite automaton (NFA), transition diagrams and Language recognizers. Finite Automata: NFA to DFA conversion, minimization of FSM, equivalence between two FSM's, Finite Automata with output- Moore and Mealy machines.

UNIT - II Regular Languages: Regular sets, regular expressions, identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Context Free Grammars: Definition, Ambiguity in context free grammars. Simplification of Context Free Grammars. Chomsky normal form, Greibach normal form, Enumeration of properties of CFLs (proofs omitted), Chomsky's hierarchy of languages.

UNIT – III Push Down Automata: Push down automata, definition, model, acceptance of CFL, by finalstate and by empty stack.

Turing Machine: Turing Machine, definition, model, design of TM, counter machine, types of Turing machines (proofs not required).

Overview of Compilation: Phases of compilation-lexical analysis, regular grammar and regular expression for common programming language features, Pass and phases of translation, interpretation, bootstrapping, data structures in compilation

UNIT – IV Top Down Parsing: Back Tracking, LL(1), Recursive Descent Parsing, Predictive Parsing, Pre-processing steps required for predictive parsing.

Bottom Up Parsing: Shift Reduce Parsing, LR and LALR Parsing, Error Recovery in Parsing, Handling Ambiguous grammar.

Semantic Analysis: Intermediate forms of source programs-abstract syntax tree, polish notation and three address codes, conversion of popular programming languages language constructs into intermediate code forms.

UNIT – V Code Optimization: Consideration for optimization, scope of optimization, loop optimization, frequency reduction folding, DAG representation, reduction in strengths.

Object Code Generation: Object code forms, machine dependent code optimization, register allocation and assignment generic code generation algorithms.

TEXT BOOKS:

1. Introduction to Automata Theory Languages and Computation, Hopcroft H.E. and Ullman J. D. Pearson Education, 2002.

2. Aho, A. V., Lam, M. S., Sethi, R., & Ullman, J. D. (2007). Compilers: Principles, Techniques, and Tools, 2nd edition. Pearson Education.

REFERENCE BOOKS:

1. Introduction to Theory of Computation – Sipser 2nd edition Thomson, 2005.
2. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan Rama R, Pearson Education, 2009.
3. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley, 1991.
4. Theory Of Computation: A Problem - Solving Approach, Kavi Mahesh, Wiley India Pvt. Ltd, 2011.
5. Elements of Theory of Computation, Lewis H.P. &Papadimition C.H. Pearson/PHI, 1981.
6. Theory of Computer Science – Automata languages and computation -Mishra and Chandrashekaran, 2nd edition, PHI, 2006.

25CS22L01: ALGORITHM DESIGN AND ANALYSIS AND COMPILER DESIGN LAB

B. Tech.CSE – II Year, II Sem.
Prerequisite(s): 25CS11001:Programming for Problem Solving, 25CS12001:Data Structures,25CS21001:Advanced Data Structures

L	T	P/D	C
-	-	2/-	1

ALGORITHM DESIGN AND ANALYSIS LAB:

Course Objectives: Develop ability to

1. Develop proficiency in evaluating algorithms using asymptotic notations, including best-, average-, and worst-case time/space complexities, and solving related recurrence relations.
2. Master various algorithmic strategies—divide-and-conquer, greedy, dynamic programming, backtracking, and branch-and-bound—identifying suitable use cases and demonstrating their application.
3. Critically assess and contrast different algorithms in terms of efficiency, scalability, and correctness through rigorous analytical reasoning and empirical evaluation.

Course Outcomes Mapping

CO	Course Outcome (CO)	Related POs and PSOs	Related Experiments	BTL	Related SDGs
CO1	Design and implement efficient algorithms for searching, sorting, and graph processing, and analyze their performance with respect to time and space complexity.	PO1, PO2, PO4, PO11, PSO2	1, 2, 3	4	4,9,11
CO2	Apply classical graph algorithms such as Prim's, Kruskal's, Dijkstra's, and Floyd–Warshall to solve real-world problems involving minimum spanning trees and shortest paths.	PO1, PO2, PO4, PSO2	1, 2	3	
CO3	Develop optimal solutions using Dynamic Programming techniques, including Matrix Chain Multiplication and Optimal Binary Search Trees.	PO1, PO2, PO4, PO11, PSO2	4, 5, 6	5	
CO4	Use backtracking strategies to solve complex combinatorial problems such as Subset Sum, N-Queens, and Hamiltonian Cycles, and evaluate their computational behaviour.	PO1, PO2, PO4, PSO2	7	6	

List of Experiments:

WEEK 1: a) Implement Prim's Algorithm to find the Minimum Cost Spanning Tree (MCST) for a given graph.
b) Implement Kruskal's Algorithm to find the Minimum Cost Spanning Tree (MCST) for a given graph.

