HYDERABAD

Indian in Character International in Excellence

SYLLABUS BOOK

Department of

INFORMATION TECHNOLOGY

Bachelor of Technology (B. Tech)

Coverage: Up to II **B. Tech,** II **Semester**

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

Medbowli, Meerpet, Hyderabad - 500097, Telangana, India

I Year I Semester

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25MA101BS	Matrices and Calculus	3	1	0	4
2.	25PH102BS	Advanced Engineering Physics	3	0	0	3
3.	25CS105ES	Programming for Problem Solving	3	0	0	3
4.	25EE104ES	Basic Electrical Engineering	3	0	0	3
5.	25ME105ES	Engineering Drawing and Computer Aided Drafting	2	0	2	3
6.	25PH106BS	Advanced Engineering Physics Lab	0	0	2	1
7.	25CS107ES	Programming for Problem Solving Lab	0	0	2	1
8.	25EE108ES	Basic Electrical Engineering Lab	0	0	2	1
9.	25CS109ES	IT Workshop	0	0	2	1
10.		Induction Program				
		Total Credits	**14**	**1**	**10**	**20**

I Year II Semester

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25MA201BS	Ordinary Differential Equations and Vector Calculus	3	0	0	3
2.	25CH202BS	Engineering Chemistry	3	0	0	3
3.	25CS205ES	Data Structures	3	0	0	3
4.	25EC204ES	Electronic Devices and Circuits	3	0	0	3
5.	25EN205HS	English for Skill Enhancement	3	0	0	3
6.	25CH206BS	Engineering Chemistry Lab	0	0	2	1
7.	25CS207ES	Data Structures Lab	0	0	2	1
8.	25EN208HS	English Language and Communication Skills Lab	0	0	2	1
9.	25ME209ES	Engineering Workshop	0	0	2	1
10.	25CS208ES	Python Programming Lab	0	0	2	1
		Total Credits	**15**	**0**	**10**	**20**

II YEAR I SEMESTER

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25MA301BS	Mathematical and Statistical Foundations	3	0	0	3
2.	25IT301PC	Computer Organization and Microprocessor	3	0	0	3
3.	25IT302PC	Java Programming	3	0	0	3
4.	25IT303PC	Operating Systems	3	0	0	3
5.	25IT304PC	Software Engineering	3	0	0	3
6.	25MA306PC	Computational Mathematics Lab	0	0	2	1
7.	25IT305PC	Java Programming Lab	0	0	2	1
8.	25IT306PC	Operating Systems Lab	0	0	2	1
9.	25IT307PC	Software Engineering Lab	0	0	2	1
10.	25IT308SD	Data Visualization-R Programming/Power BI/ Tableau/Google Chart	0	0	2	1
		Total Credits	**15**	**0**	**10**	**20**

II YEAR II SEMESTER

S.No.	Course Code	Course Title	L	T	P	Credits
1.	25CS401PC	Discrete Mathematics	3	0	0	3
2.	25IT401PC	Data Communications and Computer Networks	3	0	0	3
3.	25IT402PC	Formal Languages and Automata Theory	3	0	0	3
4.	25IT403PC	Database Management Systems	3	0	0	3
5.	25IT404PC	Web Programming	3	0	0	3
6.	25MS406HS	Innovation and Entrepreneurship	2	0	0	2
7.	25IT405PC	Computer Networks lab	0	0	2	1
8.	25IT406PC	Database Management Systems Lab	0	0	2	1
9.	25IT407PC	Web Programming Lab	0	0	2	1
10.	25IT408SD	Node JS/ React JS/ Django, UI Design - Flutter	0	0	2	1
11.	25VA400HS	Indian Knowledge System	1	0	0	1
		Total Credits	**18**	**0**	**08**	**22**

III YEAR I SEMESTER (27 Hours)

S. No.	Course Code	Course Title	L	T	P	Credits
1.	25IT501PC	Introduction to IoT	3	0	0	3
2.	25IT502PC	Machine Learning	3	0	0	3
3.	25IT503PC	Algorithm Design and Analysis	3	0	0	3
4.	25IT510PE	Professional Elective-I	3	0	0	3
5.	25IT510OE	Open Elective-I	2	0	0	2
6.	25IT504PC	Internet of Things Lab	0	0	2	1
7.	25IT505PC	Machine Learning Lab	0	0	2	1
8.	25IT506PC	Algorithm Design and Analysis Lab	0	0	2	1
9.	25IT507PC	Field Based Project/ Internship	0	0	4	2
10.	25IT508SD	Full Stack Development	0	0	2	1
11.	25VA500HS/ 25VA501HS	Gender Sensitization*/ Human Values and Professional Ethics*	1	0	0	0.5+0.5
		Total Credits	**15**	**0**	**12**	**21**

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

III YEAR II SEMESTER (25 Hours)

S. No	Course Code	Course Title	L	T	P	Credits
1.	25IT601PC	Compiler Design	3	0	0	3
2.	25IT602PC	Deep Learning	3	0	0	3
3.	25MS603HS	Business Economics and Financial Analysis	3	0	0	3
4.	25IT620PE	Professional Elective-II	3	0	0	3
5.	25IT620OE	Open Elective-II	2	0	0	2
6.	25IT604PC	Compiler Design Lab	0	0	2	1
7.	25IT605PC	Deep Learning Lab	0	0	2	1
8.	25IT606PC	Development Operations (DevOps) Lab	0	0	2	1
9.	25EN607HS	English for Employability Skills Lab	0	0	2	1
10.	25IT608SD	Big data-Spark	0	0	2	1
11.	25VA600ES	Environmental Science	1	0	0	1
		Total Credits	**15**	**0**	**10**	**20**

IV YEAR I SEMESTER (25 Hours)

S. No.	Course Code	Course Title	L	T	P	Credits
1.	25IT701PC	Information Security	3	0	0	3
2.	25IT702PC	Natural Language Processing	3	0	0	3
3.	25MS703HS	Fundamentals of Management	3	0	0	3
4.	25IT730PE	Professional Elective-III	3	0	0	3
5.	25IT740PE	Professional Elective-IV	3	0	0	3
6.	25IT730OE	Open Elective - III	2	0	0	2
7.	25IT704PC	Information Security Lab	0	0	2	1
8.	25IT705PC	Natural Language Processing Lab	0	0	2	1
9.	25IT706PC	Industry Oriented Mini Project/ Internship	0	0	4	2
		Total Credits	**17**	**0**	**8**	**21**

IV YEAR II SEMESTER (48 Hours)

S. No.	Course Code	Course Title	L	T	P	Credits
1.		Professional Elective - V	3	0	0	3
2.		Professional Elective - VI	3	0	0	3
3.	IT801PC	Project Work	0	0	42	14
		Total Credits	**6**	**0**	**42**	**20**

Professional Electives						
	PE1	**PE2**	**PE3**	**PE4**	**PE5**	**PE6**
1	Distributed systems	Cloud computing	Distributed database	Scalable architecture for Large Applications	High Performance Computing	Augmented Reality & Virtual Reality
2	Artificial Intelligence	Data analytics	Data mining	Data engineering with python	Reinforcement Learning	Generative AI
3	Number theory& Cryptography	Web security	Vulnerability assessments & Penetration Testing	Secure coding principles	Blockchain technology	Digital forensics
4	Optimization Techniques	Software Testing Methodology	Software Project Management	Distributed Systems	Information Retrieval Systems	Storage Area Networks
5	Mobile Application Development	Mobile Application Security	Mobile Computing	5G Technologies	Quantum Computing	Green Computing

OPEN ELECTIVES

Open Elective-I:

1	Java Programming
2	Mathematics for Machine Learning

Open Elective-II:

1	Full Stack development
2	Scripting Languages

Open Elective-III:

1	Big Data Technologies
2	Development and Operations (DevOps)

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

MATRICES AND CALCULUS								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA101BS	Basic Sciences	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Basic Principles of Algebra and Calculus								

I. COURSE OVERVIEW:

This course provides a comprehensive introduction to Matrices & Caluculus, covering fundamental concepts, techniques and applications in Mathematics, Physics, Engineering and other fields.

II. COURSE OBJECTIVES:

The students will try to learn:

I	Applying basic operations on matrices and their properties.
II	Concept of a rank of the matrix and applying this concept to know the consistency and solving the system of linear equations.
III	Concept of eigen values and eigen vectors and to reduce the quadratic form to canonical form
IV	Geometrical approach to the mean value theorems and their application to the mathematical problems
V	Finding maxima and minima of functions of two and three variables.
VI	Evaluation of multiple integrals and their applications.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Write the matrix representation of a set of linear equations and to analyse the solution of the system of equations
CO2:	Find the Eigen values and Eigen vectors and Reduce the quadratic form to canonical form using orthogonal transformations.
CO3:	Apply the mean value theorems and Studying curve tracing in Cartesian coordinates.
CO4:	Find the extreme values of functions of two variables with/ without constraints.
CO5:	Evaluate the multiple integrals and apply the concept to find areas, volumes

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12	
CO1	3	3		1								1	
CO2	2	2		1								1	
CO3	2	2										1	
CO4	2	2		1	2							1	
CO5	1	1										1	
	2.0	**2.0**		**1.0**	**2.0**							**1.0**	

V. COURSE CONTENT:

UNIT- I: (10 Hrs)

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: (10 Hrs)

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) – Fnding 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: (10 Hrs)

Single Variable Calculus

Limit and Continuous of 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).

Curve Tracing: Curve tracing in cartesian coordinates.

UNIT - IV: (10 Hrs)

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: (10 Hrs)

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.

VI. TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 45th Edition, 2024-25.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 10th Editon, 2020.

VII. REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9thEdition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
4. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ADVANCED ENGINEERING PHYSICS								
I Semester: EEE / ECE /IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25PH102BS	**Basic Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Principles of Algebra and Calculus								

I. COURSE OVERVIEW:

Engineering Physics is a multidisciplinary course blending physics principles with engineering applications to solve complex technological problems, covering core areas like Quantum Mechanics, Optics, Materials Science, Electromagnetism, and Semiconductors, preparing students for roles in research, innovation, and advanced tech fields like Nanotechnology and Photonics through a mix of theory, lab work, and practical projects.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To study crystal structures, defects, and material characterization techniques like XRD and SEM.

