CMR COLLEGE OF ENGINEERING & TECHNOLOGY
(UGC AUTONOMOUS)
B. Tech–Computer Science and Engineering
CBCS & OUTCOME BASED COURSE STRUCTURE & SYLLABUS
(Effective for the students admitted into I year from the Academic Year 2025-26)

SEMESTER - I							
S. No	**Course Code**	**Course Title**	**Course Category**	**Hours per Week**			**Credits**
				L	**T**	**P**	**C**
1	A500001	Matrices and Calculus	BSC	3	1	0	4
2	A500008	Advanced Engineering Physics	BSC	3	0	0	3
3	A502202	Basic Electrical Engineering	ESC	3	0	0	3
4	A505201	Programming for Problem Solving	ESC	3	0	0	3
5	A500502	Advanced Engineering Physics Laboratory	BSC	0	0	2	1
6	A500506	Introduction to Social Innovation	HSMC	0	0	2	1
7	A502501	Basic Electrical Engineering Laboratory	ESC	0	0	2	1
8	A505501	Programming for Problem Solving Laboratory	ESC	0	0	2	1
9	A503503	Computer Aided Engineering Drawing	ESC	0	1	2	2
10	A505504	IT Workshop	ESC	0	0	2	1
			Total:	**12**	**2**	**12**	**20**
			Total hours per week:	**26**			

SEMESTER - II							
S. No	**Course Code**	**Course Title**	**Course Category**	**Hours per Week**			**Credits**
				L	**T**	**P**	**C**
1	A500002	Ordinary Differential Equations and Vector Calculus	BSC	3	0	0	3
2	A500009	Engineering Chemistry	BSC	3	0	0	3
3	A500101	English for Skill Enhancement	HSMC	3	0	0	3
4	A504201	Electronic Devices and Circuits	ESC	3	0	0	3
5	A505301	Data Structures	PCC	3	0	0	3
6	A500504	English Language and Communication Skills Laboratory	HSMC	0	0	2	1
7	A500503	Engineering Chemistry Laboratory	BSC	0	0	2	1
8	A505507	Python Programming Laboratory	ESC	0	0	2	1
9	A503502	Engineering Exploration and Practice	HSMC	0	0	2	1
10	A505505	Data Structures Laboratory	PCC	0	0	2	1
			Total:	**15**	**0**	**10**	**20**
			Total hours per week:	**25**			
Total Number of Credits B.Tech I Year: 40							

SEMESTER - III							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	A500006	Computer Oriented Statistical Methods	BSC	3	0	0	3
2	A505302	Computer Organization and Architecture	PCC	3	0	0	3
3	A505303	Object Oriented Programming through JAVA	PCC	3	0	0	3
4	A505304	Software Engineering	PCC	3	0	0	3
5	A505305	Database Management Systems	PCC	3	0	0	3
6	A500501	Computational Mathematics Laboratory	BSC	0	0	2	1
7	A505508	Object Oriented Programming through JAVA Laboratory	PCC	0	0	2	1
8	A505509	Software Engineering Laboratory	PCC	0	0	2	1
9	A505510	Data Base Management Systems Laboratory	PCC	0	0	2	1
10	A505701	Skill Development Course (Node Js, React JS and Django)	SDC	0	0	2	1
11	A500903	Gender Sensitization	VAC	1	0	0	1
	A500904	Human Values and Professional Ethics					
Total:				**16**	**0**	**10**	**21**
Total hours per Week:				**26**			

SEMESTER - IV							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	A505306	Discrete Mathematics	PCC	3	0	0	3
2	A505307	Operating Systems	PCC	3	0	0	3
3	A505308	Algorithm Design and Analysis	PCC	3	0	0	3
4	A505309	Computer Networks	PCC	3	0	0	3
5	A505310	Machine Learning	PCC	3	0	0	3
6	A500507	Social Innovation and Entrepreneurship	HSMC	0	1	2	2
7	A505511	Operating Systems Laboratory	PCC	0	0	2	1
8	A505512	Computer Networks Laboratory	PCC	0	0	2	1
9	A505513	Machine Learning Laboratory	PCC	0	0	2	1
10	A505702	Skill Development Course (Data Visualization)	SDC	0	0	2	1
11	A500901	Environmental Science	VAC	1	0	0	1
Total:				**16**	**1**	**10**	**22**
Total hours per Week:				**27**			
Total Number of Credits B.Tech II Year: 83							

SEMESTER - V							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	XXXXXX	Professional Core Course	PCC	3	0	0	3
2	XXXXXX	Professional Core Course	PCC	3	0	0	3
3	XXXXXX	Professional Core Course	PCC	3	0	0	3
4	XXXXXX	Professional Elective-I	PEC	3	0	0	3
5	XXXXXX	Open Elective-I	OEC	2	0	0	2
6	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
7	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
8	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
9	XXXXXX	Skill Development Course (UI Design – Flutter)	SDC	0	0	2	1
10	A505801	Field Project	PROJ	0	0	4	2
	A505802	Internship					
11	A500902	Indian Knowledge System	VAC	1	0	0	1
		Total:		**15**	**0**	**12**	**21**
		Total hours per Week:		**27**			

SEMESTER - VI							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	XXXXXX	Professional Core Course	PCC	3	0	0	3
2	XXXXXX	Professional Core Course	PCC	3	0	0	3
3	XXXXXX	Professional Core Course	PCC	3	0	0	3
4	XXXXXX	Professional Elective-II	PEC	3	0	0	3
5	XXXXXX	Open Elective – II	OEC	2	0	0	2
6	A500505	English for Employability Skills Laboratory	HSMC	0	0	2	1
7	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
8	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
9	XXXXXX	Professional Core Course Laboratory	PCC	0	0	2	1
10	XXXXXX	Skill Development Course Laboratory	SDC	0	0	2	1
		Total:		**14**	**0**	**10**	**19**
		Total hours per Week:		**24**			
Total Number of Credits B.Tech III Year : 123							

SEMESTER - VII							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	A500103	Fundamentals of Management	HSMC	3	0	0	3
2	XXXXXX	Professional Core Course	PCC	3	0	0	3
3	XXXXXX	Professional Core Course	PCC	3	0	0	3
4	XXXXXX	Professional Elective-III	PEC	3	0	0	3
5	XXXXXX	Professional Elective – IV	PEC	3	0	0	3
6	XXXXXX	Open Elective – III	OEC	2	0	0	2
7	A566504	Professional Core Course Laboratory	PCC	0	0	2	1
8	A505518	Professional Core Course Laboratory	PCC	0	0	2	1
9	A505803	Industry Oriented Mini Project	PROJ	0	0	4	2
	A505804	Summer Internship					
Total:				**17**	**0**	**8**	**21**
Total hours per Week:				**25**			

SEMESTER - VIII							
S. No	Course Code	Course Title	Course Category	Hours per Week			Credits
				L	T	P	C
1	PEC	Professional Elective – V	PEC	3	0	0	3
2	PEC	Professional Elective – VI	PEC	3	0	0	3
3	A505805	Project Work	PROJ	0	0	28	14
Total:				**6**	**0**	**28**	**20**
Total hours per Week:				**34**			
Total Number of Credits B.Tech IV Year : 164							

(A500001) MATRICES AND CALCULUS
(Common to All)

L	T	P	C
3	1	0	4

B.Tech I Year I Sem

UNIT-I: Matrices
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: Eigen values and Eigen vectors
Linear Transformation and Orthogonal Transformation: Eigen values – Eigen vectors 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
Limits and Continuous functions and its 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).

Definition of Improper Integral: Beta and Gamma functions and their applications.

UNIT-IV: Multivariable Calculus (Partial Differentiation and applications)
Definitions of Limit and continuity – Partial Differentiation: Euler's Theorem – Total derivative – Jacobian – Functional dependence & independence. Applications: Maxima and minima of functions of two variables and three variables using method of Lagrange multipliers.

UNIT-V: Multivariable Calculus (Integration)
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 triple integrals.

TEXTBOOKS
1. Higher Engineering Mathematics (36th Edition), B.S. Grewal, Khanna Publishers, 2010.
2. Advanced Engineering Mathematics (5th Edition), R.K. Jain and S.R.K. Iyengar, Narosa Publications, 2016.

REFERENCES
1. Advanced Engineering Mathematics (9th Edition), Erwin Kreyszig, John Wiley & Sons, 2006.
2. Calculus and Analytic geometry (9th Edition), G.B. Thomas and R.L. Finney, Pearson, Reprint, 2002.

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

1. Formulate the matrix representation of a system of linear equations and analyze the corresponding solution set.
2. Determine the eigenvalues and eigenvectors of a matrix, and reduce a quadratic form to its canonical form using orthogonal transformations.
3. Apply the mean value theorems to solve relevant problems in mathematical analysis. Find solution of improper integrals by using Beta and Gamma function
4. Find the extreme values of functions of two variables, both with and without constraints.
5. Evaluate multiple integrals and apply the concept to calculate areas and volumes.

CO-PO MAPPING

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

END

(A500008) ADVANCED ENGINEERING PHYSICS

(Common to All)

L	T	P	C
3	0	0	3

B.Tech I Year I Sem

UNIT-I: CRYSTALLOGRAPHY

Introduction: Unit cell, space lattice, basis, lattice parameters; crystal structures, Bravais lattices, packing factor: SC, BCC, FCC; Miller indices, inter-planar distance, defects in crystals (qualitative): point defects, line defects, surface defects and volume defects.

CONCEPT OF NANOMATERIALS & MATERIALS CHARACTERIZATION:

Surface to volume ratio, X -ray diffraction: Bragg's law, powder method, crystallite size - Debye Scherrer's formula, scanning electron microscopy (SEM): block diagram, working principle.

UNIT-II: QUANTUM MECHANICS

Introduction, de-Broglie hypothesis, 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, Kronig-Penny Model (qualitative), classification of solids, concept of discrete energy levels and quantum confinement in nanomaterial.

SEMICONDUCTORS AND DEVICES:

Intrinsic and Extrinsic semiconductors (qualitative), Hall effect, Construction, principle of operation and characteristics of P-N Junction diode. Direct and indirect band gap semiconductors – LED and Solar cells, their structure, materials, working principle and characteristics.

UNIT-III: QUANTUM COMPUTING

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

UNIT-IV: MAGNETIC 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: magnets for electric vehicles (EV).

DIELECTRIC MATERIALS :

Introduction to dielectric materials, types of polarization (qualitative): electronic, ionic & orientation; ferroelectric, piezoelectric, pyroelectric materials and their applications: Ferroelectric Random-Access Memory (Fe-RAM), production of Ultrasonics by piezoelectric method.

UNIT-V: LASER

Introduction to laser, characteristics of laser, Einstein coefficients and their relations, metastable state, population inversion, pumping mechanism, lasing action, Ruby laser, He-Ne laser, CO_2 laser, semiconductor diode laser, applications: Bar code scanner, LIDAR for autonomous vehicle.

FIBER OPTICS

Introduction to Fiber optics, total internal reflection, construction of optical fiber, acceptance angle, numerical aperture, classification of optical fibers, losses in optical fiber (qualitative), applications: optical fibers for communication system.

TEXTBOOKS

1. Crystallography: An Introduction (3rd Edition), Walter Borchardt-Ott, Springer, 2011.
2. Introduction to Solid State Physics (9th Edition) Charles Kittel, John Wiley & Sons, Inc, 2018
3. Introduction to Classical and Quantum Computing, (1st Edition) Thomas G. Wong, Rooted Grove, 2022.
4. Physics of Semiconductor devices (4th edition), Simon.MSze and Kwok K . Ng, Wiley Student Edition, 2006.

REFERENCES

1. Quantum Computing (1st Edition), Jozef Gruska, McGraw Hill, 1999.
2. Quantum Computation and Quantum Information (10th Edition), Cambridge University Press, 2010

3. Optical Fiber Communications Principles and Practice (3rd Edition), Pearson Education Limited,2009
4. Essentials of Nano science& Nanotechnology(1stedition), Narasimha Reddy Katta, Typical Creatives NANO DIGEST, 2021
5. Engineering Physics(3rd edition), PK Palanisam, SciTech Publications, 2015.