WEEK 2: From a given source vertex in a weighted connected graph, find the shortest paths to all other vertices using Dijkstra's Algorithm.

WEEK 3: Write and implement an algorithm to determine articulation points and biconnected components in a given graph using DFS.

WEEK 4: Implement the Matrix Chain Multiplication algorithm using Dynamic Programming.

WEEK 5: Implement an algorithm to construct the Optimal Binary Search Tree (OBST) for a given set of identifiers and probabilities using Dynamic Programming.

WEEK 6: Write a program to solve the All-pairs shortest paths problem using FloydWarshall algorithm for a given graph.

WEEK 7: a) Given a set $S = \{s_1, s_2, \ldots, s_n\}$ of positive integers and a target sum d, find all subsets whose sum equals d using backtracking.
Example: $S = \{1, 2, 5, 6, 8\}$, $d = 9 \rightarrow$ Solutions: {1, 2, 6}, {1, 8}. Display an appropriate message if no solution exists.
b) Implement the N-Queens Problem using Backtracking.

TEXT BOOK:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni, and Rajasekaran, 2nd edition, Universities Press.
2. Design and Analysis of algorithms, Aho, Ullman, and Hopcroft, Pearson education.
3. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 3rd edition, PHI Learning

COMPILER DESIGN LAB

Prerequisite(s): 25CS11001:Programming for Problem Solving,25CS12001:Data Structures, 25CS21001:Advanced Data Structures

Course Objectives: Develop ability to

1. Understand mathematical methods of computing devices called abstract machines namely finite automata, pushdown automata and turing machines.
2. Explain deterministic and non-deterministic machines.
3. Identify different formal language classes and their relationships.
4. Design grammars and recognizers for different formal languages.
5. Determine the decidability and intractability of computational problems and comprehend the hierarchy of problems arising in computer science.
6. Illustrate all the major phases in process of compilation.

Course Outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Build and analyze deterministic and nondeterministic finite automata, evaluate language acceptance.	PO1, PO2	I,II	3,4	4,9
CO2	Construct regular expressions and context-free grammars and categorize languages within the Chomsky hierarchy.	PO1, PO2	II,III	3,4	4,9
CO3	Design Pushdown automata and Turing Machines for various languages, and relate these abstract machines to the early phases of compilation.	PO1, PO2	III	2	4,9
CO4	Perform lexical and syntax analysis for high level programs.	PO1, PO2	IV	3	4,8,9
CO5	Apply code optimization techniques, construct DAGs, handle register allocation, and generate efficient machine-level or object-level code.	PO1, PO2	V	3	7,9,12

List of Experiments:

1. Develop a lexical analyzer to recognize a few patterns inc (ex. Identifiers, constants, comments, operators etc.)
2. Implement the lexical analyzer using JLex, flex or other lexical analyzer generating tools.
3. Design predictive parser for the given language.
4. write recursive descent parser for the grammar E->E+T E->T T->T*F T->F F->(E)/id.
5. Write a C program to calculate first function for the grammar E->E+T E->T T->T*F T->F F->(E)/id
6. Write a YACC program to implement a top down parser for the given grammar.
7. Convert the bnf rules into yacc form and write code to generate abstract syntax tree.

25CS22L02: OPERATING SYSTEMS AND ASSEMBLY LEVEL PROGRAMMING LAB

B.Tech. II Year, II Sem.

L	T	P/D	C
-	-	2	1

Prerequisites: 25CS11001:Programming for Problem Solving
25CS12001:Data Structures, 25CS21001:Advanced Data Structures

Course Objectives:

1. To provide an understanding of the design aspects of operating system concepts through simulation.
2. Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix.

Course outcomes (COs): At the end of the course, student would be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management	PO1,PO2,PO3,PO4,PO11, PSO1,PSO2	I-III(OS)	3	4, 7, 8, 9, 12
CO2	Able to implement C programs using Unix system calls	PO1,PO2,PO3,PO4,,PO8, PO11, PSO1,PSO2	IV-V (OS)	4	
CO3	Explain computer architecture, stored-program concepts, and I/O–memory organization	PO1,PO2,PO3,PO4,,PO5, PO11, PSO1,PSO2	I-III(ALP)	3	
CO4	Develop micro programs and understand micro programmed control unit design.	PO1,PO2,PO3,PO4, PO11, PSO1,PSO2	IV(ALP)	4	
CO5	Write 8086 assembly programs and distinguish parallel processing architectures.	PO1,PO2,PO3,PO4, PO11, PSO1,PSO2	V(ALP)	4	