II. To understand fundamental concepts of quantum mechanics and their applications in solids and nanomaterials.

III. To introduce quantum computing principles, quantum gates, and basic quantum algorithms.

IV. To learn the properties and applications of magnetic and dielectric materials.

V. To explore the working and applications of lasers and fibre optics in modern technology.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze crystal structures, identify defects, and apply XRD and SEM techniques for material characterization.
CO2:	Apply quantum mechanical principles to explain particle behaviour and energy band formation in solids.
CO3:	Understand quantum computing concepts, use quantum gates, and explain basic quantum algorithms.
CO4:	Classify magnetic and dielectric materials and explain their properties, synthesis, and applications.
CO5:	Apply the principles of lasers and fibre optics and their applications in communication and sensing.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

UNIT- I: (12 Hrs)

Crystallography & Materials Characterization

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

UNIT – II: (12 Hrs)

Quantum Mechanics

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

UNIT - III: (12 Hrs)

Quantum Computing

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

UNIT - IV: (12 Hrs)

Magnetic and Dielectric Materials

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

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

UNIT - V: (12 Hrs)

Laser and Fibre Optics

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

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

VI. TEXT BOOKS:

1. Walter Borchardt-Ott, Crystallography: An Introduction, Springer.
2. Charles Kittel, Introduction to Solid State Physics, John Wiley & Sons, Inc.
3. Thomas G. Wong, Introduction to Classical and Quantum Computing, Rooted Grove

VII. REFERENCE BOOKS:

1. Jozef Gruska, *Quantum Computing*, McGraw Hill
2. Michael A. Nielsen & Isaac L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press.
3. John M. Senior, *Optical Fiber Communications Principles and Practice*, Pearson Education Limited.

VIII. ELECTRONICS RESOURCES:

1. https://shijuinpallotti.wordpress.com/wp-content/uploads/2019/07/optical-fiber-communications-principles-and-pr.pdf
2. https://www.geokniga.org/bookfiles/geokniga-crystallography_0.pdf
3. https://dpbck.ac.in/wp-content/uploads/2022/10/Introduction-to-Solid-State-PhysicsCharles-Kittel.pdf
4. https://www.thomaswong.net/introduction-to-classical-and-quantum-computing-1e4p.pdf
5. https://www.fi.muni.cz/usr/gruska/qbook1.pdf
6. https://profmcruz.wordpress.com/wp-content/uploads/2017/08/quantum-computation-and-quantum-information-nielsen-chuang.pdf

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

PROGRAMMING FOR PROBLEM SOLVING								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS105ES	Engineering Sciences	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Mathematics Fundamentals and Computer Knowledge								

I. COURSE OVERVIEW:

This course is Programming for Problem Solving is a foundational, general-purpose language developed in the early 1972s, known for its efficiency, portability, and influence on modern languages like C++, Java, and Python. The course assumes no prior programming experience and focuses on building core skills through hands-on coding, debugging, and problem-solving. By the end, learners can write, compile, and run simple to moderately complex programs

I. COURSE OBJECTIVES:

The students will try to learn:

I To learn the fundamentals of computers.

II To understand the various steps in program development.

III To learn the syntax and semantics of the C programming language.

IV To learn the usage of structured programming approaches in solving problems.

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	To write algorithms, to draw flowcharts for solving problems and to convert the algorithms/flowcharts to C programs.
CO2:	To code and test a given logic in the C programming language.
CO3:	To decompose a problem into functions and to develop modular reusable code.
CO4:	To use arrays, pointers, strings and structures to write C programs.
CO5:	Searching and sorting problems.

III. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3	2	2	1	1	1	1	1	1	1	1	3	1
CO2	3	2	2	1	1	1	1	1	1	1	1	3	2
CO3	3	3	3	2	1	1	2	1	1	1	1	3	2
CO4	3	2	2	1	1	1	1	1	1	1	1	3	2
CO5	3	2	2	1	1	1	1	1	1	1	1	2	2

IV. COURSE CONTENT:

UNIT- I: (10 Hrs)

Overview of C:C Language Elements, Variable Declarations and Data Types, Creating, Compiling and Executing a program, General Form of a C Program, Operators, Operator Precedence, Formatting Numbers in Program Output, Type Conversions.

Selection Structures: Control Structures, Conditions, if Statement, if Statements with Compound Statements, Decision Steps in Algorithms.

Repetition and Loop Statements: Repetition in Programs, Counting Loops and the while Statement, Computing a Sum or Product in a Loop, for Statement, Conditional Loops, Loop Design, Nested Loops, do-while Statement.

UNIT – II: (9 Hrs)

Arrays: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access, Using Array Elements as Function Arguments, Array Arguments, Searching and Sorting an Array, Multidimensional Arrays, Arrays of Pointers.

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

UNIT - III: (9 Hrs)

Top-Down Design with Functions: Building Programs from Existing Information, Library Functions, Top-Down Design and Structure Charts, Functions without Arguments, Functions with Input Arguments.

Pointers and Modular Programming: Pointers and the Indirection Operator, Functions with Output Parameters, Multiple Calls to a Function with Input/ Output Parameters, Scope of Names, Formal Output Parameters as Actual Arguments, Dynamic Memory Allocation.

UNIT - IV: (9 Hrs)

Recursion: The Nature of Recursion, Tracing a Recursive Function, Recursive Mathematical Functions, Recursive Functions with Array and String Parameters

Structure and Union Types: User-Defined Structure Types, Structure Type Data as Input and Output Parameters, Functions with Structured Result Values, Union Types, Enumerated Types.

UNIT - V: (8)

Text and Binary File Pointers: Input/ Output Files - Review and Further Study, Binary Files.

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

V. TEXT BOOKS:

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

VI. 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. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

VII. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/c-programming-language/
2. https://www.tutorialspoint.com/cprogramming/
3. https://www.programiz.com/c-programming
4. https://www.w3schools.com/c/c_intro.php

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

BASIC ELECTRICAL ENGINEERING								
I Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EE104ES	**Engineering Sciences**	3	0	0	3	40	60	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematics								

I. COURSE OVERVIEW:

This course Introduction to Electrical Engineer is a foundation course of electrical for all engineering branches. The concepts of DC & AC circuits, Power & Power factor, R-L-C elements, machine, transformer and electrical installations are very useful in real life. This course is applicable for industries.

II. COURSE OBJECTIVES:

The students will try to learn:

I To understand DC and Single & Three phase AC circuits

II To study and understand the different types of DC, AC machines and Transformers

III To import the knowledge of various electrical installations and the concept of power, power factor and its improvement

III. COURSE OUTCOMES:

At the end of the course, students should be able to:	
CO1:	Understand and analyze basic Electrical circuits
CO2:	Understand and analyze basic Electrical circuits Laws and Theorems
CO3:	Study the working principles of Transformers
CO4:	Study the working principles of Electrical Machines and Transformers
CO5:	Introduce components of Low Voltage Electrical Installations

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3			2	2	1						1	1
CO2	2	3	2		2	2				1		2	
CO3	1		3	1		3					2	3	1
CO4	1	2	3		1					1		2	
CO5	1	1	3	2	2	1						1	

V. COURSE CONTENT:

UNIT- I: (5 Hrs)

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: (7 Hrs)

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: (6 Hrs)

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

UNIT - IV: (7 Hrs)

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: (5 Hrs)

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**.**

VI.TEXT BOOKS:

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

VII.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 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

VIII.ELECTRONICS RESOURCE

1. https://nptel.ac.in/courses/108/104/108104139/
2. https://onlinecourses.nptel.ac.in/noc25_ee91/

TEEGALAKRISHNAREDDYENGINEERINGCOLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur (M), Hyderabad,Telangana-500097

COURSE CONTENT

25ME105ES ENGINEERING DRAWING AND COMPUTER AIDED DRAFTING								
ISemester:					**Regulations:R25**			
CourseCode	**Category**	**Hours/Week**			**Credits**	**MaximumMarks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25ME105ES	**Engineering Sciences**	2	0	2	3	40	60	100
Contact Classes:60	**Tutorial Classes:0**	**Practical Classes:60**				**Total Classes:60**		
Prerequisite: Imagination Skills Required								

I. COURSE OVERVIEW:

An Engineering drawing and Computer Aided Drafting (ED&CAD) course overview covers the fundamentals of graphic representation of designs, including manual sketching techniques and their transition to using CAD software for creating 2D and 3D models. Key Topics includes Introduction to Engineering Graphics, Orthographic Projections (Projections of points, Straight lines, Planes, Auxiliary planes, Projections of Regular Solids and sectional Views, Development of Surfaces, Isometric Projections of planes, solids, Conversions.

II. COURSE OBJECTIVES:

The students will try to learn:

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

III.COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Apply the principles of Engineering Drawing to create drawings of various geometric constructions, conic sections, curves and scales as per BIS standards by conventional mode.
CO2:	Construct orthographic projections for points, lines and planes in different quadrants and Auxiliary views of planes both in Conventional and Computer Aided Drafting
CO3:	Draw the sectional views and true shape of sections of solids, by applying principles of projections both in conventional and Computer Aided Drafting
CO4:	Draw the development of surfaces of solids in real time situations by conventional mode.
CO5:	Develop isometric and orthographic views of the objects both in conventional and Computer Aided Drafting.

IV. COURSE MARTICULATION MATRIX

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

V. COURSE CONTENT:

UNIT-I: **Introduction to Engineering Graphics (Conventional) (14Hrs)**

Principles of Engineering Graphics and their Significance, Geometrical Constructions,Scales,Plain and Diagonal, Conic Sections including the Rectangular Hyperbola, General method only. Cycloid, Epicycloid and Hypocycloid

UNIT-II: **Orthographic Projections (Conventional and Computer Aided) (12 Hrs)**

Principles of Ortho graphic Projections, Conventions, Projections of Points and Lines, Projections of Plane regular geometric figures. Auxiliary Planes. Computer aided orthographic projections, points, lines and planes. Introduction to Computer aided drafting, views, commands and conics.

UNIT III

Projections of Regular Solids (Conventional and Computer Aided) (14Hrs)

Auxiliary Views, Sections or Sectional views of Right Regular Solids, Prism, Cylinder, Pyramid, Cone, Auxiliary views, Computer aided projections of solids, sectional views

UNIT-IV:

Development of Surfaces (Conventional): **(10Hrs)**

Prism, Cylinder, Pyramid and Cone.

UNIT-V: **Isometric Projections (Conventional and Computer Aided) (10Hrs)**

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

VI. TEXT BOOKS:

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

VII. REFERENCE BOOKS:

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

VIII. ELECTRONIC RESOURCES:

1. https://soaneemrana.com/onewebmedia/ENGINEERING%20DRAWING%20BY%20N.D%20BHATT.pdf
2. https://ia800105.us.archive.org/0/items/TextbookOfEngineeringDrawing_201802/Textbook%20of%20Engineering%20Drawing.pdf

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ADVANCED ENGINEERING PHYSICS LAB								
I Semester: EEE / ECE /IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25PH106BS	**Basic Sciences**	-	-	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: Nill**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic lab handling skills in introductory courses								

I. COURSE OVERVIEW:

An Engineering Physics Lab course provides hands-on experience to bridge theory and practice, focusing on experiments in optics, electronics, mechanics, and modern physics (lasers, fibers) to develop practical skills, precise measurement techniques, data analysis, and technical report writing, enabling students to apply physics principles to real-world engineering problems and design solutions.