COURSE OUTCOMES:

On completion of the course students will be able to

1. Analyze crystal structures, identify defects, and apply XRD and SEM techniques for material characterization.
2. Apply quantum mechanical principles to explain particle behavior and energy band formation in solids and classify semiconductor devices.
3. Understand quantum computing concepts, use quantum gates, and explain basic quantum information process.
4. Classify magnetic and dielectric materials and explain their properties, synthesis, and applications.
5. Appreciate the principles of lasers and fiber optics and their applications in communication.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	-	-	-	-	-	-	-	-
CO2	3	3	2	-	2	-	-	-	-	-	-
CO3	3	3	2	2	2	-	-	-	-	-	-
CO4	3	3	2	-	-	-	-	-	-	-	-
CO5	3	3	2	-	2	-	-	-	-	-	-

END

(A502202) BASIC ELECTRICAL ENGINEERING

(Common to ECE, CSE, CSD and CSM)

L	T	P	C
3	0	0	3

B.Tech I Year I Sem

Prerequisites: Mathematics

UNIT-I:
D.C. Circuits: 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. Time-domain analysis of first-order RL and RC circuits.

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

UNIT-III:
Transformers: Ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Auto-transformer and three-phase transformer connections.

UNIT-IV:
Electrical Machines: Construction and working principle of dc machine, performance characteristics of dc shunt machine. Generation of rotating magnetic field, Construction and working of a three-phase induction motor, Significance of torque-slip characteristics. Single-phase induction motor, Construction and working. Construction and working of synchronous generator

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

TEXTBOOKS

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.

REFERENCES

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. M. S. Sukhija, T. K. Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford, 1st Edition, 2012.
4. Abhijit Chakrabarthi, Sudipta Debnath, Chandan Kumar Chanda, “Basic Electrical Engineering”, 2nd Edition, McGraw Hill, 2021.
5. L. S. Bobrow, “Fundamentals of Electrical Engineering”, Oxford University Press, 2011.
6. E. Hughes, “Electrical and Electronics Technology”, Pearson, 2010.
7. V. D. Toro, “Electrical Engineering Fundamentals”, Prentice Hall India, 1989

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

1. Understand and analyze basic concepts of DC Circuits
2. Understand and analyze basic concepts of AC Circuits
3. Discus the technical aspects of transformers
4. Study the working principles of Electrical Machines.
5. Introduce components of Low Voltage Electrical Installations

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	-	-	-	-	-	-	-	-	-	-	-
CO2	-	-	-	-	-	-	-	-	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-

END

(A505201) PROGRAMMING FOR PROBLEM SOLVING

(Common to CSE, CSD, CSM, ECE and EEE)

L	T	P	C
3	0	0	3

B.Tech I Year I Semester

UNIT-I:

Algorithms & Flowchart: Introduction to Algorithms, Characteristics of Algorithms, Introduction to flowcharts, Various symbols used in flowcharts, Algorithms and Flowcharts for various mathematical problems.

Introduction to C Programming: Executable Statements, General Form of a C Program, C Language Elements, Variable Declarations and Data Types, Operators, Precedence and Associativity, Arithmetic Expressions and its evaluations, Formatting Input/Output statements.

Decision Statements: Control Structures, Conditions, if Statement, if Statements with Compound Statements, Switch-Case statement.

UNIT-II:

Loop Control Statements: Repetition in Programs, Looping Statements – While, do-while, for Loop, Nested Loops, Jumping Statements – Goto, Break and Continue Statements.

Functions: Overview, Library functions, defining a function, accessing a function, function prototype, passing arguments to a function, Scope Rules – Storage Classes.

Recursion: The Nature of Recursion, Tracing a Recursive Function, Recursive Mathematical Functions.

UNIT-III:

Pointers: Pointers and the Indirection Operator, Declaration & Initialization of a pointer, Multiple Calls to a Function with Input/Output Parameters, Formal Output Parameters as Actual Arguments, Pointer – Arithmetic, Pointer to Pointer, Dynamic Memory Allocation.

Arrays: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access, Passing Arrays to Functions, Parallel Arrays, Multidimensional Arrays, Pointers and Arrays.

UNIT-IV:

Strings: String Basics, defining a String, Initialization of Strings, Reading and Writing a String, String Library Functions, Pointers and Strings.

Structures and Unions: Introduction, defining a Structure, processing a Structure, User-Defined Structure Types, Array of Structures, Nested Structures, Self-referential Structures, Structures and Pointers, Structures and Functions, Unions, Enumerated Data type.

UNIT-V:

Text and Binary File Pointers: Input/Output Files – Basic file Operations, Random Access Files, Binary Files, Command Line Arguments.

Searching and Sorting: Basic searching in an array of elements (linear and binary search techniques), Basic algorithms to sort an array of elements (Bubble, Insertion and Selection sort algorithms).

TEXTBOOKS

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
2. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson

REFERENCES

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

COURSE OUTCOMES:

On completion of the course students will be able to

1. Develop algorithms and flowcharts for solving computational problems and implement them using C language syntax.
2. Write C programs using control structures such as conditional, iterative, and jumping statements.
3. Design modular programs using user-defined functions, recursion, and demonstrate understanding of scope and storage classes.
4. Apply advanced C constructs such as pointers, arrays, strings, and structures to solve real-time problems.
5. Perform file handling operations and implement searching and sorting algorithms using C language.

CO-PO MAPPING

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

END

(A500502) ADVANCED ENGINEERING PHYSICS LABORATORY

(Common to All)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

(Any 8 experiments are to be performed)

1. Determination of Planck's constant using Photo Electric Effect.
2. Determination of energy gap of a semiconductor.
3. Determination of Hall coefficient and carrier concentration of a given semiconductor.
4. Study of V-I characteristics of a LED
5. Study of V-I characteristics of a Solar Cell and find its Fill factor.
6. Determination of magnetic moment of a bar magnet and horizontal earth magnetic field.
7. Study of B-H curve of a ferromagnetic material.
8. Determination of dielectric constant of a given material.
9. Study of V-I & L-I characteristics of a given laser diode
10. a. Determination of wavelength of a laser using diffraction grating.
    b. Determination of LASER beam divergence
11. a. Determination of numerical aperture of a given optical fiber.
    b. Determination of bending losses of a given optical fiber.

COURSE OUTCOMES:

On completion of the course students will be able to

1. Determine the energy gap using semiconductors using experimental methods.
2. Appreciate and apply the principles of quantum physics in the field of optoelectronics
3. Analyse the variation of Magnetic fields and their properties
4. Examine and interpret the variation of dielectric properties of a material.
5. Demonstrate working knowledge of laser systems and optical fiber parameters through experimental study.

CO-PO MAPPING

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

END

(A500506) INTRODUCTION TO SOCIAL INNOVATION

(Common to All)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

WEEK-1

Types and features of community- Rural, Suburban, Urban and Regional

WEEK-2

Service based learning, Aims of Community based projects, Sustainable Development Goals

WEEK-3

Community visit, Report Writing, Resource Diagram, Chapati Diagram, Transect Walk

WEEK-4

The non-profit sector, public sector, the private sector, the informal sector

WEEK-5

Poster presentation on four sectors

WEEK-6

Process of Design Thinking

WEEK-7

Social organizations and enterprises, social movements

WEEK-8

Social softwares and open-source methods

WEEK-9

Introduction to Ethics, moral values, significance of professional ethicscode of conduct for engineers

WEEK-10

Identify ethical dilemmas in different tasks of engineering, applying moral theories and codes of conduct forresolution of ethical dilemmas

WEEK-11

Case studies on Engineering Ethics

WEEK-12

Steps for Patent filing and Startups, Procedure for grants of patents. Indian Scenario of Patenting, International cooperation on Intellectual Property, Documentation, Panel Presentation

TEXTBOOKS:

1. Social Entrepreneurship for the 21st Century: Innovation Across the Non-Profit, Private and Public Sectors; Georgia Levenson Keohane; Tata McGraw Hill
2. Solving Problems with Design Thinking - Ten Stories of What Works (Columbia Business School Publishing) Hardcover – 20 Sep 2013 by Jeanne Liedtka (Author), Andrew King (Author), Kevin Bennett (Author)

REFERENCE BOOKS:

1. Fundamentals of Intellectual Property (English) 1st Edition (Paperback, Dr. Kalyan C. Kankanala) Publisher: Asia Law House ISBN: 9789381849514, 938184951X Edition: 1st Edition, 2012.
2. Indian Patent Law (English, Paperback, Kalyan C. Kankanala) Publisher: Oxford University Press-New Delhi, ISBN: 9780198089605, 0198089600 Edition: 2012.
3. Social Enterprises: An Organizational Perspective edited; Benjamin Gidron, Yeheskel Hasenfeld; Palgrave Macmillan

4. Hasso Plattner, Christoph Meinel and Larry Leifer (eds), "Design Thinking: Understand – Improve – Apply", Springer, 2011.
5. Engineering Ethics: An Industrial Perspective; Gail Baura; Elsevier
6. Intellectual Property and Financing Strategies for Technology Startups; Gerald B. Halt, Jr., John C. Donch, Jr., Amber R. Stiles, Robert Fesnak; Springer

COURSE OUTCOMES:

On completion of the course students will be able to

1. Identify community issues through community Interaction
2. Illustrate the factors affecting social innovation in various sectors
3. Apply design thinking concept to analyze the community problems
4. Adopt the ethical values in implementing social innovation
5. Describe the process of property rights and patent filing.

CO-PO MAPPING

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

END

(A502501) BASIC ELECTRICAL ENGINEERING LABORATORY

(Common to ECE, CSE, CSD and CSM)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

Prerequisites: Basic Electrical Engineering

List of experiments/demonstrations:

PART- A (compulsory)

1. Verification of KVL and KCL
2. Verification of Thevenin's and Norton's theorem
3. Transient Response of Series RL and RC circuits for DC excitation
4. Resonance in series RLC circuit
5. Calculations and Verification of Impedance and Current of RL, RC and RLC series Circuits.
6. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single-Phase Transformer
7. Performance Characteristics of a DC Shunt Motor
8. Torque-Speed Characteristics of a Three-phase Induction Motor.

PART-B (any two experiments from the given list)

1. Verification of Superposition theorem.
2. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
3. Measurement of Active and Reactive Power in a balanced Three-phase circuit
4. Magnetization Characteristics of DC Shunt Generator.

TEXTBOOKS:

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. M. S. Sukhija, T. K. Nagsarkar, "Basic Electrical and Electronics Engineering", Oxford, 1st Edition, 2012.
4. Abhijit Chakrabarthi, Sudipta Debnath, Chandan Kumar Chanda, "Basic Electrical
5. Engineering", 2nd Edition, McGraw Hill, 2021.
6. L. S. Bobrow, "Fundamentals of Electrical Engineering", Oxford University Press, 2011.
7. E. Hughes, "Electrical and Electronics Technology", Pearson, 2010.
8. V. D. Toro, "Electrical Engineering Fundamentals", Prentice Hall India, 1989

COURSE OUTCOMES:

On completion of the course students will be able to

1. Verify the basic Electrical circuit Laws through different experiments.
2. Analyze the transient responses of R, L and C circuits for DC input.
3. Calculate the Impedance and Current of RL, RC and RLC series Circuits.
4. Evaluate the performance of Electrical Machines through various testing methods.
5. Measure the Active and Reactive Power in a single-phase transformer.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	-	-	-	-	-	-	-	-	-	-	-
CO2	-	-	-	-	-	-	-	-	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-

*****END*****

(A505501) PROGRAMMING FOR PROBLEM SOLVING LABORATORY

(Common to EEE, ECE, CSE, CSD and CSM)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

[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]

Operators and Expressions:

Practice Programs:

- Write a simple program that prints the results of all the operators available in C, Read required operand values from standard input.
- Write a C program to swap the contents of any two operands using suitable bitwise operator.
- Write a C program to compute $s = ut + \frac{1}{2} at^2$ [Read u, t & a values from keyboard].
- Write a C program for the simple and compound interest.