List of Experiments:

Operating Systems and Assembly Language Programming Lab

Week 1:

Write C programs to simulate the following CPU Scheduling algorithms a) FCFS
b) SJF c) Round Robin d) priority

Week 2:

a. Architecture of 8086 microprocessor

b. Instruction Set of 8086 microprocessor

Week 3:
Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, lseek, stat, fork, exit)

Week 4:
a. Write a program to display string "Computer Science and Engineering".
b. Write an Assembly Language Program (ALP) to display multiple strings line by line.
c. Write an Assembly Language Program (ALP) to find the maximum of three numbers

Week 5:
Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.

Week 6:
a. Write an Assembly Language Program (ALP) to print numbers from 0 to 9
b. Write an Assembly Language Program (ALP) to check whether a given number is even or odd.

Week 7:
Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.

Week 8:
a. Write an Assembly Language Program (ALP) to find the factorial of a number.
b. Write an Assembly Language Program (ALP) to print fibo series up to 5 numbers.

Week 9:
Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues d) Shared Memory

Week 10:
a. Write an Assembly Language Program (ALP) to take n values from user and calculate their sum.(BL contains the result)
b. Write an Assembly Language Program (ALP) to take n values from user and calculate maximum and minimum values

Week 11:
Write C programs to simulate the following memory management techniques a) Paging b) Segmentation

Week 12:

a. Write 8086 Assembly Language Program (ALP) to transfer a block of data from one location to another.

b. Write an Assembly Language Program (ALP) to reverse the given string.

c. Write an Assembly Language Program (ALP) to perform addition of two 2X2 matrices.

Week 13:

Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal

Week 14:

a. Write an Assembly Language Program (ALP) for linear search.

b. Write an Assembly Language Program (ALP) to take n values from user and sort them in ascending order.

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced programming in the Unix environment, W. R. Stevens, Pearson education.
3. Advanced Micro Processor and Peripherals, Hall/A K Ray, McGraw Hill Education, 2006.

25CS22L03: WEB TECHNOLOGIES LAB

B.Tech. II Year, II Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisites: 25CS21002: OBJECT ORIENTED PROGRAMMING

Course Objectives:

1. To understand web fundamentals including client-server architecture, HTML5 semantic structure, and CSS3 responsive styling.
2. To master modern JavaScript programming for dynamic client-side interactions and DOM manipulation.
3. To develop single-page applications using React framework with hooks and state management techniques.
4. To implement server-side logic using Java Servlets for request handling and database integration via JDBC.
5. To develop full-stack server-side web applications using Node.js fundamentals, Express.js routing/middleware, static content serving, EJS/Pug templating, and RESTful CRUD APIs with Postman testing.

Course outcomes Mapping

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Explain web architecture and develop responsive UIs using HTML5 semantics, CSS3 Flexbox/Grid, and client-side styling	PO1,2,3,4,11 PSO1,2	1	1,2,3	4,9
CO2	Apply ES6+ JavaScript features for DOM manipulation, event handling, asynchronous operations, and JSON-based client-side data handling.	PO1,2,3,4,5,11 PSO1,2	2	3,4	9
CO3	Design and implement React applications using components, hooks, routing, and state management	PO1,2,3,4,5,11 PSO1,2	3	3,4	9
CO4	Develop server-side applications with Java Servlets for lifecycle management, request processing, JSON-based client–server communication, session management, and JDBC CRUD operations.	PO1,2,3,4,5,11 PSO1,2	4	3,4	9
CO5	Build scalable Node.js/Express servers implementing middleware chains, REST CRUD routes using JSON and MongoDB, templating engines, static file serving, and error management.	PO1,2,3,4,5,8,11 PSO1,2	5	3,4,5	9

List of Experiments:

Week 1

1. Create a simple HTML personal profile page using semantic tags and embed audio/video elements.
2. Develop a responsive navigation bar using only CSS, including a hamburger menu for mobile view.
3. Design a registration form using HTML5 form validation: required fields, patterns, min/max constraints, and custom error messages.

Week 2

1. Build a CSS box-model demonstration page showing how padding, margin, and border affect element dimensions.
2. Create an animated button set using CSS transitions and transforms for hover, focus, and active states.
3. Develop a small personal portfolio webpage using external CSS and a responsive hero section.

Week 3

1. Write a JavaScript-based calculator to perform addition, subtraction, multiplication, and division with proper input validation.
2. Build a dynamic list application using DOM manipulation where users can add/remove items, and store and retrieve data in localStorage using JSON format.
3. Create an event-driven image carousel with previous/next controls and auto-play using setInterval and clearInterval.