II. COURSE OBJECTIVES:

The students will try to learn:

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

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Synthesize and analyze nanomaterials such as magnetite (Fe_3O_4) using chemical methods.
CO2:	Determine key electrical, magnetic, and optical properties of semiconductors and other functional materials
CO3:	Characterize semiconductors using Hall effect and energy gap measurement techniques
CO4:	Demonstrate working knowledge of laser systems and optical fiber parameters through experimental study
CO5:	Apply scientific methods for accurate data collection, analysis, and technical report writing

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11
CO1	1			3		2			1		2
CO2			3		2	1			1		2
CO3	2		2		2	3				2	
CO4	2				2	3			1	2	
CO5		2	3		2				1		1

COURSE CONTENT:

List of Experiments:

1. Synthesis of magnetite (Fe_3O_4) powder using sol-gel method.
2. Determination of energy gap of a semiconductor.
3. Determination of Hall coefficient and carrier concertation of a given semiconductor.
4. Determination of magnetic field along the axis of a current carrying coil.
5. Study of B-H curve of a ferro magnetic material.
6. Determination of work function and Planck's constant using photoelectric effect.
7. Determination of dielectric constant of a given material.
8. Determination of Curie's temperature of a given ferroelectric material.
9. Determination of wavelength of a laser using diffraction grating.
10. Study of V-I & L-I characteristics of a given laser diode.
11. Determination of numerical aperture of a given optical fibre.
    Determination of bending losses of a given optical fibre.

Note: Any 8 experiments are to be performed.

I. REFERENCE BOOKS:

1. Practical physics by Vijay Kumar and T Radha Krishna (JNTUH).

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

PROGRAMMING FOR PROBLEM SOLVING LAB								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS105ES	Engineering Sciences	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic Mathematics Fundamentals and Computer Knowledge								

I. COURSE OVERVIEW:

Programming for Problem Solving Lab is intended to provide hands-on exposure to problem solving using the C programming language. This course enables students to understand the program development process and to work effectively with an Integrated Development Environment (IDE) for writing, compiling, debugging, and executing programs.

II. COURSE OBJECTIVES:

The students will try to learn:

I To work with an IDE to create, edit, compile, run and debug programs

II To analyze the various steps in program development.

III To develop programs to solve basic problems by understanding basic concepts in C like operators, control statements etc.

IV To develop modular, reusable and readable C Programs using the concepts like functions, arrays etc.

V To Write programs using the Dynamic Memory Allocation concept.

VI To create, read from and write to text and binary files

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Formulate algorithms for simple problem-solving tasks and Translate algorithms into correct and executable C programs.
CO2:	Identify and correct syntax and logical errors in programs.
CO3:	Use arrays, strings, structures, and pointers to represent and manipulate data.
CO4:	Apply modular programming concepts using functions, dynamic memory allocation, and file handling.
CO5:	Develop efficient solutions using sorting, searching, and standard problem-solving techniques.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	3	2	2	–	1	–	–	–	–	–	–	3	2
CO2	2	3	2	–	1	–	–	–	–	–	–	2	2
CO3	3	2	2	–	2	–	–	–	–	–	–	3	2
CO4	2	2	3	1	2	–	–	–	–	–	–	2	3
CO5	3	3	2	1	2	–	–	–	–	–	–	3	2

V. COURSE CONTENT:

PRACTICE SESSIONS:

Simple numeric problems:

a) Write a program for finding the max and min from the three numbers.
b) Write the program for the simple, compound interest.
c) Write a program that prints a multiplication table for a given number and the number of rows in the table. For example, for a number 5 and rows = 3, the output should be:
   5 x 1 = 5
   5 x 2 = 10
   5 x 3 = 15
d) Write a program that shows the binary equivalent of a given positive number between 0 to 255.

Expression Evaluation:

a) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement).
b) Write a program that finds if a given number is a prime number.
c) Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.
d) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

Arrays, Pointers and Functions:

a) Write a C program to find the minimum, maximum and average in an array of integers.
b) Write a C program that uses functions to perform the following:
   I. Addition of Two Matrices
   II. Multiplication of Two Matrices
c) Write a program for reading elements using a pointer into an array and display the values using the array.
d) Write a program for display values reverse order from an array using a pointer.

Files:

a) Write a C program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.
b) 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).

Strings:

a) Write a C program that uses functions to perform the following operations:
   I. To insert a sub-string into a given main string from a given position.
   II. To delete n Characters from a given position in a given string
b) Write a C program to determine if the given string is a palindrome or not (Spelled same in both directions with or without a meaning like madam, civic, noon, abcba, etc.)
c) Write a C program that displays the position of a character ch in the string S or – 1 if S doesn't contain ch.
d) Write a C program to count the lines, words and characters in a given text.

Sorting and Searching:

a) Write a C program that uses non-recursive function to search for a Key value in a given list of integers using linear search method.
b) Write a C program that uses non-recursive function to search for a Key value in a given sorted list of integers using binary search method.
c) Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.
d) Write a C program that sorts the given array of integers using selection sort in descending order
e) Write a C program that sorts the given array of integers using insertion sort in ascending order
f) Write a C program that sorts a given array of names.

VI. TEXT BOOKS:

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

VII. 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. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

VIII. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/c-programming-language/
2. https://www.tutorialspoint.com/cprogramming/
3. https://www.programiz.com/c-programming
4. https://www.w3schools.com/c/c_intro.php

EGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSECONTENT

BASIC ELECTRICAL ENGINEERING LAB								
II Semester: IT					**Regulations:** R25			
CourseCode	**Category**	**Hours/Week**			**Credits**	**MaximumMarks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EE108ES	**Engineering Sciences**	0	0	2	1	40	60	100
ContactClasses:45	**TutorialClasses:00**	**PracticalClasses:02Hrs/week**				**TotalClasses:20**		
Prerequisite:Basic Electrical Engineering								

I. COURSEOVERVIEW:

Focuses on hands-on verification of circuit laws and theorems (Ohm's, Kirchhoff's), resonance, series/parallel RLC circuits, Thevenin's/Norton's theorems, practical knowledge on the fundamentals of DC and AC circuits.

II. COURSEOBJECTIVES:

Thestudentswilltrytolearn:

To measure the electrical parameters for different types of DC and AC circuits using conventional and theorems approach.

To study the transient response of various R, L and C circuits using different excitations.

To determine the performance of different types of DC, AC machines and Transformers.

III. COURSEOUTCOMES:

Attheendofthecoursestudentsshouldbeableto:	
CO1:	Verify the basic Electrical circuits through different experiments.
CO2:	Evaluate the performance calculations of Electrical Machines and Transformers through various testing methods.
CO3:	Analyze the transient responses of R, L and C circuits for different input conditions.

IV. COURSECONTENT:

Thefollowingexperimentsarerequiredtobeconductedcompulsory:

List of experiments/demonstrations:

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.

In addition to the above eight experiments, at least any two of the experiments fromthe followinglist arerequired to be conducted

1. Verification of Superposition theorem.
2. Three Phase Transformer: Verification of Relationship between Voltages and Currents (Star-Delta, Delta-Delta, Delta-star, Star-Star)
3. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
4. Measurement of Active and Reactive Power in a balanced Three-phase circuit
5. No-Load Characteristics of a Three-phase Alternator

VI 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.

VII REFERENCEBOOKS:

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. AbhijitChakrabarthi, SudiptaDebnath, 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.

VIII.ELECTRONICSRESOURCES

1. https://nptel.ac.in/courses/108/104/108104139/
2. https://nptel.ac.in/courses/108/106/108106172/
3. https://ocw.mit.edu/search/ocwsearch.htm?q=laboratory

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSECONTENT

IT WORKSHOP								
IIISemester:IT					**Regulations: R25**			
CourseCode	**Category**	**Hours/Week**			**Credits**	**MaximumMarks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS109ES	**Engineering Sciences**	0	0	2	1	40	60	100
ContactClasses:00	**TutorialClasses:00**	**PracticalClasses:30**			**TotalClasses:30**			
Prerequisite:								

I. COURSEOVERVIEW:

This Laboratory course is designed to provide hands-on practical knowledge of essential information technology tools and computer systems. This course introduces students to computer hardware, operating systems, office productivity tools, internet applications, and basic programming concepts.

II. COURSEOBJECTIVES:

The IT Workshop for engineers is a training lab course spread over 60 hours. The modules include training on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel, PowerPoint and Publisher.

III. COURSEOUTCOMES:

At the end of the course students should be able to:	
CO1:	Perform Hardware trouble shooting .
CO2:	Understand Hardware components and inter dependencies .
CO3:	Safe guard computer systems from viruses/worms
CO4:	Document/Presentation preparation
CO5:	Perform calculations using spreadsheets

IV: COURSE CONTENT:

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 configuring das dual boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Internet &World wide web:

TASK 1

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 Words 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 Capin 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, Text boxes, 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 Scheduler-Features to be covered: Gridlines, Format Cells, Summation, autofill, 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

Power point:

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 –Back ground, textures, Design Templates, Hidden slides.

IV. TEXTBOOKS:

1.Comdex Information Technology course tool kit Vikas Gupta, *WILEY Dream tech*
2.The Complete Computer upgrade and repair book,3rd edition Chery lAS chmidt, *WILEY Dream tech*
3. Introduction to Information Technology, ITL Education Solutions limited, *Pearson Education.*

V. REFERENCEBOOKS:

1. PC Hardware-A Handbook–KateJ.Chase*PHI*(Microsoft)
2. LaTeX Companion–Leslie Lamport, *PHI/Pearson.*
3. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. *CISCO Press, Pearson Education*.
4.IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan —CISCO Press, *Pearson Education.*

VI. ELECTRONICSRESOURCES:

1. https://www.learningcomputer.com/tutorials/?utm_source=chatgpt.com.
2. https://www.youtube.com/upciss
3. https://process.spoken-tutorial.org/index.php/Software-Training?utm_source=chatgpt.com

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

25MA201BS: ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS								
I Semester: EEE / ECE / CSE / IT / CSE (AI & ML) / CSE (DS)					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA201BS	**Basic Sciences**	3	1	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Basic understanding of level Mathematics concepts								

I. COURSE OVERVIEW:

This course provides a comprehensive foundation in **Differential Equations and Vector Calculus**, bridging the gap between theoretical calculus and practical engineering applications. It moves from solving first-order and higher-order ODEs to mastering **Laplace Transforms**, a vital tool for analyzing linear systems and initial value problems. Finally, it explores **Vector Field Theory**, culminating in the major integration theorems (Green, Gauss, and Stokes) used to solve complex physical problems involving flux and circulation.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Methods of solving the differential equations of first and higher order.
II. Concept, properties of Laplace transforms
III. Solving ordinary differential equations using Laplace transforms techniques.
IV. The physical quantities involved in engineering field related to vector valued functions.
V. The basic properties of vector valued functions and their applications to line, surface and volume integrals

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Identify whether the given differential equation of first order is exact or not
CO2:	Apply the concept of differential equation to real world problems.
CO3:	Use the Laplace transforms techniques for solving ODE's
CO4:	Use gradient to evaluate directional derivatives and conservative vector field.
CO5:	Calculate the line, surface and volume integrals and converting them from one to another.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12	
CO1	2	3		1	1							1	
CO2	1	2			2							2	
CO3	3	2		2								1	
CO4	2	1		2								1	
CO5	2	2										1	
	2.0	2.0		1.7	1.5							1.2	

V. COURSE CONTENT:

UNIT- I: (12 Hrs)

First Order ODE

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: (12 Hrs)

Ordinary Differential Equations of Higher Order

Higher order linear differential equations with constant coefficients: Non-Homogeneousterms of the type e^{ax}, , $\sin ax$, $\cos ax$, polynomials in x, $e^{ax}V(x)$ and $xV(x)$, Method of variation of parameters.