Additional Programs:

- Write a program that reads the radius of a circle (as a float value) and computes and prints the diameter, the circumference and the area, consider π value as a symbolic constant.
- Write a program that asks the user to enter the total time elapsed, in seconds, since an event and converts the time to hours, minutes and seconds. The time should be displayed as hours: minutes: seconds. [Hint: Use the remainder operator]

Decision statements:

Practice Programs:

- Write a C program for finding the max and min from the given three numbers.
- Write a C program to find the roots of a Quadratic equation.
- 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).

Additional Programs:

- Write a C program to calculate the electricity bill. Read starting and ending meter readings. The charges are as follows:

No. of Units Consumed	**Unit Cost (per unit)**
>=500	5.00 Rs/unit
>=200 to <500	3.50 Rs/unit
>=100 & <200	2.50 Rs/unit
Less than 100	1.50 Rs/unit

- Write a C program to convert years into 1. MINUTES 2. HOURS 3. DAYS 4. MONTHS 5. SECONDS using switch-case statement.

Loop Control Statements:

Practice Programs:

- 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

- Write a C program to print all the prime numbers between the given limits.

- Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome or not.
- Write a C program to construct a pyramid of numbers as follows:

```
1                    a
1  2                 b  b
1  2  3              c  c  c
1  2  3  4           d  d  d  d
```

Additional Programs:

- 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.
- Write a C program to check whether the given year is leap-year or not using goto statement.
- Write a C program to summate the Sin Series of n terms [Hint: input x and n terms]

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

Functions & Recursion:

Practice Programs:

- Write a C program to find the sum of any two integers, using function.
- Write C programs that uses both recursive and non-recursive functions:
  a. To find the factorial of a given integer.
  b. To find the GCD (greatest common divisor) of two given integers.

Additional Programs:

- Write a C program to check whether the given number is Armstrong or not? using function
- Write a C program to convert the given decimal number into equivalent binary form.

Pointers and Arrays:

Practice Programs:

- Write a C program to perform different arithmetic operations using pointers.
- Write a C program to swap two numbers using call-by-value and call-by-reference concept.
- Write a C program to find the minimum, maximum and average in an array of integers.
- Write a C program that uses functions to perform the Addition of Two Matrices

Additional Programs:

- Write a C program that uses functions to perform the Multiplication of Two Matrices
- Write a program for display values in reverse order from an array using a pointer.
- Write a C program to read and display array elements using pointers only, and compute the minimum, maximum, and average using pointer operations.

Strings:

Practice Programs:

- Write C programs without using String-Handling functions:
  a. to find the length of a given string
  b. to append one string at end of another string
- 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, etc.) without using string handling functions
- Write a C program that displays the position of a character ch in the string S or – 1 if S doesn't contain ch.
- Write a C program to count the lines, words and characters in a given text.

Additional Programs:

- 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
- Write a C program to find the length of a given string including and excluding spaces using pointers.

- Write a C program to read string from keyboard and display it using character pointer.

Structures:

Practice Programs:

- Write a C program to read and display a student structure with the following data items: student_name, student_rno, student_percentage.
- Write a C program to copy the structure elements from one structure variable to another.
- Write a C program to declare pointer to structure and display the contents of the structure.

Additional Programs:

- Write a C program to find the sum of any two complex numbers using function.
- Write a C program to read and display roll number, full name and date of birth of a student using nested structures.
- Write a C program to create enumerated data type for 12 months. Display their values in integer constants.

Files:

Practice Programs:

- Write a C program to write data to text file and read it.
- Write a C program which copies one file to another, replacing all lowercase characters with their uppercase equivalents and read the result file.

Additional Programs:

- Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file) using CLA.
- Write a C program to read and display the contents of an existing file by skipping the first n-characters from the beginning of the file. [Hint: Use fseek() funcion]

Sorting and Searching:

Practice Programs:

- 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.
- Write a C program that uses recursive and non-recursive functions to search for a Key value in a given sorted list of integers using binary search method.
- Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.

Additional Programs:

- Write a C program that sorts the given array of integers using selection sort in descending order
- Write a C program that sorts the given array of integers using insertion sort in ascending order
- Write a C program that sorts a given array of names.

TEXTBOOKS:

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
2. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson

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

COURSE OUTCOMES:

On completion of the course students will be able to

1. Develop and Execute C programs using basic input/output, operators, and control flow constructs.
2. Solve real-time problems using loops, user-defined functions, and recursion.
3. Apply pointer, array, string, and structure concepts to build efficient C programs.
4. Implement file operations and command-line arguments to read, write, and manipulate data.
5. Write programs for basic searching and sorting techniques using iterative and recursive logic.

CO-PO MAPPING

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

END

(A503503) COMPUTER AIDED ENGINEERING DRAWING

(Common to EEE, ECE, CSE, CSM and CSD)

L	T	P	C
0	1	2	2

B.Tech I Year I Sem

UNIT-I

Introduction to Engineering Drawing:

Principles of Engineering Drawing and their Significance, Geometrical Constructions, Introduction to Computer Aided Drafting Tool, Computer aided drafting of Conic Sections: Ellipse, Parabola and Hyperbola – General Method (eccentricity) only. Computer aided drafting of Cycloid, Epicycloids and Hypocycloid, Computer aided drafting of Scales – Plain & Diagonal Scales

UNIT-II

Orthographic Projections:

Introduction to Principles of Orthographic Projections – Conventions – Projections of Points, Lines, and Projections of Plane regular geometric figures using Computer aided drafting tool.

UNIT-III

Projections of Regular Solids:

Introduction to Regular Solids – Prism, Cylinder, Pyramid, Cone – Regular views using Computer aided drafting tool.

UNIT-IV

Isometric Projection:

Principles of Isometric Projection – Isometric Scale – Isometric Views –Conventions – Isometric Views of Lines, Plane Figures, Simple and Compound Solids – Isometric Projection of objects having non- isometric lines, Isometric Projection of Spherical Parts using Computer aided drafting tool.

UNIT-V

Conversion of Isometric Views to Orthographic Views and Vice-versa – Conventions, Conversion of orthographic projection into isometric view

TEXTBOOKS:

1. Engineering Drawing, N.D. Bhatt, Charotar Publishers, 54th Edition, 2023
2. Computer Aided Engineering Drawing, K. Balaveera Reddy et al, CBS Publishers, 2nd Edition, 2015

REFERENCE BOOKS:

1. Engineering Drawing, M. B. Shah, B.C. Rane, Pearson, 3rd Edition, 2015
2. Engineering Drawing, Basant Agrawal and C M Agrawal, McGraw Hill, 3rd Edition, 2019
3. Engineering Graphics and Design, WILEY, John Wiley and sons Inc, 3rd Edition, 2020
4. Engineering Drawing and graphics Using AutoCAD, T. Jeyapoovan, Vikas, S.Chand and Company Ltd, 3rd Edition, 2010

COURSE OUTCOMES:

On completion of the course students will be able to

1. Understand and Apply concepts to construct engineering curves using Computer aided drafting tool
2. Apply the Orthographic projection for Points, Lines and Planes by Drafting tool
3. Construct and interpret Orthographic projections of Solids using Computer aided drafting tool
4. Create the Orthographic view to Isometric view using Computer aided drafting tool
5. Conversion of Orthographic view to Isometric view & vice versa using Computer aided drafting tool

CO-PO MAPPING

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

*****END*****

(A505504) IT WORKSHOP

(Common to CSE, CSD and CSM)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

PC Hardware

- **Task 1**: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.
- **Task 2**: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.
- **Task 3**: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.
- **Task 4**: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Internet & World Wide Web

- **Task1:** Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.
- **Task 2:** Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.
- **Task 3**: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.
- **Task 4:** Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

- **Task 1** – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of LaTeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.
- **Task 2**: Using LaTeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.
- **Task 3**: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.
- **Task 4**: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns,
- Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs
- and Mail Merge in word.

Excel

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool. Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

- **Task 1:** Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

- **Task 2:** Calculating GPA - Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP
- **Task 3:** Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional Formatting.

PowerPoint

- **Task 1:** Students will be working on basic power point utilities and tools which help them create basic PowerPoint presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.
- **Task 2**: Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.
- **Task 3**: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

REFERENCE BOOKS:

1. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech
2. The Complete Computer upgrade and repair book, 3rd edition Cheryl A Schmidt, WILEY Dreamtech
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
4. PC Hardware - A Handbook – Kate J. Chase PHI (Microsoft)
5. LaTeX Companion – Leslie Lamport, PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson
7. and Ken Quamme. – CISCO Press, Pearson Education.
8. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan
   – CISCO Press, Pearson Education.

COURSE OUTCOMES:

On completion of the course students will be able to

1. Perform Hardware troubleshooting
2. Understand Hardware components and inter dependencies
3. Safeguard computer systems from viruses/worms
4. Document/ Presentation preparation
5. Perform calculations using spreadsheet

CO-PO MAPPING

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

END

(A500002) ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS
(Common to All)

L	T	P	C
3	0	0	3

B.Tech I Year II Sem

UNIT-I: First Order Ordinary Differential Equations
Exact differential equations – Equations reducible to exact differential equations – linear 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
Higher order linear differential equations with constant coefficients: Non-Homogeneous terms of the type e^{ax}, $\sin ax$, $\cos ax$, polynomials in x, $e^{ax}V(x)$ and $x\,V(x)$ – Method of variation of parameters.

UNIT-III: Laplace Transforms
Laplace Transforms: Laplace Transform of standard functions – First shifting theorem – Laplace transforms of functions multiplied by 't' and divided by 't' – Laplace transforms of derivatives and integrals of function – Evaluation of integrals by Laplace transforms – Laplace transform of periodic functions – Inverse Laplace transform by different methods, convolution theorem (without proof). Applications: solving Initial value problems by Laplace Transform method.

UNIT-IV: Vector Differentiation
Vector point functions and scalar point functions – Gradient – Divergence and Curl – Directional derivatives – Vector Identities – Scalar potential functions – Solenoidal and Irrotational vectors.

UNIT-V: Vector Integration
Line, Surface and Volume Integrals. Theorems of Green, Gauss and Stokes (without proofs) and their applications.

TEXTBOOKS
1. Higher Engineering Mathematics (36th Edition), B.S. Grewal, Khanna Publishers, 2010.
2. Advanced Engineering Mathematics (5th Edition), R.K. Jain and S.R.K. Iyengar, Narosa Publications, 2016.

REFERENCES
1. Advanced Engineering Mathematics (9th Edition), Erwin Kreyszig, John Wiley & Sons, 2006.
2. Calculus and Analytic geometry (9th Edition), G.B. Thomas and R.L. Finney, Pearson, Reprint, 2002.

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

1. Determine whether a given first-order differential equation is exact, linear or Bernoulli's and apply the concepts to model and analyze real-world problems.
2. Solve higher-order differential equations and apply Method of variation of parameters.
3. Utilize Laplace transform techniques for solving ordinary differential equations.
4. Find Gradient, Divergence, Curl and Directional derivatives of vector point functions and scalar point functions
5. Evaluate line, surface, and volume integrals in various coordinate systems. Transform one type of integral into another using the appropriate vector integral theorems.

CO-PO MAPPING

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

END

(A500009) ENGINEERING CHEMISTRY

(Common to EEE, ECE, CSE, CSM, CSD)

L	T	P	C
3	0	0	3

B.Tech I Year II Sem

UNIT-I: Water and its treatment

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

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

UNIT-II: Electrochemistry 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 electrodes, reference electrodes - Primary reference electrode - Standard Hydrogen Electrode (SHE), Secondary reference electrode - Calomel electrode. Construction, working and determination of p^H of unknown solution using SHE and Calomel electrode.