Week 4

1. Use JavaScript array methods to process student marks and represent input and output data using JSON objects and arrays.
2. Implement a form handling script that validates user input and displays inline error messages dynamically.
3. Develop a number-guessing game with limited attempts, score tracking, and responsive feedback messages.

Week 5

1. Build a React component tree consisting of header, footer, and content components using props for communication.
2. Create a Todo application in React using useState to add, remove, and complete tasks.
3. Develop a controlled user profile form in React with state-based validation.

Week 6

1. Use the useRef hook to auto-focus an input field on mount and measure element dimensions.
2. Create a counter application using useReducer with increment, decrement, and reset actions.
3. Implement a theme switcher (dark/light mode) using React Context API.

Week 7

1. Develop a HelloServlet that responds with a formatted HTML page and a JSON response for client-side consumption.
2. Create a contact form servlet that reads form parameters and returns validation results in JSON format.
3. Implement a cookie demo servlet to set, read, and expire cookies across multiple requests.

Week 8

1. Write a JDBC program to connect to a MySQL database and convert database query results into JSON responses.
2. Implement JDBC insert, update, and delete operations and send operation status as JSON data.

Week 9

1. Create a servlet filter that logs incoming requests and blocks requests missing a required header.
2. Create a file upload servlet to accept multipart/form-data and save uploaded files to the server.

Week 10

1. Installing Node.js and Running Basic JavaScript Programs
2. Creating a Simple HTTP Server Using Node.js Core ‘http’ Module
3. Understanding Node.js Event Loop and Asynchronous Operations

Week 11

1. Build a basic Express.js application with GET and POST routes that accept and return JSON request and response bodies.
2. Implement middleware for JSON request parsing and validation.
3. Serving Static HTML, CSS, JS Files Using Express Static Middleware

Week 12

1. Creating Dynamic Pages Using EJS or Pug Templating Engine.
2. Develop REST APIs with CRUD endpoints in Express.js using MongoDB as the backend database and JSON as the data exchange format.
3. Test REST APIs using Postman / Thunder Client with JSON-based requests and responses.

25CS22SD2:NODE JS/REACT JS

B.Tech. II Year II Sem.

L	T	P/D	C
-	-	2/-	1

Prerequisite(s):(25CS21002) Object Oriented Programming through Java, HTML Basics

Course Objectives:

1. To implement the static web pages using HTML and do client-side validation using JavaScript.
2. To design and work with databases using Java.
3. To develop an end to end application using java full stack.
4. To introduce Node JS implementation for server-side programming.
5. To experiment with single page application development using React.

Course Outcomes: At the end of the course, student will be able to

CO	Course Outcomes	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Build a custom website with HTML, CSS, and Bootstrap and little JavaScript.	PO1, PO5, PS01	-	3	4, 9
CO2	Demonstrate Advanced features of JavaScript and learn about JDBC	PO1, PO5, PS01, PS02	-	3, 4	
CO3	Develop RESTful backend services using Node.js and Express.	PO1, PO2, PO3, PO5, PS01, PS02	-	4, 6	
CO4	Implement CRUD operations using Node.js with database connectivity.	PO1, PO2, PO3, PO5, PS01, PS02	-	3, 6	
CO5	Design and develop Single Page Applications (SPA) using React.js with routing and API integration.	PO1, PO2, PO3, PO5, PS01, PS02	-	6	

Exercises:

1. Build a responsive web application for shopping cart with registration, login, catalog and cart pages using CSS3 features, flex and grid.
2. Make the above web application responsive web application using Bootstrap framework.
3. Use JavaScript for doing client –side validation of the pages implemented in experiment 1 and experiment 2.
4. Explore the features of ES6 like arrow functions, callbacks, promises, async/await. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.
5. Develop a Node.js application that connects with the database (MySQL) and perform CRUD operations.
6. Develop a Node.js application to read, write and manipulate JSON files using fs module.
7. Develop a Node.js and Express-based controller that connects the Shopping Cart web application developed in experiment 1 with the database created in experiment 5.
8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism inNode.jsusingCookiesand Express Sessions.
9. Create a custom server using http module and explore the other modules of Node JS like OS,path, event.
10. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
11. For the above application create authorized end points using JWT (JSON Web Token).
12. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
13. Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
14. Create a TODO application in react with necessary components and deploy it into GitHub.