UNIT - III: (12 Hrs)

Laplace transforms:

Laplace Transforms: Laplace Transform of standard functions, First shifting theorem, Secondshifting theorem, Unit step function, Dirac delta function, Laplace transforms of functions when they are multiplied and divided by 't', Laplace transforms of derivatives and integralsof 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: (12 Hrs)

Vector Differentiation

Vector point functions and scalar point functions, Gradient, Divergence and Curl, Directional derivatives, Tangent plane and normal line, Vector Identities, Scalar potential functions, Solenoidal and Irrotational vectors.

UNIT - V: (12 Hrs)

Vector Integration

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

TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 45th Edition, 2024-25.
2. 2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, NarosaPublications,10th Edition, 2020.

VI. REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, John Wiley & Sons, 2018.
2. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand andCompany Limited, New Delhi.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications .Reprint. 2018..

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

25CH102BS: ENGINEERING CHEMISTRY								
II Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CH202BS	**Basic Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic understanding of level chemistry concepts								

I. COURSE OVERVIEW:

The Engineering Chemistry course provides the foundational chemical principles and applications essential for all branches of engineering. It bridges the gap between pure chemistry and engineering practices by explaining how chemical processes, materials, and technologies influence modern engineering systems.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To develop adaptability to new advances in Engineering Chemistry and acquire the essential skills to become a competent engineering professional.
2. To understand the industrial significance of water treatment, fundamental principles of battery chemistry, and the impact of corrosion along with its control methods for structural protection.
3. To impart foundational knowledge of various energy sources and their practical applications in engineering.
4. To equip students with an understanding of smart materials, biosensors, and analytical techniques applicable in engineering, industrial, environmental, and biomedical fields.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Basic properties of water and its treatment methods for both domestic and industrial use.
CO2:	Basic understanding of electrochemical processes, including corrosion and its control mechanisms.
CO3:	Comprehend principles and applications of various energy sources, such as batteries and fuels, relevant to engineering.
CO4:	Basic concepts and properties and applications of different types of polymers and engineering materials.
CO5:	Apply the principles of spectroscopic techniques (UV-Vis, IR, Raman) for material analysis and pollution detection.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

UNIT- I: (12 Hrs)

Water and its treatment (14 Hrs)

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

UNIT – II: (12 Hrs)

Electrochemistry and Corrosin (12 Hrs)

Introduction- Electrode potential, standard electrode potential, Nernst equation (no derivation), electro-chemical 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 pH 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, water-line 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: (12 Hrs)

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- Dulongs **formula – Numerical problems.**

Fossil fuels: Introduction, Classification, Petroleum - Refining of Crude oil, Cracking - Types of cracking

Synthetic Fuels – Moving bed catalytic cracking. LPG and CNG composition and uses.

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

UNIT - IV: (12 Hrs)

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 thermo setting plastics, Fiber reinforced plastics (FRP).

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

Biodegradable polymers: Polylactic acid and its applications.

UNIT - V: (12 Hrs)

Advanced Functional Materials:

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

Biosensor - Definition, Amperometric Glucose monitor sensor. Interpretative spectroscopic applications of 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.

VI. TEXT BOOKS:

1. Engineering Chemistry by P.C. Jain and M. Jain, Dhanpatrai Publishing Company, 2010.
2. Engineering Chemistry by Rama Devi, Dr. P. Aparna and Rath, Cengage learning, 2025.

VII. REFERENCE BOOKS:

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

VIII. ELECTRONICS RESOURCES:

E-Content- https://doi.org/10.1142/13094 | October 2023
E-books: https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2u

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATA STRUCTURES								
II Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25CS205ES	**Engineering and Sciences**	L	T	P	C	CIE	SEE	Total
		3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

This course provides a comprehensive introduction to data structures and algorithms used in computer programming. It emphasizes the selection of appropriate data structures, analysis of their efficiency, and implementation of algorithms for searching, sorting, hashing, and file organization.

I. COURSE OBJECTIVES:

The students will try to learn:

I. Exploring basic data structures such as stacks and queues.
II. Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
III. Introduces sorting and pattern matching algorithms.

II. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Ability to select the data structures that efficiently model the information in a problem.
CO2:	Ability to assess efficiency trade-offs among different data structure implementations or combinations.
CO3:	Implement and know the application of algorithms for sorting and pattern matching.
CO4:	Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.
CO5:	Ability to select the data structures that efficiently model the information in a problem.

III. COURSE CONTENT:

UNIT- I:

Introduction to Data Structures: Basic Terminology, Classification of Data Structures, Operation on Data Structures, abstract data types, selecting a Data Structure, Linear list — Introduction, singly linked list, Circular Linked Lists, Doubly Linked List, Stacks- Operations, Stack algorithm, Stack ADT, Stack applications, Queues- operations, Queue Algorithm, Queue ADT, Queue Applications.

UNIT – II:

Trees: Introduction, Types of Trees, creating a Binary Tree from a General Tree, traversing a Binary Tree, Binary Search Trees (BST), BST Operations- Searching, Insertion and Deletion, BST ADT, BST Applications, Threaded Binary Trees, AVL Trees, Red –Black Trees, Splay Trees

UNIT - III:

Multi way Search Trees: Introduction, B Trees, B Trees ADT, 2-3 Trees, 2-3- Tree, B* Tree, B+ Trees
Heaps: Binary Heaps, Binomial heaps, Fibonacci heaps, Comparison of Various Heaps, Applications
Searching: Introduction, Interpolation Search, Jump search

UNIT - IV:

Graphs: Introduction, Directed Graphs, Bi connected Components, Representation of Graphs, Graph Traversal Algorithms, Graph ADT, Applications of Graphs
Sorting: Radix Sort, Heap sort, Shell Sort, Tree Sort,

UNIT – V

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
Files and their Organization: Introduction, Data hierarchy, File Attributes, Text and Binary Files, Basic

IV. TEXT BOOKS:

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.

V. REFERENCE BOOKS:

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

VI. ELECTRONICS RESOURCES:

1. NPTEL – Data Structures and Algorithms (Video Lectures & Course) https://nptel.ac.in/courses/106/102/064/
2. GeeksforGeeks – Data Structures & Algorithms Tutorials and Practice https://www.geeksforgeeks.org/data-structures-and-algorithms-online-courses-free-and-paid/
3. Coursera – Data Structures Course (University of California San Diego) https://www.coursera.org/learn/data-structures
4. W3Schools (Data Structures Tutorial Index) https://www.w3schools.in/data-structures/tutorials/
5. Google Tech Dev Guide – Data Structures Resources & Practice https://techdevguide.withgoogle.com/resources/topics/data-structures/
6. GitHub – DSA Resources (Community Compiled Lists) https://github.com/GDSC-KIIT/DSA-Resources

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ELECTRONIC DEVICES AND CIRCUITS								
II Semester: IT / CSE (AI & ML) / CSE (DS)					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EC204ES	Engineering Sciences	3	0	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes:**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite:								

I. COURSE OVERVIEW:

This course introduces fundamental semiconductor devices and their behavior, including diodes, BJTs, and FETs. It covers their characteristics, applications, and the analysis of basic electronic circuits. The course also explores rectifiers, voltage regulation, amplifier design, and advanced semiconductor technologies like FinFETs and CNTFETs. Emphasis is placed on developing a strong foundation for analog circuit design and understanding modern device technologies in electronics.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To understand the characteristics of semiconductor diodes and its applications.
2. To understand the operations and Characteristics of BJT configurations.
3. To analyze the Biasing techniques for BJT.
4. To analyze the transistor parameters in various configurations.
5. To acquire the knowledge of special purpose electronic devices and their application

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze the electrical characteristics and models of semiconductor diodes and apply them in rectifier and clipping circuits.
CO2:	Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.
CO3:	Evaluate the various biasing networks for BJTs and determine the operating point for amplifier
CO4:	Analyze transistor amplifier circuits using h-parameter models and assess performance for various configurations.
CO5:	Analyze the structure, working, and characteristics of JFETs, MOSFETs, and advanced devices like Fin FETs and CNTFETs, and acquire the knowledge of special purpose devices.

IV. COURSE CONTENT:

UNIT- I: (10 Hrs)

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: (14 Hrs)

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: (12 Hrs)

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: (14 Hrs)

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: (10)

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, Introduction of Advanced Devices: FinFETs and CNTFETs

V. TEXT BOOKS:

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.

VI. 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.

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ENGLISH FOR SKILL ENHANCEMENT								
II Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EN205HS	**Humanities & Social Sciences**	3	-	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Basic understanding of vocabulary, grammar, reading, and writing skills in English								

I. COURSE OVERVIEW:

English for Skill Enhancement focuses on skill development, fostering ideas and practice of language skills in various contexts and cultures in the areas of vocabulary, grammar, reading and writing.

II. COURSE OBJECTIVES:

This course will enable the students to:

I. Improve their vocabulary.
II. Use appropriate sentence structures in their oral and written communication.
III. Develop their reading and study skills.
IV. Equip students to write paragraphs, essays, précis and draft letters.
V. Acquire skills for Technical report writing

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Choose appropriate vocabulary in their oral and written communication.
CO2:	Demonstrate their understanding of the rules of functional grammar and sentence structures.
CO3:	Develop comprehension skills from known and unknown passages.
CO4:	Write paragraphs, essays, précis and draft letters.
CO5:	Write abstracts and reports in various contexts.

IV. COURSE CONTENT:

UNIT- I: (12 Hrs)

Theme:	Perspectives **Lesson on 'The Generation Gap' by Benjamin M. Spock from the prescribed textbook titled *English for the Young in the Digital World* published by Orient Black Swan Pvt. Ltd.**
Vocabulary:	The Concept of Word Formation-The Use of Prefixes and Suffixes-Words Often Misspelt - Synonyms and Antonyms
Grammar:	Identifying Common Errors in Writing with Reference to Parts of Speech particularly Articles and Prepositions – Degrees of Comparison
Reading:	Reading and Its Importance-Sub Skills of Reading–Skimming and Scanning.
Writing:	Sentence Structures and Types -Use of Phrases and Clauses in Sentences- Importance of Proper Punctuation-Techniques for Writing Precisely–Nature and Style of Formal Writing.