Corrosion: Introduction- Definition, causes and effects of corrosion - Theories of corrosion, chemical and electrochemical theories of corrosion, Types of corrosion: galvanic, waterline and pitting 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.

UNIT-III: Energy sources

Batteries: Introduction - Classification of batteries - Primary, secondary and reserve batteries with examples. Construction, working and applications of Zn-air 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, Calorific value - Units, HCV & LCV- Dulong's formula - Numerical problems.

Fossil fuels: Introduction, Classification, Petroleum - Refining of Crude oil, Cracking - Types of cracking - Moving bed catalytic cracking. LPG and CNG composition and uses.

Synthetic fuels: Fischer-Tropsch process, Introduction and applications of Hythane and Green Hydrogen.

Biofuels: Biodiesel..

UNIT-IV: Polymers

Definition - Classification of polymers: Based on origin and tacticity with examples - Types of polymerization - Addition (free radical addition mechanism) and condensation polymerization. Plastics, Elastomers and Fibers: Definition and applications (PVC, Buna-S, Nylon-6,6). Differences between thermoplastics and thermosetting plastics, Fiber reinforced plastics (FRP).

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

Biodegradable polymers: Polylactic acid and its applications.

UNIT-V: Advanced Functional Materials

Smart materials: Introduction, Classification with examples - Shape Memory Alloys - Nitinol, Piezoelectric materials - quartz and their engineering applications.

Biosensor - Definition, Amperometric Glucose monitor sensor.

Interpretative spectroscopic applications: UV-Visible spectroscopy for Analysis of pollutants in dye industry, IR spectroscopy in night vision-security, Pollution Under Control - CO sensor (Passive Infrared detection), Raman spectroscopy (application) - Tumour detection in medical applications.

TEXTBOOKS

1. Engineering Chemistry by J. Saroja, and D. Divya, Skytech Publishing Company, 2025.
2. Engineering Chemistry by P.C. Jain and M. Jain, Dhanpatrai Publishing Company, 2010.
3. Engineering Chemistry by Rama Devi, P. Aparna and Rath, Cengage learning, 2025.

REFERENCES

1. Engineering Chemistry: by Thirumala Chary Laxminarayana & Shashikala, Pearson Publications (2020)
2. Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company (P) Ltd. Delhi 2011.
3. Engineering Chemistry by Shikha Agarwal, Cambridge University Press, Delhi 2015.
4. Engineering Analysis of Smart Material Systems by Donald J. Leo, Wiley, 2007.
5. Challenges and Opportunities in Green Hydrogen by Editors: Paramvir Singh, Avinash Kumar Agarwal, Anupma Thakur, R.K Sinha.
6. Raman Spectroscopy in Human Health and Biomedicine, https://www.worldscientific.com/doi/epdf/10.1142/13094
7. E-Content: https://doi.org/10.1142/13094 | October 2023
8. E-books: https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2u.

COURSE OUTCOMES:

On completion of the course students will be able to

1. Apply the principles of water chemistry to estimate hardness using EDTA and analyze water treatment processes, including disinfection, defluoridation, softening, and desalination methods.
2. Explain electrochemical concepts, determine electrode potentials, and evaluate corrosion mechanisms; propose appropriate corrosion control techniques for engineering applications.
3. Analyze the working and applications of batteries and fuel cells; evaluate the characteristics, calorific value, and environmental impact of fossil fuels, synthetic fuels, and biofuels.
4. Classify polymers, understand polymerization mechanisms, and examine the properties and engineering applications of plastics, elastomers, conducting polymers, and biodegradable polymers.
5. Identify smart materials, piezoelectric materials, and biosensors; utilize spectroscopic techniques (UV-Vis, IR, Raman) for environmental and biomedical applications.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	1	1	2	-	-	-	-	**2**
CO2	3	3	2	2	1	2	-	-	-	-	**2**
CO3	3	2	3	2	2	2	-	-	-	-	**2**
CO4	3	2	2	1	1	2	-	-	-	-	**2**
CO5	3	3	2	3	2	3	-	-	-	-	**2**

END

(A500101) ENGLISH FOR SKILL ENHANCEMENT
(Common to All)

L	T	P	C
3	0	0	3

B.Tech I Year II Sem

UNIT-I: 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: Digital Transformation
Lesson on '*Emerging Technologies*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan 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: 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 Black Swan 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: Entrepreneurship
Lesson on '*Why a Start-Up Needs to Find its Customers First*' by Pranav Jain from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan Pvt. Ltd.

Vocabulary: Standard Abbreviations in English – Inferring Meanings of Words through Context – 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: Integrity and Professionalism
Lesson on '*Professional Ethics*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan 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.*

TEXTBOOKS

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

REFERENCES

1. *Practical English Usage,* Swan,Michael, Oxford University Press. New Edition..(2016).
2. *English Grammar Just for You. Karal, Rajeevan.*Oxford University Press. New Delhi (*2023).*
3. *Communication Skills –A Workbook.*Sanjay Kumar & Pushp Lata. Oxford University Press New Delhi (2022).
4. *English for Technical Communication for Engineering Students.*Vishwamohan, AyshaMc Graw-Hill Education India Pvt. Ltd.(2013)

COURSE OUTCOMES:

On completion of the course students will be able to

1. Choose appropriate vocabulary in their oral and written communication.
2. Demonstrate their understanding of the rules of functional grammar and sentence structures.
3. Develop comprehension skills from known and unknown passages.
4. Write paragraphs, essays, précis and draft letters.
5. Write abstracts and reports in various contexts.

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	-	**2**	-	-	-	-	-	-	2	-	-
CO2	-	-	-	-	-	-	-	-	-	3	2
CO3	-	-	-	-	-	-	-	-	-	2	-
CO4	-	-	-	-	-	-	-	-	3	-	2
CO5	-	-	-	-	-	-	-	-	-	3	

*****END*****

(A504201) ELECTRONIC DEVICES AND CIRCUITS

(Common to All)

L	T	P	C
3	0	0	3

B.Tech (CSE) I Year II Sem

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.

REFERENCE BOOKS:

1. Bell, David A. Electronic Devices and Circuits. Oxford University Press, 5th ed., 2008.
2. Neamen, Donald A. Electronic Circuit Analysis and Design. McGraw-Hill, 2nd ed., 2001.
3. Salivahanan, S., and N. Suresh Kumar. Electronic Devices and Circuits. McGraw-Hill Education, 4th ed., 2017.
4. Razavi, Behzad. Fundamentals of Microelectronics. Wiley, 2nd ed., 2013.
5. Taur, Yuan, and Tak H. Ning. Fundamentals of Modern VLSI Devices. Cambridge University Press, 2nd ed., 2009.

COURSE OUTCOMES:

On completion of the course students will be able to

1. Analyze the characteristics of semiconductor diodes and apply them in rectifier clippers and clipping circuits.
2. Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.
3. Design appropriate biasing networks for BJTs and determine the operating point for amplifier applications.
4. Analyze transistor amplifier circuits using h-parameter models and assess performance for various configurations.
5. Analyze the structure, working, and characteristics of JFETs, MOSFETs, and advanced devices like FinFETs and CNTFETs, and compare modern device technologies

CO-PO MAPPING

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

*****END*****

(A505301) DATA STRUCTURES
(Common to CSE, CSM and CSD)

L	T	P	C
3	0	0	3

B.Tech (CSE) I Year II Sem

UNIT – I

Introduction to Data Structures: Basic Terminology, Classification of Data Structures, Abstract data types, selecting a Data Structure,

Linear list – Introduction, singly linked list, Circular Linked Lists, Doubly Linked List, Stacks- Operations, Stack ADT, Stack applications, Queues- operations, Queue ADT, Queue Applications.

UNIT - II

Trees: Introduction, Tree – Terminology, 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 Applications, Threaded Binary Trees

UNIT – III

AVL Trees- Rotations, Operations-Insertion, Deletion & Search, Overview of: Red –Black Trees & Splay Trees,

Multi way Search Trees: Introduction, B Trees and B+ Trees

Heaps: Binary Heaps, Binomial heaps, Fibonacci heaps, Comparison of Various Heaps

UNIT - IV

Graphs: Introduction, Graph - Terminology, Representation of Graphs, Graph Traversal Algorithms, Applications of Graphs

Sorting: Radix Sort, Heap sort, Shell Sort

UNIT – V

Searching: Introduction, Interpolation Search, Jump search

Hashing and Collision: Introduction, Hash Tables, Hash Functions, Different Hash Functions: Division Method, Multiplication Method, Mid-square Method, Folding Method; collisions: Collision Resolution by Open Addressing, Collision Resolution by Chaining

TEXTBOOKS:

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

REFERENCE:

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

COURSE OUTCOMES:

On completion of the course students will be able to

1. Explain the basic concepts, classifications, and terminology of data structures and abstract data types and also Implement linear data structures such as stacks, queues, and linked lists, and apply them to solve real-world computational problems.
2. Apply binary search for efficient data storage and manipulation
3. Apply non-linear data structures like AVL trees, heaps, and multi-way trees for efficient data storage and manipulation.
4. Analyze graph traversal algorithms, sorting techniques, and their computational complexities for problem-solving.
5. Implement and evaluate searching and hashing techniques with appropriate collision resolution methods for efficient data access

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	-	3	-	-	-	-	-	-
CO2	2	3	2	-	3	-	-	-	-	-	-
CO3	2	3	3	-	3	-	-	-	-	-	-
CO4	2	3	2	3	3	-	-	-	-	-	-
CO5	2	3	2	3	3	-	-	-	-	-	-

*****END*****

(A500504) ENGLISH LANGUAGE AND COMMUNICATION SKILLS LABORATORY
(Common to All)

L	T	P	C
0	0	2	1

B.Tech (CSE) I Year II Sem

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:
1. **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: Describing Objects, Situations, 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.)*

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

Suggested Software:
1. Punctuation Made Easy by Darling Kindersley.
2. **Free Mobile App**: The official OALD 10th Edition app provides **100 free sample entries.**
3. **Free Access**: Limited to downloadable samples (table of contents, sample pages, copyright information) available on the Cambridge website.

References:

1. *Communicative English – A workbook.*Shobha, KN &Rayen, J. Lourdes. Cambridge University Press.(2019).

2. *English Language Communication Skills – Lab Manual cum Workbook.*Cengage Learning India Pvt. Ltd.(2022).
3. *Five Minute Activities – A Resource Book for Language Teachers*Ur, Penny and Wright, Andrew.Cambridge University Pres(2022).

COURSE OUTCOMES:

On completion of the course students will be able to

1. Listen actively and identify important information in spoken texts
2. Interpret the speech and infer the intention of the speaker
3. Improve their accent for intelligibility
4. Speak fluently with clarity and confidence
5. Use the language in real life situations

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	-	-	-	-	-	-	-	-	2	-	-
CO2	-	-	-	-	-	-	-	1	-	2	-
CO3	-	-	-	-	-	-	-	-	-	2	-
CO4	-	-	-	-	-	-	-	-	2	-	-
CO5	-	-	-	-	-	-	-	-	-	2	2

END

(A500503) ENGINEERING CHEMISTRY LABORATORY

(Common to EEE, ECE, CSE, CSM, CSD)

L	T	P	C
0	0	2	1

B.Tech CSE I Year I Sem

I. **Volumetric Analysis:** Estimation of Hardness of water by EDTA Complexometric method.

II. **Conductometry:**

1. Estimation of the concentration of strong acid by Conductometry.
2. Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.