REFERENCE BOOKS:

1. Jon Duckett, Beginning HTML, XHTML, CSS, and Java Script, Wrox Publications, 2010.
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O'Reilly Media, 2nd Edition, 2008.
3. Vasan Subramanian, Pro MERNStack, Full Stack Web App Development with Mon o, Express, React, and Node , 2nd Edition, APress.

25MS22001: INNOVATION AND ENTREPRENEURSHIP

B.Tech. II Year II Sem.

L	T	P/D	C
2	-	-/-	2

Prerequisite(s): Nil

Course Objectives: Develop an ability to

1. To familiarize on the basic concepts of innovation, entrepreneurship and its importance.
2. To Identify and analyze the process of problem-opportunity identification, market segmentation, and idea generation techniques.
3. To initiate prototype development and understand minimum viable product.
4. To develop initial business and financial planning and Go-to-Market strategies
5. To impart knowledge on establishing startups, venture pitching and IPR

Course Outcomes Mapping: At the end of the course, students would be able to

CO	Course Outcomes (CO)	Related POs and PSOs	Related Units	BTL	Related SDGs
CO1	Apply the entrepreneurship and the entrepreneurial process and its significance in economic development.	5,12	I, III	3	9
CO2	Assess the problem from an industry perspective and generate solutions using the design thinking principles.	6,12	II	4,6	
CO3	Assess market competition, estimate market size, and develop a prototype.	3,12	III	4, 6	
CO4	Analyze business and financial planning models and Go-to-Market strategies.	11,12	IV	4	
CO5	Able to build a start-up, register IP and identify funding opportunities.	6,7,11	V	5, 6	

UNIT I: Fundamentals of Innovation and Entrepreneurship

Innovation: Introduction, need for innovation, Features, Types of innovations, innovations in manufacturing and service sectors, fostering a culture of innovation, planning for innovation.

Entrepreneurship: Introduction, types of entrepreneurship attributes, mindset of entrepreneurial and intrapreneurial leadership, Role of entrepreneurs in economic development. Woman Entrepreneurship, Importance of on-campus startups. Understanding to build entrepreneurial mindset attributes and networks individuals while on campus.

Core Teaching Tool: Simulation, Game, Industry Case Studies (Personalized for students – 16 industries to choose from), Venture Activity.

UNIT II: Problem and Customer Identification

Identification of gap, problem, analyzing the problem from a industry perspective, real-world

problems**,** market and customer segmentation, validation of customer problem fit, Iterating problem-customer fit, Competition and Industry trends mapping and assessing initial opportunity, Porter's Five Force Model. Idea generation, Ideation techniques: Brainstorming, Brain writing, Round robin, and SCAMPER, Design thinking principles, Mapping of solution to problem.
Core Teaching Tool: Several types of activities including: Class, game, GenAI, 'Get out of the Building' and Venture Activity.

UNIT III: Opportunity assessment and Prototype development
Identify and map global competitors, review industry trends, and understand market sizing: TAM, SAM, and SOM. Assessing scope and potential scale for the opportunity.
Understanding prototyping and Minimum Viable Product(MVP).
Developing a prototype: Testing, and validation.
Core Teaching Tool: Venture Activity, no-code Innovation tools, Class activity

UNIT IV: Business & Financial Models
Introduction to Business Model and types, Lean Canvas Approach: 9-block lean canvas model, building lean canvas for your startup. Business planning: components of Business plan- Sales plan, People plan and financial plan, Financial Planning: Types of costs, preparing a financial plan for profitability using a financial template, understanding the basics of Unit economics, Economies of Scale and analyzing financial performance. Go-To-Market(GTM) approach—Selecting the Right Channel, creating digital presence, and building customer acquisition strategy.
Core Teaching Tool: Founder Case Studies – Sama and Securely Share; Class activity and discussions; Venture Activities.

UNIT V: Startups and IPR
Startup requirements, building founding team members and mentors, pitch preparation, start-up registration process, funding opportunities and schemes, institutional support to entrepreneurs, startup lifecycle, documentation, legal aspects in startup, venture pitching readiness, National Innovation Startup Policy (NISP) and its features.
Patents, Designs, Patentability, Procedure for grants of patents. Indian Scenario of Patenting, International Scenario: International cooperation on Intellectual Property. Patent Rights: Scope of Patent Rights. Copyright, trademark, and GI. Licensing and transfer of technology.
Core Teaching Tool: Expert talks; Cases; Class activity and discussions; Venture Activities.

Text Books:

1. Innovation and Entrepreneurship by John R Bessant, JoeTidd, 2024, Wiley.
2. A Practical Guide for Launching Customer Centric Ventures by Ajay Batra, The Stratup Launch Book, 2020, Wiley. (For Core Teaching Tool).