UNIT – II: (12 Hrs)

Theme:	Digital Transformation **Lesson on '*Emerging Technologies*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Homophones, Homonyms and Homographs
Grammar:	Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement.
Reading:	Reading Strategies-Guessing Meaning from Context–Identifying Main Ideas–Exercises for Practice
Writing:	Paragraph Writing – Types, Structures and Features of a Paragraph - Creating Coherence – Linkers and Connectives - Organizing Principles in a Paragraph – Defining- Describing People, Objects, Places and Events – Classifying- Providing Examples or Evidence - Essay Writing - Writing Introduction and Conclusion.

UNIT - III: (12 Hrs)

Theme:	Attitude and Gratitude **Poems on '*Leisure'* by William Henry Davies and '*Be Thankful'* -Unknown Author from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Words Often Confused-Words from Foreign Languages and their Use in English.
Grammar:	Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses.
Reading:	Sub-Skills of Reading – Identifying Topic Sentence and Providing Supporting Ideas – Exercises for Practice
Writing:	Format of a Formal Letter-Writing Formal Letters E.g.., Letter of Complaint, Letter of Requisition, Job Application with CV/Resume–Difference between Writing a Letter and an Email - Email Etiquette

UNIT - IV: (12 Hrs)

Theme:	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 BlackSwan 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: (12 Hrs)

Theme:	Integrity and Professionalism **Lesson on '*Professional Ethics*' from the prescribed textbook titled *English for the Young in the Digital World* published by Orient BlackSwan Pvt. Ltd.**
Vocabulary:	Technical Vocabulary and their Usage–One Word Substitutes–Collocations.
Grammar:	Direct and Indirect Speech-Common Errors in English (Covering all the other aspects of grammar which were not covered in the previous units)
Reading:	Survey, Question, Read, Recite and Review (SQ3R Method) – Inferring the Meaning and Evaluating a Text- Exercises for Practice
Writing:	Report Writing- Technical Reports-Introduction–Characteristics of a Report – Categories of Reports Formats- Structure of Reports (Manuscript Format)-Types of Reports-Writing a Technical Report.

Note: ***Listening and Speaking skills which are given under Unit-6 in AICTE Model Curriculum are covered in the syllabus of ELCS Lab Course.***

- (**Note**: As the syllabus of English given in AICTE *Model Curriculum-2018 for B.Tech. First Year is* Open-ended, besides following the prescribed textbook; it is required to prepare teaching/learning materials by the teachers collectively in the form of handouts based on the needs of the students in their respective colleges for effective teaching/learning in the class.)

V. TEXT BOOKS:

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

VI. REFERENCE BOOKS:

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

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ENGINEERING CHEMISTRY LAB								
II Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CH206BS	**Basic Sciences**	-	-	2	1	40	60	100
Contact Classes: 0	**Tutorial Classes: 0**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic lab handling skills in introductory chemistry courses								

I. COURSE OVERVIEW:

Engineering Chemistry Lab is designed to give students hands-on experimental experience to reinforce and complement the theoretical concepts learned in this course. It helps bridge the gap between theory and practice, by enabling students to perform chemical analyses relevant to engineering applications.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Students will understand and perform experiments based on core chemical principles relevant to engineering applications.
2. Students will learn to estimate the hardness of water to assess its suitability for drinking purposes.
3. Students will acquire the ability to perform acid-base titrations using instrumental methods such as conductometry, potentiometry, and pH metry.
4. Students will gain hands-on experience in synthesizing polymers like Bakelite and Nylon 6, 6 in the laboratory.
5. Students will learn to determine the unknown concentration of potassium permanganate ($KMnO_4$) using a calibration curve.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Perform volumetric and instrumental analysis to estimate the concentration of chemical species in solutions.
CO2:	To determine important parameters such as water hardness and the corrosion rate of mild steel under various conditions.
CO3:	Applying techniques like conductometry, potentiometry, and pH metry to determine concentrations or equivalence points in acid-base reactions.
CO4:	To synthesizing polymers like Bakelite and Nylon – 6, 6 in the laboratory.
CO5:	To understand the working principle of colorimetry and the relationship between absorbance and concentration (Beer-Lambert Law).

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	PSO1	PSO2
CO1	2	3		1	2	-	-	-	-	-	1		
CO2	2				2	2	-	-	2	3			
CO3	2	2	-		3	-	-	-	-	2	1		
CO4	1	2	-		1	-	1	-	2	3			
CO5	2		2				1	-	-	2	3		

V. COURSE CONTENT:

List of Experiments:

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.

II. 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

III. pH Metry: Determination of an acid concentration using pH meter.

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

V. Preparations:

1. Preparation of Bakelite.
2. Preparation Nylon – 6, 6.

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

VII. Virtual lab experiments:

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

VI. REFERENCE BOOKS:

1. Lab manual for Engineering chemistry by B. Ramadevi and P. Aparna, S Chand Publications, New Delhi (2022).
2. Vogel's text book of practical organic chemistry 5th edition.

COURSE CONTENT

DATA STRUCTURES LAB								
I Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
25CS207ES	**Engineering and Science**	L	T	P	C	CIE	SEE	Total
		-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic Mathematics Fundamentals and Computer Knowledge								

I. COURSE OVERVIEW:

The **Data Structures Lab** provides hands-on experience in implementing fundamental and advanced data structures using the C programming language. It reinforces theoretical concepts learned in the *Data Structures* theory course by enabling students to:

- Develop programs using arrays, pointers, functions, and control structures.
- Implement linear and non-linear data structures such as stacks, queues, linked lists, trees, heaps, graphs, and hash tables.
- Apply sorting and searching algorithms efficiently to solve real-world computational problems.
- Gain practical skills in modular program design, debugging, and algorithm optimization.

This lab is designed to strengthen logical thinking, programming proficiency, and problem-solving skills, bridging the gap between theory and practical applications of data structures in computer science.

II. COURSE OBJECTIVES:

The students will try to learn:

I It covers various concepts of C programming language
II It introduces searching and sorting algorithms
III It provides an understanding of data structures such as stacks and queues.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop C programs using arrays, functions, pointers, and control statements for solving computational problems.
CO2:	Implement and manipulate linear data structures such as stacks, queues, and linked lists efficiently.
CO3:	Construct and traverse non-linear data structures including binary trees, AVL trees, B-trees, and graphs.
CO4:	Apply sorting and searching algorithms such as radix sort, heap sort, shell sort, interpolation search, and binary search in practical scenarios.
CO5:	Implement and analyze hash functions and collision resolution techniques to store and retrieve data effectively.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

List of Experiments

1. Write a program that uses functions to perform the following operations on singly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implement stack (its operations) using
   i) Arrays ii) ADT
5. Write a program that implement Queue (its operations) using
   i) Arrays ii) ADT
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
   i) Radix Sort, ii) Heap sort, iii) Shell Sort, iv) Tree Sort
7. Write a program to implement the tree traversal methods (Recursive and Non-Recursive).
8. Write a program to implement
   i) Binary Search tree ii) B Trees iii) B+ Trees iv) AVL trees v) Red - Black trees
9. Write a program to implement the graph traversal methods.
10. Write a program to implement the following Hash Functions: i) Division Method, ii) Multiplication Method, iii) Mid-square Method, iv) Folding Method.

V. TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

VI. REFERENCE BOOKS:

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

VII. ELECTRONICS RESOURCES:

1. NPTEL – Data Structures and Algorithms (Video Lectures & Course) https://nptel.ac.in/courses/106/102/064/
2. GeeksforGeeks – Data Structures & Algorithms Tutorials and Practice https://www.geeksforgeeks.org/data-structures-and-algorithms-online-courses-free-and-paid/
3. Coursera – Data Structures Course (University of California San Diego) https://www.coursera.org/learn/data-structures
4. edX – Learn Data Structures & Algorithms Courses https://www.edx.org/learn/data-structures
5. W3Schools (Data Structures Tutorial Index) https://www.w3schools.in/data-structures/tutorials/
6. Google Tech Dev Guide – Data Structures Resources & Practice https://techdevguide.withgoogle.com/resources/topics/data-structures/
7. GitHub – DSA Resources (Community Compiled Lists) https://github.com/GDSC-KIIT/DSA-Resources

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

ENGLISH LANGUAGE AND COMMUNICATION SKILLS (ELCS) LAB								
II Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25EN208HS	**Humanities & Social Sciences**	-	-	2	1	40	60	100
Contact Classes: 0	**Tutorial Classes: 0**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Basic listening and speaking skills in English								

I. COURSE OVERVIEW:

The **English Language and Communication Skills (ELCS) Lab** focuses on listening and speaking skills, particularly on the production and practice of sounds of language and familiarizes the students with the use of English in everyday situations both in formal and informal contexts.

II. COURSE OBJECTIVES:

To enable students, develop their active listening skills:

I	To enable students, develop their active listening skills
II	To equip students with necessary training in listening, so that they can comprehend the speech of people from different linguistic backgrounds
III	To improve their pronunciation and neutralize accent
IV	To enable students express themselves fluently and appropriately
V	To practice speaking in social and professional contexts

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Listen actively and identify important information in spoken texts
CO2:	Interpret the speech and infer the intention of the speaker
CO3:	Improve their accent for intelligibility
CO4:	Speak fluently with clarity and confidence
CO5:	Use the language in real life situations

IV. COURSE CONTENT:

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

a. Computer Assisted Language Learning (CALL) Lab, which focuses on listening skills

b. Interactive Communication Skills (ICS) Lab, which focuses on speaking skills

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

Exercise– I
CALL Lab:
Instruction: Speech Sounds-Listening Skill-Importance–Purpose-Types-Barriers-Active Listening
Practice: Listening to Distinguish Speech Sounds (Minimal Pairs)-Testing Exercises
ICS Lab:
❖ **Diagnostic Test – Activity titled '*Express Your View*'**
Instruction: Spoken and Written language - Formal and Informal English - Greetings - Introducing Oneself and Others
Practice: Any Ice-Breaking Activity

Exercise–II
CALL Lab:
Instruction: Listening vs. Hearing- Barriers to Listening
Practice: Listening for General Information - Multiple Choice Questions - Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)
ICS Lab:
Instruction: Features of Good Conversation–Strategies for Effective Communication
Practice: Role Play Activity - Situational Dialogues –Expressions used in Various Situations –Making Requests and Seeking Permissions – Taking Leave - Telephone Etiquette

Exercise-III
CALL Lab:
Instruction: Errors in Pronunciation–Tips for Neutralizing Mother Tongue Influence (MTI)
Practice: Differences between British and American Pronunciation –Listening Comprehension Exercises
ICS Lab:
Instruction: 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

Minimum Requirement of Infrastructural Facilities for ELCS Lab

1. Computer Assisted Language Learning (CALL) Lab

The Computer Assisted Language Learning Lab has to accommodate 40 students with 40 systems, with one master console, LAN facility, and English language learning software for self-study by students.

System Requirements (Hardware Components):

- Computer network with LAN facility (minimum 40 systems with multimedia) with the following specifications:
  i) Computers with suitable configuration
  ii) High-fidelity headphones

2. Interactive Communication Skills (ICS) Lab

The Interactive Communication Skills Lab should be a spacious room with movable chairs and audio-visual aids, a public address system, a TV or LCD, a digital stereo audio and video system, and a camcorder.