III. **Potentiometry:**

1. Estimation of concentration of Fe^{+2} ion by Potentiometry using $KMnO_4$.
2. Estimation of concentration of strong acid with strong base by Potentiometry using quinhydrone

IV. **p^H Metry:** Determination of an acid concentration using p^H meter.

V. **Colorimetry:** Verification of Lambert-Beer's law using $KMnO_4$.

VI. **Preparations:**

1. Preparation of Bakelite.
2. Preparation of bioplastic from Starch.

VII. **Corrosion:** Determination of rate of corrosion of mild steel in the presence and absence of inhibitor.

VIII. **Virtual lab experiments:**

1. Construction of Fuel cell and it's working.
2. Smart materials for Biomedical applications.
3. Batteries for electric vehicles.
4. Functioning of solar cells and its applications

TEXTBOOKS:

1. Engineering Chemistry Lab manual (1st edition), J. Saroja, and D. Divya, Skytech Publishing Company (2025)
2. Lab manual for Engineering chemistry (1st edition), B. Ramadevi and P. Aparna, S Chand Publications, New Delhi (2022)

REFERENCE BOOKS:

1. Vogel's textbook of practical organic chemistry (5th edition)
2. Inorganic Quantitative Analysis (3rd edition), A.I. Vogel, ELBS Publications.
3. College Practical Chemistry (1st edition), V.K. Ahluwalia, Narosa Publications Ltd. New Delhi (2007).

VIRTUAL LABS LINKS:

1. https://www.vlab.co.in/broad-area-chemical-sciences
2. https://chemcollective.org/
3. https://phet.colorado.edu/en/simulations/filter?subjects=chemistry&type=html
4. https://www.labster.com/discipline/chemistry

COURSE OUTCOMES:

On completion of the course students will be able to

1. Estimate the hardness of water, concentrations of acids, bases, and metal ions using volumetric, conductometric, potentiometric, and p^H metric techniques.
2. Verify Lambert-Beer's law using colorimetric analysis and interpret spectrophotometric data for chemical quantification.
3. Synthesize polymers such as Bakelite and bioplastics from starch and relate their properties to real-world engineering applications in material science.
4. Evaluate the rate of corrosion of mild steel under different environments and assess the effectiveness of corrosion inhibitors.
5. Simulate the functioning of fuel cells, smart materials, batteries, and solar cells through virtual laboratory simulations and assess their engineering applications.

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	-	2	-	-	-	-	-	-
CO2	3	3	2	-	2	-	-	-	-	-	-
CO3	3	2	2	-	2	-	-	-	-	-	-
CO4	3	2	2	-	2	-	-	-	-	-	-
CO5	3	3	3	-	3	2	-	-	-	-	-

(A505507) PYTHON PROGRAMMING LABORATORY

(Common to All)

L	T	P	C
0	0	2	1

B.Tech I Year I Sem

List of Experiments:

1.
   I. Use a web browser to go to the Python website http://python.org. This page contains information about Python and links to Python-related pages, and it gives you the ability to search the Python documentation.
   II. Start the Python interpreter and type help() to start the online help utility.
1. Start a Python interpreter and use it as a Calculator.
2. Write a program to calculate compound interest when principal, rate and number of periods are given.
3. Read the name, address, email and phone number of a person through the keyboard and print the details.
4. Print the below triangle using for loop.

   5
   4 4
   3 3 3
   2 2 2 2
   1 1 1 1 1
5. Write a program to check whether the given input is digit or lowercase character or uppercase character or a special character(use 'if-else-if' ladder)
6. Python program to print all prime numbers in a given interval (use break)
7. Write a program to convert a list and tuple into arrays.
8. Write a program to find common values between two arrays.
9. Write a function called palindrome that takes a string argument and returnsTrue if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.
10. Write a function called is_sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.
11. Write a function called has_duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.
12. Write a function called remove_duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.
13. The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add "I", "a", and the empty string.
14. Write a python code to read dictionary values from the user. Construct a function to invert its content. i.e., keys should be values and values should be keys.
15. Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
16. Remove the given word in all the places in a string?
17. Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding upper case letter and the rest of the letters in the word by corresponding letters in lower case without using a built-in function?
18. Writes a recursive function that generates all binary strings of n-bit length
19. Write a python program that defines a matrix and prints
20. Write a python program to perform multiplication of two square matrices
21. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.
22. Use the structure of exception handling all general-purpose exceptions.
23. Write a function called draw_rectangle that takes a Canvas and a Rectangle as arguments and draws a representation of the Rectangle on the Canvas.
24. Add an attribute named color to your Rectangle objects and modify draw_rectangle so that it uses the color attribute as the fill color.

25. Write a function called draw_point that takes a Canvas and a Point as arguments and draws a representation of the Point on the Canvas.
26. Define a new class called Circle with appropriate attributes and instantiate a few Circle objects. Write a function called draw_circle that draws circles on the canvas.
27. Write a python code to read a phone number and email-id from the user and validate it for correctness.
28. Write a Python code to merge two given file contents into a third file.
29. Write a Python code to open a given file and construct a function to check for given words present in it and display on found.
30. Write a Python code to Read text from a text file, find the word with most number of occurrences
31. Write a function that reads a file *file1* and displays the number of words, number of vowels, blank spaces, lower case letters and uppercase letters.
32. Import numpy, Plotpy and Scipy and explore their functionalities.
33. Install NumPypackage with pip and explore it.
34. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
35. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as Submit and Reset.

TEXTBOOKS:

1. Supercharged Python: Take your code to the next level, Overland
2. Learning Python, Mark Lutz, O'reilly

REFERENCE BOOKS:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming A Modular Approach with Graphics, Database, Mobile, and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson
3. Introduction to Python Programming, Gowrishakar S, Veena A, CRC Press
4. Programming with Python, A User's Book, Michael Dawson, Cengage Learning, India Edition
5. Python for Data Science, Dr. Mohd Abdul Hameed, Wiley publications
6. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech press
7. Introduction to Python, Gowrishankar S, Veena A., CRC Press

COURSE OUTCOMES:

On completion of the course students will be able to

1. Understand and apply basic Python syntax, data types, control structures, and string operations.
2. Develop Python programs using functions, recursion, and data structures like lists, tuples, dictionaries, and arrays.
3. Implement object-oriented programming concepts and GUI applications using Python modules and Tkinter.
4. Perform file handling and text processing operations including reading, writing, searching, and analyzing textual data.
5. Utilize scientific libraries (NumPy, SciPy, Matplotlib) and design logic-based applications including digital gates and GUI tools.

CO-PO MAPPING

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

*****END*****

(A503502) ENGINEERING EXPLORATION AND PRACTICE
(Common to All)

L	T	P	C
0	0	2	1

B.Tech (CSE) I Year II Sem

Week-1

Difference between Science and Engineering, Scientist and Engineer needs and wants, various disciplines of engineering, some misconceptions of engineering, Expectation for the 21st century engineer. Significance of teamwork, Importance of communication in engineering profession

Week-2

Engineering Design Process, Need statement to Problem conversion, Pair wise comparison chart, decision matrix,Concepts of reverse engineering

Week-3

Project management tools: Checklist, Timeline, Gantt chart, Requirement Analysis

Week-4

Basic Components of a Mechanism, Degrees of Freedom or Mobility of a Mechanism, 4 Bar Chain, Crank RockerMechanism, Slider Crank Mechanism

Week-5

3-D Modelling of a Box with two holes and curvature

Week-6

3-D Modelling of Electronic Enclosure and Assembly of two parts

Week-7

Introduction to various platform-based development, Introduction to basic components, transducers, actuators andsensors, Introduction to Tinkercad

Week-8

Introduction to Arduino, basics of programming

Week-9

Interfacing Arduino with actuators and transducers

Week-10

Interfacing Arduino with Sensors, Liquid Crystal Display (LCD)

Week-11

Assembly and Crafting the Prototype

Week-12

Test and Validate the Prototype, Documentation, Panel Presentation

TEXTBOOKS:

1. Concepts in Engineering Design – 2016; by Sumesh Krishnan (Author), Dr. Mukul Shukla (Author),Publisher: Notion Press.
2. Workshop Practice, B. L. Juneja, Cengage, 2016

REFERENCE:

1. A Ghosh and AK Malik: Theory of Mechanism and Machine; East West Press (Pvt) Ltd., New Delhi.
2. Arduino Cookbook, 2nd Edition by Michael Margolis: O'Reilly Media
3. Introduction to autocad®2017-2D and 3D design by Bernd S. Palm and Alf Yarwood, Routledge

(Taylor and Francis group)
4. Engineering Fundamentals: An Introduction to Engineering (Mind Tap Course List) 5th Edition by Saeed Moaveni
5. Software Project Management (SIE), (Fifth Edition); Bob Hughes, Mike Cotterell, Rajib Mall; Published by Tata McGraw-Hill Education Pvt. Ltd (2011) ; ISBN 10: 0071072748 ISBN 13: 9780071072748

COURSE OUTCOMES:

On completion of the course students will be able to

1. Explain the importance of engineering professions in the world.
2. Identify multi-disciplinary approach required in solving an engineering problem
3. Build a mechanism for a given application
4. Create basic 3D models and animations
5. Design a mechatronic system using Mechanical and Electronic components

CO-PO MAPPING

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

*****END*****

(A505505) DATA STRUCTURES LABORATORY
(Common to CSE, CSD and CSM)

L	T	P	C
0	0	2	1

B.Tech (CSE) I Year II Sem

Basic Programs:

1. Write a C program to implement the following operations on to a 1D Array:
   a. INSERT b. DELETE c. SEARCH d. TRAVERSE
2. Write a C program to implement Self-referential Structure.
3. Write a C program to Perform Dynamic Memory Allocation.

Linked List:

1. Write a C program to implement Single linked list i) Insertion ii) Deletion iii)Display
2. Write a function to reverse the nodes of a Single linked list

Additional:

1. Write a program that uses functions to perform the following operations on doubly linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on circular linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal

Stacks & Queues:

1. Write a program that implements Stack (its operations) using Arrays
2. Write a program that implements Queue (its operations) using Arrays
3. Write a program that implements Circular Queue (its operations) using Arrays

Additional:

1. Write C programs to implement Stack ADT using Linked List
2. Write C programs to implement Queue ADT using Linked List
3. Write C programs to implement Circular Queue ADT using Linked List

Applications of Stacks:

1. Write a C program to Convert the given Infix Expression to Postfix Expression.
2. Write a C program to Evaluate the given Postfix Expression.

Trees:

1. Write a C program to implement Binary search tree
   i) Insertion ii) deletion iii) Traversals
2. Write a C program to implement binary search tree traversals:
   i) Pre- Order ii) Post –Order iii) In-Order
3. (A) Write a C Program to Check if a Given Binary Tree is an AVL Tree or Not
   (B) Write a C program to find height of a Binary tree

TEXTBOOKS:

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:

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

COURSE OUTCOMES:

On completion of the course students will be able to

1. Explain the importance of engineering professions in the world.
2. Identify multi-disciplinary approach required in solving an engineering problem

3. Build a mechanism for a given application
4. Create basic 3D models and animations
5. Design a mechatronic system using Mechanical and Electronic components

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	-	3	-	-	-	-	-	-
CO2	3	3	3	-	3	-	-	-	-	-	-
CO3	3	3	3	-	3	-	-	-	-	-	-
CO4	2	3	3	3	3	-	-	-	-	-	-
CO5	2	3	2	3	3	-	-	-	-	-	-

*****END*****

(A500006) COMPUTER ORIENTED STATISTICAL METHODS

L	T	P	C
3	0	0	3

B.Tech (CSE) II Year I Sem

UNIT-I: Random Variables and Probability Distributions
Concept of a Random Variable – Discrete Probability Distributions – Continuous Probability Distributions – Mean of a Random Variable – Variance of a Random Variable

Discrete Probability Distributions: Binomial Distribution – Poisson distribution

UNIT-II: Continuous Distributions and sampling
Uniform Distribution – Normal Distribution – Areas under the Normal Curve – Applications of the Normal Distribution – Normal Approximation to the Binomial Distributions. **Fundamental Sampling Distributions:** Random Sampling – Some Important Statistics – Sampling Distributions – Sampling Distribution of Means – Central Limit Theorem.

UNIT-III: Estimation &Tests of Hypotheses (Large Samples)
Introduction – Statistical Inference – Classical Methods of Estimation – Single Sample: Estimating the mean – Standard error of a point estimate. Statistical Hypotheses: General Concepts – Testing a Statistical Hypothesis. Single sample: Tests concerning a single mean. Two samples: Tests on two mean (Unknown for equal variance). One sample: Test on a single proportion. Two samples: Tests on two proportions.