Note: <u>English language teachers are requested to prepare materials and handouts for each activity for the use of those materials in CALL and ICS Labs.</u>

Suggested Software

- Cambridge Advanced Learner's English Dictionary with CD
- Grammar Made Easy by Dorling Kindersley
- Punctuation Made Easy by Dorling Kindersley
- Oxford Advanced Learner's Compass, 10th Edition
- English in Mind (Series 1–4) by Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

25ME109ES ENGINEERING WORKSHOP								
ISemester:					**Regulations: R25**			
CourseCode	**Category**	**Hours/Week**			**Credits**	**MaximumMarks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25ME209ES	**Engineering Sciences**	0	0	2	1	40	60	100
Contact Classes: 30	**Tutorial Classes: Nill**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite:NIL								

I. COURSE OVERVIEW:

An Engineering workshop lab provides essential hands-on experience that bridges the gap between theoretical knowledge and practical applications. The course, typically for first-year students, familiarizes them with basic trades (Carpentry, Fitting, Tin smithy, Welding, Foundry, House Wiring, Black Smithy) ,materials, tools, Sequence of operations and safety practices and also Demonstration and exposure on Plumbing, Machine Shop, Metal Cutting , Power tools in construction and Wood Working.

II. COURSE OBJECTIVES:

The students will try to learn:

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

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Demonstrate various machines, tools and their operations.
CO2:	Apply different workshop trades like fitting, carpentry, foundry and welding
CO3:	Practice workshop trades like Tim smithy, Black smithy
CO4:	Identify suitable tools for different trades of engineering processes including drilling, material removing, measuring, chiseling
CO5:	Apply basic electrical engineering knowledge for house wiring practice.

IV. COURSE ARTICULATION MATRIX

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

COURSE CONTENT:

1. TRADES FOR EXERCISES: At least two exercises from each trade:

i. **Carpentry:** T-Lap Joint, Dovetail Joint, Mortise and Tenon Joint (4 Hrs)
ii. **Fitting :** V-Fit, Dovetail Fit and Semi-circular fit (4 Hrs)
iii. **Tin Smithy**: Square Tin, Rectangular Tray and Conical Funnel (4 Hrs)
iv. **Foundry**: Preparation of Greensand Mould using Single Piece and Split Pattern (4 Hrs)
v. **Welding Practice:** Arc Welding and Gas Welding (4 Hrs)
vi. **House wiring**: Parallel and Series ,Two-way Switch and Tube Light (2 Hrs)
vii. **Black Smithy**: Round to Square, Fan Hook and S-Hook (4 Hrs)

2. TRADES FOR DEMONSTRATION AND EXPOSURE: (4 Hrs)

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

VI TEXT BOOKS:

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

VII REFERENCE BOOKS:

1. Workshop Manual, K.Venu gopal, Anuradha Publications, 2012th edition, 2012.

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

PYTHON PROGRAMMING LAB								
I Semester: C S E						**Regulations: R25**		
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIE	SEE	Total
25CS208ES	**Engineering and Science**	-	-	2	1	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: A course on Programming for Problem Solving								

I. COURSE OVERVIEW:

The **Python Programming Lab** aims to provide hands-on experience in programming using Python. The lab emphasizes practical implementation of programming constructs, control structures, data types, and advanced features like file handling, object-oriented programming, and GUI design. Students will develop problem-solving skills and gain experience in creating applications using Python libraries and modules, including digital logic simulations, data processing, and visualization.

II. COURSE OBJECTIVES:

The students will try to learn:

I To install and run the Python interpreter
II To learn control structures.
III To Understand Lists, Dictionaries in python
IV To Handle Strings and Files in Python

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop the application specific codes using python.
CO2:	Understand Strings, Lists, Tuples and Dictionaries in Python
CO3:	Verify programs using modular approach, file I/O, Python standard library
CO4:	Implement Digital Systems using Python
CO5:	Design GUI-based applications using Python libraries such as Tkinter.

IV. COURSE CONTENT:

PRACTICE SESSIONS:

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.

2. Start a Python interpreter and use it as a Calculator.
3. Write a program to calculate compound interest when principal, rate and number of periods are given.
4. Read the name, address, email and phone number of a person through the keyboard and print the details.
5. Print the below triangle using for loop. 5

   4 4
   3 3 3
   2 2 2 2
   1 1 1 1 1
6. 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)
7. Python program to print all prime numbers in a given interval (use break)
8. Write a program to convert a list and tuple into arrays.
9. Write a program to find common values between two arrays.
10. 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.
11. 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.
12. 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.
13. 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.
14. The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add "I", "a", and the empty string.
15. 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.
16. Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
17. Remove the given word in all the places in a string?
18. 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?
19. Writes a recursive function that generates all binary strings of n-bit length
20. Write a python program that defines a matrix and prints
21. Write a python program to perform multiplication of two square matrices
22. 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.
23. Use the structure of exception handling all general-purpose exceptions.
24. 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.
25. Add an attribute named color to your Rectangle objects and modify draw_rectangle so that it uses the color attribute as the fill color.
26. 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.
27. 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.
28. Write a python code to read a phone number and email-id from the user and validate it for correctness.
29. Write a Python code to merge two given file contents into a third file.
30. 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.
31. Write a Python code to Read text from a text file, find the word with most number of occurrences
32. 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.
33. Import numpy, Plotpy and Scipy and explore their functionalities.
34. Install NumPypackage with pip and explore it.
35. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
36. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as

Submit and Reset.

V. TEXT BOOKS:

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

VI. 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

VII. ELECTRONICS RESOURCES:

1. Python Official Website: https://www.python.org
2. NPTEL Python Programming: https://nptel.ac.in/courses/106/106/106106144/
3. GeeksforGeeks Python Tutorials: https://www.geeksforgeeks.org/python-programming-language/
4. W3Schools Python Tutorial: https://www.w3schools.com/python/
5. TutorialsPoint Python Tutorial: https://www.tutorialspoint.com/python/index.htm

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

25MA301BS: Mathematical and Statistical Foundations								
III Semester: CSD, IT & CSM					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA301BS	Basic Sciences	3	1	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Mathematics courses of first year of study								

I. COURSE OVERVIEW:

This course provides core quantitative skills, covering number theory like GCD/primes. Focus on statistical techniques and methods using computers.

II. COURSE OBJECTIVES:

The students will try to learn:

1. The Number Theory basic concepts useful for cryptography etc
2. The theory of Probability, and probability distributions of single and multiple randomvariables
3. The sampling theory and testing of hypothesis and making inferences
4. The Curve fitting, Correlation and Regression for the given data.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Apply the number theory concepts to cryptography domain
CO2:	Apply the concepts of probability and distributions to some case studies
CO3:	Correlate the material of one unit to the material in other units
CO4:	Resolve the potential misconceptions and hazards in each topic of study.
CO5:	Fit the Curve , Correlation and Regression for the given data.

IV. COURSE ARTICULATION MATRIX

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12	
CO1	2	2		2								1	
CO2	2	3			1							1	
CO3	2	2			2							1	
CO4	1	2			1							1	
CO5	2	2		1	1							1	
	1.8	2.2		1.5	1.3							1.0	

V. COURSE CONTENT:

UNIT- I: (8 Hrs)

Greatest Common Divisors and Prime Factorization: Greatest common divisors, The Euclidean algorithm, The fundamental theorem of arithmetic, Factorization of integers and the Fermat numbers. Congruences: Introduction to congruences, linear congruences, The Chinese remainder theorem, Systems of linear congruences

UNIT – II: (10 Hrs)

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 - III: (10 Hrs)

Continuous Probability Distributions: Uniform distribution, Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial Distribution.
Fundamental Sampling Distributions: Random Sampling, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem.

UNIT - IV: (10 Hrs)

Tests of Hypotheses (Large and Small samples): Statistical Hypotheses: General Concepts, testing a Statistical Hypothesis, Single sample: Testsconcerning a single mean, two samples: tests on two means. Test on single proportion, Tests on two proportions, F - Distribution.

UNIT - V: (10 Hrs)

Applied Staistics : Curve fitting by the method of least squares, Fitting of straight lines, Second degree parabolas and more general curves, Correlation and Regression, Rank correlation,

VI. TEXT BOOKS:

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & StatisticsFor Engineers & Scientists, 9th Ed. Pearson Publishers.
2. Introduction to Analytic Number Theory, Tom M Apostal, Narosa publishing house.
3. Kenneth H Rosen, Elementary Number Theory & its Applications, Sixth edition Addison Wesley.
4. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.

VII. REFERENCE BOOKS:

1. T.T. Soong, Fundamentals of Probability and Statistics For Engineers, John Wiley & Sons, Ltd, 2004
2. Sheldon M Ross, Probability and statistics for Engineers and scientists, academic press.
3. Miller and Freund's, Probability and Statistics for Engineers, 9th Edition, Pearson Educations,2020
4. Elementary Number Theory , James K Strayer

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

COMPUTER ORGANIZATION AND MICROPROCESSOR								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT301PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Digital Logic Design								

I. COURSE OVERVIEW:

This course Computer Organization and Microprocessor is a foundation concepts of digital computer organization, instruction execution, control unit design, and arithmetic operations. It also introduces the architecture and programming of the 8086 microprocessors, including instruction sets, addressing modes, assembly language programming, interrupts, and input–output organization.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To understand basic components of computers.
2. To understand the architecture of the 8086 processor.
3. To understand the instruction sets, instruction formats and various addressing modes of 8086.
4. To understand the representation of data at the machine level and how computations are performed at machine level.
5. To understand the memory organization and I/O organization.
6. To understand the parallelism both in terms of single and multiple processors.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Able to understand the basic components and the design of CPU, ALU and Control Unit.
CO2:	Ability to understand memory hierarchy and its impact on computer cost/performance.
CO3:	Ability to understand the advantage of instruction level parallelism and pipelining for high performance Processor design.
CO4:	Ability to understand the instruction set, instruction formats and addressing modes of 8086.
CO5:	Ability to write assembly language programs to solve problems.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

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

Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt, Complete Computer Description.

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

UNIT – II: (14 Hrs)

Central Processing Unit: The 8086 Processor Architecture, register organization, Physical memory organization, General Bus Operation, I/O Addressing Capability, Special Processor Activities, Minimum and Maximum mode system and timings.

8086 Instruction Set and Assembler Directives-Machine language instruction formats, Addressing modes, Instruction set of 8086, Assembler directives and operators.

UNIT - III: (10 Hrs)

Assembly Language Programming with 8086- Machine level programs, Machine coding the programs, Programming with an assembler, Assembly Language example programs. Stack structure of 8086, Interrupts and Interrupt service routines, Interrupt cycle of 8086, Interrupt programming, Passing parameters to procedures, Macros, Timings and Delays.

UNIT - IV: (12 Hrs)

Computer Arithmetic: Introduction, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating - point Arithmetic operations. Input-Output Organization: Peripheral Devices, Input- Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access, Input –Output Processor (IOP), Intel 8089 IOP.