UNIT-IV: Tests of Hypotheses (Small Samples)
Students t-distribution, its properties: Test of significance difference between sample mean and population means; difference between means of two small samples. Snedecor's F-distribution and its properties. Two- sample tests concerning variances: F-distribution.

UNIT-V: Stochastic Processes and Markov Chains
Introduction to Stochastic processes- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain, Steady state condition, Markov analysis.

TEXTBOOKS
1. Probability & Statistics for Engineers & Scientists (9th Edition), Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Pearson Publishers.
2. Fundamentals of Mathematical statistics, S C Gupta and V K Kapoor, Khanna publications.

REFERENCES
1. Fundamentals of Probability and Statistics for Engineers, T.T. Soong, John Wiley & Sons, Ltd, 2004.
2. Probability and statistics for Engineers and scientists, Sheldon M Ross, academic press

COURSE OUTCOMES:
On completion of the course students will be able to
1. Apply the concepts of Random variable and distributions to case studies.
2. Formulate and solve problems involving random variables and apply statistical methods for analysing experimental data.
3. Apply the concepts of statistical estimation to real-world data.
4. Perform hypothesis testing in the context of practical case studies.
5. Correlate the concepts of one unit to the concepts in other units.

CO-PO MAPPING

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

END

(A505302) COMPUTER ORGANIZATION AND ARCHITECTURE

B. Tech (CSE): II Year I Semester

L	T	P	C
3	0	0	3

UNIT - I:

Boolean Algebra and Logic Gates: Binary codes, Binary Storage and Registers, Binary logic. **Digital logic gates. Data Representation**: Data types, Complements, Fixed Point Representation, Floating Point Representation

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

UNIT - II:

Combinational Logic: Combinational Circuits, Analysis procedure Design procedure, Binary Adder-Subtractor Decimal Adder, Binary multiplier, magnitude comparator, Decoders, Encoders, Multiplexers, HDL for combinational circuits.

Sequential Logic: Sequential circuits, latches, Flip-Flops Analysis of clocked sequential circuits, state Reduction and Assignment, Design Procedure**.** Registers, shift Registers, Ripple counters, synchronous counters, other counters.

UNIT III

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, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT - IV

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

UNIT - V

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.

TEXTBOOKS:

1. Digital Design – M. Morris Mano, Third Edition, Pearson/PHI.
2. Computer System Architecture – M. Morris Mano, Third Edition, Pearson/PHI.

REFERENCE BOOKS:

1. Switching and Finite Automata Theory, ZVI. Kohavi, Tata Mc Graw Hill.
2. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5th Edition, McGraw Hill.
3. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson

Course Outcomes:

On completion of the course students will be able to

1. Understand the basics of instruction sets and their impact on processor design.
2. Demonstrate an understanding of the design of the functional units of a digital computer system.
3. Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
4. Design a pipeline for consistent execution of instructions with minimum hazards.
5. Recognize and manipulate representations of numbers stored in digital computers

CO-PO Mapping:

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

*****END*****

(A505303) OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B. Tech (CSE): II Year I Semester

L	T	P	C
3	0	0	3

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, inner classes.

UNIT - II

Inheritance, Packages and Interfaces – Hierarchical abstractions, Base class object, subclass, types of inheritance 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 multiprocessing, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads.

UNIT - IV

Exploring String class, Object class, Exploring java.util package, Exploring java.io package

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

UNIT - V

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, JRadioButton MenuItem, JSeperator. creating a popup menu. Collection Framework in Java: Introduction to Java Collections, Overview of Java Collection Frame Work, Generics, Commonly used Collection Classes-Array List, Vector, Hash Table, Stack, Enumeration, Iterator, String Tokenizer, Random, Scanner, Calendar and Properties.

TEXTBOOKS:

1. Java the complete reference, 13th edition, Herbert schildt, Dr. Denny Coward, Mc Graw Hill.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson education.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John Wiley& sons.
2. An Introduction to OOP, third edition, T. Budd, Pearson education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson education.
4. An introduction to Java programming and object-oriented application development, R.A. Johnson-Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.

8. Java and Object Orientation, an introduction, John Hunt, second edition, Springer.

Course Outcomes:

On completion of the course students will be able to

1. Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, garbage collection.
2. Demonstrate the implementation of inheritance (multilevel, hierarchical and multiple) by using extend and implement keywords
3. Use multithreading concepts to develop inter process communication.
4. Demonstrate how Java handles events using event listeners and adapter classes.
5. Understand and explain the architecture of the Java Collection Framework, and evaluate the suitability of various collection types

CO-PO Mapping:

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

END

(A505304) SOFTWARE ENGINEERING

B. Tech (CSE): II Year I Semester

L	T	P	C
3	0	0	3

UNIT – I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths. A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI). **Process models:** The waterfall model, Spiral model, Incremental Process Models, Concurrent Models, Component based development and Agile methodology.

UNIT – II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT – III

Design Engineering: Design process and design quality, design concepts, the design model. **Creating an architectural design:** software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, activity diagrams, use case diagrams, component diagrams.

UNIT – IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white- box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT – V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM. Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXTBOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.

REFERENCE BOOKS:

1. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

Course Outcomes:

On completion of the course students will be able to

1. Understand the basic role and nature of software and acquire knowledge to select process framework and process model for business system.
2. Ability to translate end-user requirements into system and software requirements and able to identify, analyse, validate, manage, and specify the requirements for the development of an application.
3. Demonstrate an ability to use the Design tools for designing software architecture and conceptual model through UML diagram.
4. Concede product quality through testing techniques employing appropriate metrics by understanding the practical challenges associated with the development of a significant software system.

5. Apply the Risk management strategies, Quality management strategies and Quality Assurance knowledge for handling the Application.

CO-PO Mapping:

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

*****END*****

(A505305) DATABASE MANAGEMENT SYSTEMS
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester

L	T	P	C
3	0	0	3

Prerequisites: A course on "Data Structures".

UNIT - I

Database System Applications: A Historical Perspective, File Systems versus a DBMS, the Data Model, 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.

UNIT - III

SQL: QUERIES, 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.

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 - IV

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 and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions.

UNIT - V

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing (static and extensible hashing) Treebased Indexing, Comparison of File Organizations, Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM),B+ Trees: A Dynamic Index Structure.

TEXTBOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

Course Outcomes:

On completion of the course students will be able to

1. Understand DBMS architecture, data models, and conceptual database design using ER diagrams.
2. Apply relational model concepts and perform queries using relational algebra and calculus.
3. Write complex SQL queries and apply normalization for schema refinement.
4. Explain transaction management, concurrency control, and recovery techniques in DBMS.
5. Analyze file organization and indexing methods for efficient data storage and retrieval.

CO-PO Mapping:

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

END

(A500501) COMPUTATIONAL MATHEMATICS LABORATORY
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester	L	T	P	C
	0	0	2	1

Prerequisites: A course on "Matrices, Iterative methods and ordinary differential equations".

UNIT - I: Eigen values and Eigenvectors: **6P**

Programs:

- Finding real and complex Eigen values.
- Finding Eigen vectors.

UNIT-II: Solution of Algebraic and Transcendental Equations **6P**

Bisection method, Newton Raphson Method

Programs:

- Root of a given equation using Bisection method.
- Root of a given equation Newton Raphson Method.

UNIT-III: Linear system of equations: **6P**

Jacobi's iteration method and Gauss-Seidal iteration method

Programs:

- Solution of given system of linear equations using Jacobi's method
- Solution of given system of linear equations using Gauss-Seidal method

UNIT-IV: First-Order ODEs **8P**

Exact and non-exact equations, Applications: exponential growth/decay, Newton's law of cooling.

Programs:

- Solving exact and non-exact equations
- Solving exponential growth/decay and Newton's law of cooling problems

UNIT-V: Higher order linear differential equations with constant coefficients **6P**

Programs:

- Solving homogeneous ODEs
- Solving non-homogeneous ODEs

TEXTBOOKS:

1. MATLAB and its Applications in Engineering, Rajkumar Basal, Ashok Kumar Geo, Manoj Kumar Sharma, Pearson publication.
2. Kenneth A. Lambert, The fundamentals of Python: First Programs, 2011, Cengage Learnings.
3. Think Python First Edition, by Allen B. Downey, Orielly publishing.
4. Introduction to Python Programming, William Mitchell, Povel Solin, Martin Novak et al., NCLab Public Computing, 2012.
5. Introduction to Python Programming, ©Jacob Fredslund, 2007.

REFERENCE BOOKS:

1. An Introduction to Python, John C. Lusth, The University of Alabama, 2011.
2. Introduction to Python, ©Dave Kuhlman, 2008.

Course Outcomes:

On completion of the course students will be able to

1. Understand and compute eigenvalues and eigenvectors of matrices, including both real and complex cases, and apply these concepts to solve related mathematical problems.
2. Apply numerical methods such as the Bisection method and Newton-Raphson method to find roots of algebraic and transcendental equations with accuracy.
3. Solve systems of linear equations using iterative methods like Jacobi and Gauss-Seidel methods and analyze their convergence properties.
4. Formulate and solve first-order ordinary differential equations (ODEs), including exact and non-exact equations, and apply these to real-world problems such as exponential growth/decay and Newton's law of cooling.
5. Solve higher-order linear differential equations with constant coefficients, both homogeneous and non-homogeneous, using appropriate analytical techniques.

CO-PO Mapping:

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

*****END*****

(A505508) OBJECT ORIENTED PROGRAMMING THROUGH JAVA LABORATORY

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester

L	T	P	C
0	0	2	1

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint yourself with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. 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. Handle any possible exceptions like divided by zero.
3. A) Develop an applet in Java that displays a simple message.
   B) Develop an applet in Java that receives an integer in one text field, and computes its factorial
4. Value and returns it in another text field, when the button named "Compute" is clicked.
5. Write a Java 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 Num 2 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.
6. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.
7. Write a Java program for the following:
   - Create a doubly linked list of elements.
   - Delete a given element from the above list.
   - Display the contents of the list after deletion.
8. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in the selected color. Initially, there is no message shown.
9. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
10. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas.
11. Write a java program to display the table using Labels in Grid Layout.
12. 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 (Use Adapter classes).
13. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: 16. use hash tables).
14. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.
15. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

TEXTBOOKS:

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.

REFERENCE BOOKS:

1. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
2. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

Course Outcomes:

On completion of the course students will be able to

1. Use IDEs like Eclipse/NetBeans to develop, debug, and refactor Java programs efficiently.
2. Design GUI applications and applets using Java AWT and Swing components.
3. Implement multithreading and inter-thread communication for concurrent programming.
4. Apply object-oriented programming concepts such as inheritance, abstraction, and exception handling.
5. Handle file I/O operations, event handling, and data structures like hash tables and linked lists in Java.

CO-PO Mapping:

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

*****END*****

(A505509) SOFTWARE ENGINEERING LABORATORY
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester

L	T	P	C
0	0	2	1

Prerequisites: A course on "Programming for Problem Solving".

List of Experiments:

Do the following seven exercises for any two projects given in the list of sample projects or any other

Projects:

1. Development of problem statements.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Preparation of Software Configuration Management and Risk Management related documents.
4. Study and usage of any Design phase CASE tool
5. Performing the Design by using any Design phase CASE tools.
6. Develop test cases for unit testing and integration testing
7. Develop test techniques for various white box and black box testing cases.

Sample Projects:

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
8. Smart Farming Recommendation System
9. Deaf & Mute Inclusive Learning and Employment Platform
10. Real-Time Blood Donation Platform
11. Surplus Food Distribution System

TEXTBOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill

Course Outcomes: The student will learn to

1. Understand the software development life cycle and identify appropriate models for different types of software projects.
2. Prepare key software engineering documents including SRS, design documents, and testing plans.
3. Apply CASE tools for software design and modeling.
4. Design and develop test cases for unit, integration, white-box, and black-box testing.
5. Analyze and manage software configuration and risks in software projects.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	-	-	3	-	-	-	2	1	2	3	-
CO2	3	3	2	2	3	-	-	-	2	2	2	3	2
CO3	3	3	2	2	3	-	-	-	2	2	2	3	2
CO4	3	3	2	2	3	-	-	-	2	2	2	3	2
CO5	3	3	2	2	3	2	-	-	2	2	2	3	2

*****END*****

(A505510) DATABASE MANAGEMENT SYSTEMS LABORATORY
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester

L	T	P	C
0	0	2	1

Prerequisites: A course on "Database Management Systems".

List of Experiments:

1. Concept design with E-R Model

Draw an ER diagram for the following

a. There are two entity sets **Employee** and **Department**. These entity sets are participating in a relationship **works in.** The relationship set is converted into relation with attributes EmpNo from **Employee** relation, D_id from **Department** relation and **Since**, the attribute of the relationship set itself.

2. Relational Model

a. Convert the above ER diagram into Relational Model

3. Normalization

Consider the following table.

a. Normalize the given Relation. Consider the given schema is in first normal form and
Schema (Student id, Student name, Project Id, Project name, City, country, ZIP)
Primary key(Student id, Project id)
Fd's: Project Id-→ Project name
country-→ZIP,ZIP

b. Normalize the given Table to the BCNF
Consider the set of Functional Dependencies.
Eid → EName, Ph.no, Empcity, CityZip PrimaryKey = (EmpId, ProjId)
ProjId → ProjName, ProjLeader

Eid	E.Name	Ph.no	Proj Id	Proj Name	Proj Leader	Emp City	City Zip
101	John	98765623,998234123	P03	Project103	Grey	ModelTown	110033
102	John	89023467	P01	Project101	Christian	Badarpur	110044
103	Ryan	76213908	P04	Project104	Hudson	Naraina	110028
104	stephine	98132452	P02	Project102	Petro	HariNagar	110064

EmpId,ProjId → ProjLeader
EmpCity → CityZip
ProjId → ProjLeader

4. Practicing DDL commands

5. Practicing DML commands

1. a. Create a user and grant all permissions to the user.
   b. Insert any three records in the employee table and use rollback. Check the result.
   c. Add a primary key constraint and not null constraint to the employee table.
   d. Insert null values to the employee table and verify the result.
2. a. Create a user and grant all permissions to the user.
   b. Insert values in the department table and use commit.
   c. Add constraints like unique and not null to the department table.
   d. Insert repeated values and null values into the table.
3. a. Create a user and grant all permissions to the user.
   b. Insert values into the table and use commit.
   c. Delete any three records in the department table and use rollback.
   d. Add constraint primary key and foreign key to the table

4. a. Create a user and grant all permissions to the user.
   e. Insert records in the sailor table and use commit.
   f. Add save point after insertion of records and verify save point.
   g. Add constraints not null and primary key to the sailor table.
5. a. Create a user and grant all permissions to the user.
   b. Use revoke command to remove user permissions.
   c. Change password of the user created.
   d. Add constraint foreign key and not null.
6. a. Create a user and grant all permissions to the user.
   b. Update the table reserves and use savepoint and rollback.
   c. Add constraint primary key , foreign key and not null to the reserves table
   d. Delete constraint not null to the table column

6. (A) Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)
(B) Nested, Correlated subqueries

a. Find the Sid's of sailors who have reserved a red or a green boat.
b. Find the names of sailors who have reserved a red and a green boat.
c. Find the names of sailors who have reserved a red but not green boats.
d. Find all sids of sailors who have a rating of 10 or reserved boat 104.
e. Find the names of sailors who have reserved boat 103 using independent nested query.
f. Find the names of sailors who have reserved a red boat.
g. Find the names of sailors who have not reserved a red boat.
c. Find the names of sailors who have reserved boat number 103 using correlated nested query.
a. Find sailors whose rating is better than some sailor called 'Horatio'.
b. Find sailors with the highest rating.
d. Find the names of sailors who have reserved both a red and a green boat using nested queries.
a. Find the names of sailors who have reserved all boats.

7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.

a. who is the youngest sailor
b. Find the name of the sailor who have maximum rating
c. What is the average rating of all Sailors
d. how many sailors are there with the rating above 7
e. The following SQL lists the number of customers in each country, sorted high to low
f. The following SQL lists the number of customers in each country, sorted high to low (Only include countries with more than 5 customers):

8. Triggers (Creation of insert trigger, delete trigger, update trigger)

1.

a. Create a pl/sql trigger which will calculate the total marks and percentage of students after insert/update the details of a student in database.

b. Write a trigger that keeps backup of deleted records of emp_trig table. Deleted records of emp_trigger inserted in emp_backup table.

2. Creation of insert trigger, delete trigger, update trigger practice triggers using the passenger database.

Passenger (Passport_ id INTEGER PRIMARY KEY, Name VARCHAR (50) Not NULL, Age Integer Not NULL, Sex Char, Address VARCHAR (50) Not NULL);

a. Write a Insert Trigger to check if the Passport_id is exactly six digits or not.

b. Write a trigger on passenger to display messages '1 Record is inserted', '1 record is deleted', '1 record is updated' when insertion, deletion and updation are done on passenger respectively.

9. Procedures and functions

a. Create a procedure which displays employee salary for given employee number using out variable
b. Write a pl/sql block which displays the department name for department 40.

c. Create a procedure to check whether the given number is prime or not
d. Create a function which returns week day of a given date
e. Create a function which returns number of sailors for a given rating level.
f. Create a procedure to find the lucky number of a given birth date
g. Create a function which returns average age of sailors for a given rating level.

10. Usage of Cursors

a. Display the employee names and their salary for the accepted department number.
b. Display the top N earners for an accepted department number.
c. To write a Cursor to display the list of employees who are working as a Managers or Analyst.
d. Write a Cursor to find employee with given job and deptno.
e. Write a PL/SQL block using implicit cursor that will display message, the salaries of all the employees in the 'employee' table are updated. If none of the employee's salary are updated we get a message 'None of the salaries were updated'. Else we get a message like for example, 'Salaries for 1000 employees are updated' if there are 1000 rows in 'employee'table.

11. Packages

a. creates HR package which contains Hire and Fire functions.
Hire function adds the details of employee and Fire function deletes the details of Employee.

TEXTBOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill, 3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

REFERENCE BOOKS:

1. An Introduction to Python, John C. Lusth, The University of Alabama, 2011.
2. Introduction to Python, ©Dave Kuhlman, 2008.
3. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
4. Fundamentals of Database Systems, ElmasriNavrate, Pearson Education
5. Introduction to Database Systems, C.J. Date, Pearson Education
6. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
7. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
8. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

Course Outcomes:

On completion of the course students will be able to

1. Understand and design conceptual database schemas using ER models and convert them into relational models.
2. Apply normalization techniques to minimize redundancy and improve database design up to BCNF.
3. Write SQL queries for data definition, manipulation, and complex retrieval using joins, subqueries, and aggregate functions.
4. Implement transaction management, concurrency control, and recovery mechanisms to ensure database consistency and reliability.
5. Develop and use triggers, stored procedures, functions, cursors, and packages for advanced database programming and automation.

CO-PO Mapping:

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

*****END*****

(A505701) SKILL DEVELOPMENT COURSE (NODE JS, REACT JS, and DJANGO)
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year I Semester

L	T	P	C
0	0	2	1

Prerequisites: A course on "Object Oriented Programming through Java, HTML Basics.".

List of Experiments:

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 java stand alone application that connects with the database (Oracle / mySql) and perform the CRUD operation on the database tables.
6. Create an xml for the bookstore. Validate the same using both DTD and XSD.
7. Design a controller with servlet that provides the interaction with application developed in experiment 1 and the database created in experiment 5.
8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism (Cookies, HTTP Session)
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 JavaScript, 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 MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node ,2nd Edition, APress.

Course Outcomes:

On completion of the course students will be able to

1. Build a custom website with HTML, CSS, and Bootstrap and little JavaScript.
2. Demonstrate Advanced features of JavaScript and learn about JDBC
3. Develop Server – side implementation using Java technologies like
4. Develop the server – side implementation using Node JS.
5. Design a Single Page Application using React.

CO-PO Mapping:

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

END

(A505306) DISCRETE MATHEMATICS

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
3	0	0	3

UNIT - I

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 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.

TEXTBOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe l. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2nd ed.

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

Course Outcomes:

On completion of the course students will be able to

1. Understand and construct precise mathematical proofs
2. Apply logic and set theory to formulate precise statements
3. Analyze and solve counting problems on finite and discrete structures
4. Describe and manipulate sequences
5. Apply graph theory in solving computing problems

CO-PO Mapping:

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

END

(A505307) OPERATING SYSTEMS
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
3	0	0	3

Prerequisites: A course on "Computer Programming and Data Structures and Computer Organization and Architecture".

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 Algorithms.

UNIT - V

File System Interface and Operations -Access methods, Directory Structure, Protection, File System 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, Pearson education.

REFERENCE BOOKS:

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/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

Course Outcomes:

On completion of the course students will be able to

1. Apply space and time complexity analysis using asymptotic notations.
2. Design divide-and-conquer algorithms and critically assess their runtime and space trade-offs.
3. Device backtracking and dynamic programming solutions.
4. Apply greedy methods and graph traversal algorithms
5. Analyze and Design branch-and-bound algorithms for NP-hard problems

CO-PO Mapping:

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

END

(A505308) ALGORITHMS DESIGN AND ANALYSIS

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
3	0	0	3

Prerequisites: A course on "Programming for problem solving and Data Structures".

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

Disjoint Sets: Disjoint set operations, union and find algorithms, Priority Queue- Heaps, Heapsort

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

UNIT - III

Dynamic Programming: General method, applications- Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson problem, Reliability design.

UNIT - IV

Greedy method: General method, applications- Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected components, Biconnected components.

UNIT - V

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

NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's

TEXTBOOKS:

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.

Course Outcomes:

On completion of the course students will be able to

1. Apply space and time complexity analysis using asymptotic notations.
2. Design divide-and-conquer algorithms and critically assess their runtime and space trade-offs.
3. Device backtracking and dynamic programming solutions.
4. Apply greedy methods and graph traversal algorithms
5. Analyze and Design branch-and-bound algorithms for NP-hard problems

CO-PO Mapping:

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

END

(A505309) COMPUTER NETWORKS
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
3	0	0	3

UNIT - I

Introduction: The Internet, Protocol, Network Edge, Access Networks, Network Core, Packet Switching, Circuit Switching, Delay, oss, and Throughput in Packet-Switched Networks, **Protocol reference models:** ISO-OSI, TCP/IP, Types of Network attacks, History of Computer Networking and the Internet.

UNIT-II

Application Layer: Principles of Network Applications, Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols, **The Web and HTTP, File Transfer:** FTP, Electronic Mail in the Internet, SMTP, DNS, Peer-to-Peer Applications, Socket Programming: Creating Network Applications.

UNIT - III

Transport Layer: Transport-Layer Services, Multiplexing and Demultiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N (GBN), Selective Repeat (SR), **Connection-Oriented Transport:** TCP, The TCP Connection, Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control, TCP Congestion Control, Fairness.

UNIT - IV

Network Layer: Data and Control plane, Forwarding and Routing, Network Service Models, Virtual Circuit and Datagram Networks, Router working,

The Internet Protocol (IP): Forwarding and Addressing in the Internet, Datagram Format, IPv4 Addressing, Internet Control Message Protocol (ICMP), IPv6, IP Security, Routing Algorithms- The Link-State (LS) Routing Algorithm, The Distance Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet-Intra-AS Routing in the Internet: RIP, **Intra-AS Routing in the Internet**: OSPF, Inter-AS Routing: BGP, Broadcast and Multicast Routing, Broadcast Routing Algorithms, Multicasting.