UNIT - V: (10 Hrs)

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

V: TEXT BOOKS:

1. Computer System Architecture, M. Morris Mano, Third Edition, Pearson. (UNITS – I, IV, V)
2. Advanced Microprocessors and Peripherals, K M Bhurchandi, A. K Ray ,3rd edition, McGraw Hill India Education Private Ltd. (UNITS - II, III).

VI: REFERENCE BOOKS:

1. Microprocessors and Interfacing, D V Hall, SSSP Rao, 3 rd edition, McGraw Hill India Education Private Ltd.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill.
3. Computer Organization and Architecture, William Stallings, 9th Edition, Pearson.
4. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier.

ELECTRONICS RESOURCES:

1. https://www.tutorialspoint.com/computer_organization/index.htm
2. https://www.coursera.org/learn/introduction-to-computer-organization
3. https://usemynotes.com/microprocessor/
4. https://www.tutorialspoint.com/computer_organization/pipelining.htm

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

JAVA PROGRAMMING								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT302PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Programming for problem solving								

I. COURSE OVERVIEW:

This course Java Programming is a foundation for advanced studies in software engineering, web technologies, mobile application development, and enterprise systems, and is highly applicable in the software industry and real-world problem solving.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To introduce the concept of object-oriented programming and its advantages over procedural programming.
2. To understand the concepts of classes and objects and their role in program design
3. To understand the need and usage of interfaces and their differences from abstract classes.
4. To understand the concept of multithreading and its importance in Java applications.
5. To understand the Java Collections Framework and its components.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Identify the model of Object-Oriented Programming: Abstract data types, Encapsulation, Inheritance and Polymorphism.
CO2:	Summarize the fundamental features like Interfaces, Exceptions and Collections.
CO3:	Correlate the advantages of multi-threading.
CO4:	Design interactive programs using Applets, AWT and Swings.
CO5:	Develop real time applications using the features of Java.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

OBJECT ORIENTED THINKING

Introduction, Need of object-oriented programming, principles of object-oriented languages, Applications of OOP, history of JAVA, Java Virtual Machine, Java features, Program structures, Installation of JDK.

Variables, Primitive data types, Identifiers- Naming Conventions, Keywords, Literals, Operators- Binary, Unary and Ternary, Expressions, Primitive Type conversion and casting, flow of control- branching, conditional, loops.

UNIT – II: (14 Hrs)

CLASSES, INHERITANCE, POLYMORPHISM

Classes and Objects- Classes, Objects, creating objects, methods, constructors- constructor overloading, cleaning up unused objects- Garbage collector, class variable and methods- static keyword, this keyword, arrays, Command line arguments, Nested Classes.

Strings: String, String Buffer, String Tokenizer.

Inheritance and Polymorphism- Types of Inheritance, deriving classes using extends keyword, super keyword, Polymorphism – Method Overloading, Method Overriding, final keyword, abstract classes.

UNIT - III: (10 Hrs)

INTERFACES, PACKAGES, EXCEPTIONS

Interfaces: Interface, Extending Interface, Interface Vs Abstract classes.

Packages- Creating Packages, using Packages, Access protection, java I/O package.

Files and I/O Streams: The File class, Streams, The Byte Streams, Filtered Byte Streams, The Random-Access File class.

UNIT - IV: (12 Hrs)

MULTITHREADING, EXCEPTIONS

java.lang.Thread, the main Thread, creation of new Threads, Thread priority, multithreading- using isalive() and join(), Synchronization, suspending and resuming Threads, Communication between Threads. Exploring java.io, Exploring java.util.

Exceptions - Introduction, Exception handling techniques-try…catch, throw, throws, finally block, user defined Exception.

UNIT - V: (10)

COLLECTIONS, APPLETS AND SWINGS

Collections: Overview of Collection Framework: ArrayList, Vector, TreeSet, HashMap, HashTable, Iterator, Comparator.

Applets: Applet class, Applet structure, an example of Applet program, Applet life cycle.

Swing: Introduction, JFrame, JApplet, JPanel, Components in Swings, JList and JScroll Pane, Split Pane, JTabbed Pane, Dialog Box, Pluggable Look and feel.

IV. TEXT BOOKS:

1. Java: The Complete Reference, 10th edition, Herbert Schildt, Mcgraw Hill.
2. Java Fundamentals: A Comprehensive Introduction, Herbert Schildt and Dale Skrien, TMH.
3. Java for Programming, P.J. Dietel Pearson Education.

V. REFERENCE BOOKS:

1. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Programming in Java, S. Malhotra and S. Choudhary, Oxford University Press.

VI. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/
2. https://www.javatpoint.com/java-tutorial
3. https://www.tutorialspoint.com/java/index.htm
4. https://www.javatpoint.com/java-objects-and-classes
5. https://www.tutorialspoint.com/java/java_files_io.htm
6. https://www.javatpoint.com/exception-handling-in-java
7. https://www.geeksforgeeks.org/collections-in-java-2/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

OPERATING SYSTEMS								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT303PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Programming of problem solving and Data Structures.								

I. COURSE OVERVIEW:

This course provides a comprehensive understanding of Operating System concepts, design, and functionality, focusing on how an OS manages hardware resources and provides services to users and application programs. The course covers process management, CPU scheduling, synchronization, memory management, virtual memory, deadlocks, interprocess communication, and file systems, with emphasis on both theoretical foundations and practical system calls used in modern operating systems such as UNIX/Linux.

II. COURSE OBJECTIVES:

The students will try to learn:

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

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Will be able to control access to a computer and the files that may be shared
CO2:	Demonstrate the knowledge of the components of computers and their respective roles in computing.
CO3:	Ability to recognize and resolve user problems with standard operating environments.
CO4:	Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.
CO5:	Analyze different file allocation methods and free-space management techniques used by operating systems.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

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: (14 Hrs)

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: (12 Hrs)

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: (10 Hrs)

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

UNIT - V: (10)

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.

IV. TEXT BOOKS:

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

V. 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.

VI. ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/operating-systems/
2. https://ocw.mit.edu/courses/6-004-computation-structures/
3. https://nptel.ac.in/courses/106/106/106106144/
4. https://www.youtube.com/@GateSmashers
5. https://man7.org/linux/man-pages/man2/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

SOFTWARE ENGINEERING								
III Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT304PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Programming for problem Solving and Data Structures								

I. COURSE OVERVIEW:

This course provides a systematic and disciplined approach to software development, focusing on processes, methodologies, tools, and techniques used to develop high-quality, reliable, and maintainable software systems. The course emphasizes both theoretical foundations and practical modeling techniques using UML, testing strategies, metrics, and standards.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.

II. Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Ability to translate end-user requirements into system and software requirements, using e.g. Introduce the issues to be considered in the design and development of operating system.
CO2:	UML, and structure the requirements in a Software Requirements Document (SRD).
CO3:	Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
CO4:	Will have experience and/or awareness of testing problems and will be able to develop a simple testing report.
CO5:	Analyze software risk management and quality management principles by identifying, assessing, and mitigating software risks.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

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 Development.

UNIT – II: (14 Hrs)

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: (12 Hrs)

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, use case diagrams, class diagrams, sequence diagrams, collaboration diagrams, activity diagrams and component diagrams.

UNIT - IV: (10 Hrs)

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. Network Management Challenges in IoT, Integration of SDN with IoT Networks.

UNIT - V: (10)

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.

IV. TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition. Vijay Madisetti, Arshdeep Bahga, "Internet of Things: A Hands-On Approach".
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

V. 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 Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

VI. ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/106/106106182/
2. https://www.geeksforgeeks.org/software-requirements-specification-srs/
3. https://www.visual-paradigm.com/tutorials/
4. https://www.istqb.org/
5. https://www.tutorialspoint.com/software_engineering/software_quality_management.htm

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

COMPUTATIONAL MATHEMATICS LAB (Using Python/MATLAB software)								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MA306PC	Professional Core	0	0	2	1	40	60	100
Contact Classes: 30	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 30**		
Prerequisite: Matrices, Iterative methods and ordinary differential equations								

I COURSE OBJECTIVES:

The students will try to learn:

1. Solve problems of Eigen values and Eigen Vectors using Python/MATLAB.
2. Solution of Algebraic and Transcendental Equations using Python/MATLAB
3. Solve problems of Linear system of equations
4. Solve problems of First-Order ODEs Higher order linear differential equations with constant coefficients

II COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Develop the code to find the Eigen values and Eigen Vectors using Python/MATLAB
CO2:	Develop the code to find solution of Algebraic and Transcendental Equations using Python/MATLAB
CO3:	Develop the code to find solution of Linear system of equations using Python/MATLAB
CO4:	Write the code to solve problems of First-Order ODEs.
CO5:	Write the code to solve problems of Higher order linear differential equations with constant coefficients

III COURSE CONTENT:

UNIT- I:

Eigen values and Eigenvectors:

Programs:

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

UNIT – II:

Solution of Algebraic and Transcendental Equations

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:
Jacobi's iteration method and Gauss-Seidal iteration method
Programs:
Solution of given system of linear equations using Jacobi's method

UNIT - IV:

First-Order ODEs
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
Programs:
Solving homogeneous ODEs
Solving non-homogeneous ODEs

IV TEXT BOOKS:

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.

V REFERENCE BOOKS:

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

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

JAVA PROGRAMMING LAB								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT305PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving								

I. COURSE OVERVIEW:

This laboratory course is designed to provide hands-on experience in Java programming, reinforcing object-oriented concepts through practical implementation. The course enables students to design, implement, and test Java programs covering OOP principles, strings, exception handling, multithreading, file handling, collections framework, GUI programming, event handling, and applet-based applications.

II. COURSE OBJECTIVES:

1. Understand and apply fundamental object-oriented programming concepts such as classes, objects, inheritance, polymorphism, encapsulation, and abstraction using Java.
2. Develop Java programs using constructors, method and constructor overloading, overriding, dynamic binding, keywords, and access specifiers.
3. Implement string processing, file handling, and exception handling techniques to build robust Java applications.
4. Design and implement multithreaded programs including inter-thread communication and synchronization concepts.
5. Use Java Collections Framework and apply appropriate data structures for efficient data storage and manipulation.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Analyze a problem, identify and define the computing requirements appropriate to its solution using object-oriented programming concepts.
CO2:	Design the applications using Inheritance, Polymorphism and Synchronization concepts.
CO3:	Handle exceptions at Compile time and Run time.
CO4:	Solve the real-world problems using Java Collection framework.
CO5:	Develop GUI applications using Applets, AWT and Swings.

IV: COURSE CONTENT:

TASK 1

Write java programs that implement the following

a. Class and object
b. Constructor
c. Parameterized constructor
d. Method overloading
e. Constructor overloading.