UNIT - V

The Link Layer: The Services Provided by the Link Layer, Error-Detection and -Correction Techniques Parity Checks, Checksum Methods, Cyclic Redundancy Check (CRC), Hamming code, Multiple Access
Links and Protocols, Channel Partitioning Protocols, Random Access Protocols, Taking-Turns Protocols,

DOCSIS: The Link-Layer Protocol for Cable Internet Access, Switched Local Area Networks, Link-Layer Addressing and ARP, Ethernet, Link-Layer Switches, Virtual Local Area Networks (VLANs), Link Virtualization-Multiprotocol Label Switching (MPLS), Data Center Networking, A Day in the Life of a Web Page Request. Wireless network characteristics, Wireless LAN.

TEXTBOOKS:

1. Computer Networking: A Top-Down Approach – James F.Kurose, Keith W. Ross, Pearson
2. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson/PHI

REFERENCE BOOKS:

1. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

Course Outcomes:

On completion of the course students will be able to

1. Gain the knowledge of the basic computer network technology and understand the functions of each layer in the ISO-OSI and TCP/IP reference model.
2. Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation
3. Understand the end-to-end communication between applications running on different devices over a network
4. Obtain the skills of subnetting and routing mechanisms.
5. Understanding working of the link layer for error handling and physical access of the network

CO-PO Mapping:

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

END

(A566302) MACHINE LEARNING
(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
3	0	0	3

UNIT - I
Introduction to Machine Learning: Types of Human learning, machine learning process, Well-posed learning problem, Types of machine learning and comparison, applications of machine learning.
Model Preparation, Evaluation and feature engineering: Machine learning activities, Types of data in machine learning, dataset understanding, plotting and exploration, checking data quality, remediation, data pre-processing, selecting a model, predictive and descriptive models, supervised learning model training, cross-validation and boot strapping, lazy vs eager learner, interpreting the model- underfitting, overfitting, bias-variance trade-off. Parameter for evaluating performance of classification, regression, and clustering model. Improving performance of a model.

UNIT - II
Feature Engineering: Feature transformation - feature construction, feature extraction by PCA, SVD,LDA. Feature subset selection – feature relevancy and redundancy measures. Feature selection process and approaches.
Review of Probability concepts: joint probability, conditional probability, bayes rule, Common discrete and continuous distributions, dealing with multiple random variables, central limit theorem. Bayes classifier, Multi-class Classification, Naïve Bayes classifier, Bayesian belief network.

UNIT - III
Supervised Learning - Introduction to supervised learning, Regression: Introduction of regression, Regression algorithms: Simple linear regression, Multiple linear regression, Polynomial regression model, Logistic regression, Maximum likelihood estimation.
Classification: Classification model and learning steps, Classification algorithms: Naïve Bayes classifier, Distance measures, k-Nearest Neighbor (kNN), Decision tree, Support vector machines, Kernel trick, Random Forest.

UNIT - IV
Unsupervised Learning: Introduction to unsupervised learning, Unsupervised vs supervised learning, Application of unsupervised learning, Clustering and its types, Partitioning method: k-Means and KMedoids, Hierarchical clustering, Density-based methods – DBSCAN.

UNIT - V
Artificial Neural Network: Biological neuron, Artificial neuron, Activation functions, neural network architecture, perceptron, learning process in ANN, Back propagation.
Introduction to deep learning, overview of reinforcement learning, Representation learning, Evolutionary learning.
Case-study of ML applications: Image recognition, Email spam filtering, Online fraud detection.

TEXTBOOKS:

1. Saikat Dutt, S. Chjandramouli, Das – Machine Learning, Frist Edition, Pearson
2. M N Murty, Anathanarayana V S – Machine Learning, First Edition, University Press
3. Tom M Mitchell, Machine Learning, First Edition, McGraw Hill Education, 2013.

REFERENCE BOOKS:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis
2. Stephen Marsland, ―Machine Learning – An Algorithmic Perspective, Second Edition,
3. Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.

Course Outcomes:

On completion of the course students will be able to

1. Understand and compare different types of machine learning, their processes, and applications.
2. Apply data preprocessing, feature engineering, and model selection techniques for ML tasks.
3. Implement supervised learning algorithms for regression and classification problems.
4. Utilize unsupervised learning methods for clustering and pattern discovery.
5. Develop and evaluate neural network models and explore advanced ML paradigms like deep learning and reinforcement learning

CO-PO Mapping:

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

*****END*****

(A500507) SOCIAL INNOVATION AND ENTREPRENEURSHIP
(Common to all branches)

L	T	P	C
0	1	2	2

B.Tech CSE II Year I Sem

Week-1
Identify community issues to be addressed, Requirements Analysis: Extensive User requirements analysis, Generating effective System Requirement document
Week-2
Introduction to Innovation & Entrepreneurship, Innovation vs. Invention vs. Creativity, Types of Entrepreneurs (Tech, Social, Green)
Week-3
Social Innovation – Case Studies, Impact of Social Innovation on communities
Week-4
Process of Social Innovation Prompts – identifying needs, Proposals –generating ideas,Prototyping – testing the idea in practice,
Week-5
Sustaining-developing a business model, Scaling and diffusion-growing social innovations, Systematic change
Week-6
Introduction to sustainability, Sustainability leadership, Life cycle assessment, Carbon foot print calculation
Week-7
Business Model & Start-Up Ecosystem Elements of a business model (Canvas model)
Week-8
Identify and map global competitors, review industry trends, and understand market sizing: TAM, SAM, and SOM. Assessing scope and potential scale for the opportunity
Week-9
Types of Start-Ups, Market analysis and feasibility Minimum Viable Product (MVP), Market risks and Marketing strategies, legal aspects in startup, National Innovation Startup Policy (NISP) and its features
Week-10
Government schemes for startups (Startup India, Atal Innovation Mission) Incubators, accelerators
Week-11
Financial planning, budgeting, and cost estimation for the Business model
Week-12
Funding options: Bootstrapping, Angel investors, venture pitching readiness, Documentation, Panel Presentation

TEXT BOOKS:

1. “Innovation and Entrepreneurship” by Peter F.Drucker
2. “Entrepreneurship Development” by S.S.Khanka
3. “Design Thinking” by Tim Brown

REFERENCE BOOKS:

1. AICTE Innovation Cell & Startup India Toolkit
2. Social Enterprise Law: Trust, Public Benefit and Capital Markets By Dana Brakman Reiser & Steven A. Dean
3. Introduction to Sustainability by Robert Brinkmann, Wiley-Blackwell

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

1. Understand the fundamentals of innovation, creativity, and entrepreneurs
2. Develop innovative solutions to community issues
3. Assess market competition, estimate market size, and develop a prototype.
4. Develop a scalable business model
5. Analyze Business and financial planning models and Go-to-Market strategies.

CO-PO MAPPING

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

(A505511) OPERATING SYSTEMS LABORATORY

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
0	0	2	1

List of Experiments:

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, lseek, stat, fork, exit)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques a) Paging b) Segmentation
7. Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal

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, Pearson education.

REFERENCE BOOKS:

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

Course Outcomes:

On completion of the course students will be able to

1. Simulate CPU scheduling algorithms to understand process management and performance.
2. Use UNIX/Linux system calls for file and process operations.
3. Implement deadlock avoidance and synchronization techniques using semaphores.
4. Demonstrate interprocess communication using pipes, message queues, and shared memory.
5. Simulate memory management and page replacement algorithms for efficient resource utilization.

CO-PO Mapping:

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

END

(A505512) COMPUTER NETWORKS LABORATORY

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
0	0	2	1

List of Experiments:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijsktra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. Wireshark
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.

1. How to run Nmap scan

2. Operating System Detection using Nmap

3. Do the following using NS2 Simulator

I. NS2 Simulator-Introduction

II. Simulate to Find the Number of Packets Dropped

III. Simulate to Find the Number of Packets Dropped by TCP/UDP

IV. Simulate to Find the Number of Packets Dropped due to Congestion

V. Simulate to Compare Data Rate & Throughput.

VI. Simulate to Plot Congestion for Different Source/Destination

VII. Simulate to Determine the Performance with respect to Transmission of Packets

TEXTBOOKS:

1. Computer Networks, Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI

REFERENCE BOOKS:

1. An Engineering Approach to Computer Networks, S.Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking – Behrouz A. Forouzan. 3rd Edition, TMH.

Course Outcomes:

On completion of the course students will be able to

1. Implement data link layer protocols including framing, CRC, and flow control mechanisms.
2. Apply routing algorithms such as Dijkstra's and distance vector to simulate network pathfinding and routing tables.
3. Demonstrate congestion control and buffer management techniques using algorithms like Leaky Bucket and frame sorting.
4. Perform network traffic analysis and scanning using tools like Wireshark and Nmap.
5. Simulate network performance metrics and packet transmission using NS2 simulator

CO-PO Mapping:

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

END

(A505513) MACHINE LEARNING LABORATORY

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
0	0	2	1

Prerequisites: A course on "Programming for Problem Solving and Computer Organization and Architecture".

List of Experiments:

1. Write a python program to compute Central Tendency Measures: Mean, Median, Mode, Measure of Dispersion: Variance, Standard Deviation
2. Study of Python Basic Libraries such as Statistics, Math, Numpy and Scipy
3. Study of Python Libraries for ML application such as Pandas and Matplotlib
4. Write a Python program to implement Simple Linear Regression
5. Implementation of Multiple Linear Regression for House Price Prediction using sklearn
6. Implementation of Decision tree using sklearn and its parameter tuning
7. Implementation of KNN using sklearn
8. Implementation of Logistic Regression using sklearn
9. Implementation of K-Means Clustering
10. Performance analysis of Classification Algorithms on a specific dataset (Mini Project)

TEXTBOOKS:

1. Machine Learning – Tom M. Mitchell, - MGH.

REFERENCE BOOKS:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis

Course Outcomes:

On completion of the course students will be able to

1. Apply Python programming to compute statistical measures and explore basic libraries for scientific computing.
2. Use Python libraries such as Pandas and Matplotlib for data analysis and visualization in ML applications.
3. Implement regression techniques including simple and multiple linear regression using scikit-learn.
4. Apply classification algorithms such as Decision Tree, KNN, and Logistic Regression using scikit-learn.
5. Perform clustering and evaluate classification models on real-world datasets through mini projects.

CO-PO Mapping:

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

END

(A505702) SKILL DEVELOPMENT COURSE DATA VISUALIZATION

(Common to CSE, CSM, and CSD)

B. Tech (CSE): II Year II Semester

L	T	P	C
0	0	2	1

Lab Problems:

1. Understanding Data, what is data, where to find data, Foundations for building Data Visualizations, Creating Your First Visualization?
2. Getting started with Tableau Software using Data file formats, connecting your Data to Tableau,
3. creating basic charts(line, bar charts, Tree maps), Using the Show me panel.
4. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.
5. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.
6. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data.
7. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.
8. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.
9. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.
10. Tableau file types, publishing to Tableau Online, Sharing your visualizations, printing, and Exporting.
11. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.

REFERENCE BOOKS:

1. Microsoft Power BI cookbook, Brett Powell, 2nd edition.
2. R Programming for Data Science by Roger D. Peng (References)
3. The Art of R Programming by Norman Matloff Cengage Learning India.

Course Outcomes:

On completion of the course students will be able to

1. Understand the fundamentals of data visualization and create basic visualizations using Tableau.
2. Connect and manipulate data sources in Tableau, including formatting, pivoting, and filtering data.
3. Apply calculations and custom fields to enhance visualizations and derive insights.
4. Design interactive dashboards and stories for effective data communication and storytelling.
5. Publish, share, and export visualizations using Tableau Online and explore advanced chart types.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	-	-	3	-	-	-	2	-	2
CO2	3	3	2	2	3	-	-	-	2	2	2
CO3	3	3	2	2	3	-	-	-	2	2	2
CO4	3	3	3	2	3	2	-	-	3	2	2
CO5	3	3	2	2	3	2	-	-	2	2	2

END