TASK 2

a) Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
b) Write a Java program for sorting a given list of names in ascending order.
c) Write a Java Program that reads a line of integers, and then displays each integer and the sum of all the integers (Use StringTokenizer class of java.util)

TASK 3

Write java programs that uses the following keywords
a) this b) super c) static d) final

TASK 4

a) Write a java program to implement method overriding
b) Write a java program to implement dynamic method dispatch.
c) Write a Java program to implement multiple inheritance.
d) Write a java program that uses access specifiers.

TASK 5

a) Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
b) Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
c) Write a Java program that displays the number of characters, lines and words in a text file

TASK 6

a) Write a Java program for handling Checked Exceptions.
b) Write a Java program for handling Unchecked Exceptions.

TASK 7

a) Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.
b) Write a Java program that correctly implements producer consumer problem using the concept of interthread communication.

TASK 8

Write a program illustrating following collections framework
a) ArrayList b) Vector c) Hash Table d) Stack

TASK 9

a) Develop an applet that displays a simple message.
b) Develop an applet that receives an integer in one text field and compute its factorial value and return it in another text field, when the button named "Compute" is clicked.

TASK 10

a) Write a Java program for handling mouse events.
b) Write a Java program for handling key events.

TASK 11

a) Write a java program that simulates traffic light. The program lets the user select one of three lights: red, yellow or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.
b) Write a Java program that allows the user to draw lines, rectangles and ovals.
c) Create a table in Table.txt file such that 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. Write a java program to display the table using JTable component.

IV. TEXT BOOKS:

1. Java: The Complete Reference, 10th edition, Herbert Schildt, Mcgraw Hill.
2. Java Fundamentals - A Comprehensive introduction, Herbert Schildt and Dale skrien, TMH.
3. Java for programming, P.J. Dietel Pearson education (OR) Java: How to Program P.J. Dietel and H.M. Dietel, PHI.

V. REFERENCE BOOKS:

1. Object Oriented Programming through java, P. Radha Krishna, Universities Press.
2. Thinking in Java, Bruce Eckel, Pearson Education
3. Programming in Java, S. Malhotra and S. Choudhary, Oxford University Press.

VI. ELECTRONICS RESOURCES:

1. https://docs.oracle.com/javase/tutorial/java/concepts/
2. https://docs.oracle.com/javase/8/docs/api/java/util/StringTokenizer.html
3. https://www.geeksforgeeks.org/method-overriding-in-java/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

OPERATING SYSTEMS LAB								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT306PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving & Computer Organization and Architecture.								

I. COURSE OVERVIEW:

The Operating Systems Laboratory is designed to provide practical exposure to core operating system concepts through hands-on programming using C and UNIX/Linux system calls. The course reinforces theoretical concepts such as CPU scheduling, process management, deadlocks, synchronization, interprocess communication (IPC), memory management, and page replacement algorithms by implementing and simulating them in a real programming environment.

II. COURSE OBJECTIVES:

The students will try to learn:

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

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
CO2:	Able to implement C programs using Unix system calls.
CO3:	Implement deadlock avoidance and synchronization mechanisms using semaphores.
CO4:	Develop programs using interprocess communication mechanisms such as pipes, FIFOs, message queues, and shared memory.
CO5:	Simulate memory management and page replacement algorithms and analyze their behavior.

IV: COURSE CONTENT:

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

IV. TEXT BOOKS:

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

V. 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.

VI. ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/106/106106144/
2. https://www.geeksforgeeks.org/cpu-scheduling-in-operating-systems/
3. https://www.geeksforgeeks.org/bankers-algorithm-in-operating-system/
4. https://man7.org/linux/man-pages/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

SOFTWARE ENGINEERING LAB								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT307PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving & Computer Organization and Architecture.								

I. COURSE OVERVIEW:

The Software Engineering Laboratory aims to provide hands-on experience in applying software engineering principles to real-world projects. Students work in teams or individually to develop software project documentation and artifacts by following a systematic software development life cycle (SDLC) approach.

II. COURSE OBJECTIVES:

The students will try to learn:

To have hands-on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Ability to translate end-user requirements into system and software requirements.
CO2:	Ability to generate a high-level design of the system from the software requirements.
CO3:	Develop software configuration management and risk management documents for a software project.
CO4:	Use CASE tools to model and design software systems using UML diagrams.
CO5:	Design and develop test cases for unit testing, integration testing, and apply white-box and black-box testing techniques.

IV: COURSE CONTENT:

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 cases for various white box and black box testing techniques.

Sample Project:

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

IV: TEXT BOOKS:

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

V: REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley. Operating System - A Design Approach-Crowley, TMH.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-HilL.

IV. ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/106/106106182/
2. https://www.tutorialspoint.com/software_engineering/
3. https://www.geeksforgeeks.org/software-engineering/
4. https://staruml.io/
5. https://www.visual-paradigm.com/
6. http://argouml.tigris.org/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATA VISUALIZATION - R PROGRAMMING/ POWER BI/TABLEAU/GOOGLE CHART								
III Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT308SD	**Skill Development**	0	0	2	1	40	60	100
Contact Classes: 00	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving.								

I. COURSE OVERVIEW:

This course introduces students to the principles, tools, and techniques of data visualization with a strong practical focus on Power BI, along with exposure to Tableau and Google Charts. The course enables learners to transform raw data into meaningful visual insights through charts, dashboards, and interactive stories.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Effective use of Business Intelligence (BI) technology (Tableau) to apply data visualization
2. To discern patterns and relationships in the data.
3. To build Dashboard applications.
4. To communicate the results clearly and concisely.
5. To be able to work with different formats of data sets.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand how to import data into Tableau.
CO2:	Understand concepts of Dimensions and Measures.
CO3:	Develop Programs and understand how to map Visual Layouts and Graphical Properties
CO4:	Create Dashboard, custom charts, and, publish to tableau online for any real time dataset
CO5:	Design and develop interactive dashboards and stories with appropriate use of colors, tooltips, filters, and interactivity.

IV: COURSE CONTENT:

List of Experiments:

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, creating basic charts (line, bar charts, Tree maps), Using the Show me panel.
3. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.
4. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.

5. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data
6. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.
7. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.
8. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.
9. Tableau file types, publishing to Tableau Online, sharing your visualizations, printing, and exporting.
10. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.
11. Visualize various data patterns taking any dataset from Kaggle.
12. Visualize data patterns using Google Charts by creating interactive Line, Bar, and Pie charts along with Dashboards using HTML and JavaScript.

IV. TEXT BOOKS:

1. Microsoft Power BI – *Dashboards Step by Step,* Errin O'Connor, Microsoft Press.
2. Storytelling with Data: A Data Visualization Guide for Business Professionals, Cole Nussbaumer Knaflic, Wiley.

V. REFERENCE BOOKS:

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

VI. ELECTRONICS RESOURCES:

1. https://learn.microsoft.com/en-us/power-bi/
2. https://learn.microsoft.com/en-us/power-bi/fundamentals/
3. https://www.youtube.com/@MicrosoftPowerBI
4. https://www.geeksforgeeks.org/dashboard-design-best-practices/
5. https://datasetsearch.research.google.com/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DISCRETE MATHEMATICS								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25CS401PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Mathematic- II and Programming for Problem Solving.								

I. COURSE OVERVIEW:

This course is a fundamental subject for computer science and engineering students. It provides the mathematical foundations required for reasoning about algorithms, data structures, computer logic, software development, and networking. This course focuses on logic, set theory, algebraic structures, combinatorics, and graph theory, which are essential for understanding problem-solving, formal proofs, and discrete structures in computing.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Introduces elementary discrete mathematics for computer science and engineering.
2. Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand and construct precise mathematical proofs.
CO2:	Apply logic and set theory to formulate precise statements.
CO3:	Analyze and solve counting problems on finite and discrete structures.
CO4:	Describe and manipulate sequences.
CO5:	Apply graph theory in solving computing problems.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

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: (14 Hrs)

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

UNIT - III: (10 Hrs)

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

UNIT - IV: (12 Hrs)

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: (10 Hrs)

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.

V: TEXT BOOKS:

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.

VI: REFERENCE BOOKS:

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph. P. Grimald, Pearson education, 5th edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill Publishing co.

ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/111/106/111106103/
2. https://www.geeksforgeeks.org/propositional-logic-and-predicate-logic/
3. https://www.tutorialspoint.com/discrete_mathematics/set_theory.htm

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATA COMMUNICATIONS AND COMPUTER NETWORKS								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT401PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Operating Systems and Digital Logic.								

I. COURSE OVERVIEW:

This course provides a comprehensive understanding of data communications and computer networking with a layered approach. Students will learn to design, analyze, and troubleshoot networks by understanding both theoretical and practical concepts.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Explore basic concepts of data communication
2. Introduce students to TCP/IP and OSI models along with their merits and demerits.
3. Explore in detail services offered by various layers of OSI Model.
4. Understand link layer protocols, UDP, TCP and application layer protocols.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand and explore the basics of communication and computer networks
CO2:	Understand datalink layers concepts of a computer network.
CO3:	Understand network layers concepts of a computer network
CO4:	Understand transport layers concepts of a computer network
CO5:	Understand the working of application layer protocols.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Data Communications: Components — Direction of Data flow — Networks — Components and Categories — Types of Connections — Topologies –Protocols and Standards — ISO / OSI model, **Physical layer:** Transmission modes, Multiplexing, Transmission Media, Switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT – II: (14 Hrs)

Data link layer: Introduction, Framing, and Error – Detection and Correction – Parity – LRC – CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols. 111 Medium Access sub layer: ALOHA, CSMA/CD, LAN – Ethernet IEEE 802.3, IEEE 802.5 – IEEE 802.11, Random access, Controlled access, Channelization.

UNIT - III: (10 Hrs)

Network layer: Logical Addressing, Internetworking, Tunneling, Address mapping, ICMP, IGMP, Forwarding, Uni-Cast Routing Protocols, Multicast Routing Protocols.

UNIT - IV: (12 Hrs)

Transport Layer: Process to Process Delivery, UDP and TCP protocols, Data Traffic, Congestion, Congestion Control, QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

UNIT - V: (10 Hrs)

Application Layer: Domain name space, DNS in Internet, Electronic Mail, SMTP, FTP, WWW, HTTP, SNMP.

V: TEXT BOOKS:

1. Data Communications and Networking, Behrouz A. Forouzan, Fourth Edition TMH.

VI: REFERENCE BOOKS:

1. Computer Networks, Andrew S Tanenbaum, 6th Edition. Pearson Education.
2. Computer Networking: A Top-Down Approach Featuring the Internet. James F. Kurose & Keith W. Ross, 3 rd Edition, Pearson Education
3. Data communications and Computer Networks, P.C Gupta, PHI.
4. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.

ELECTRONICS RESOURCES:

1. https://nptel.ac.in/courses/106/105/106105094/
2. https://www.geeksforgeeks.org/computer-network-tutorials/
3. https://www.tutorialspoint.com/data_communication_computer_network/index.htm
4. https://inl.info.ucl.ac.be/publications/cnp3e/

COURSE CONTENT

FORMAL LANGUAGES AND AUTOMATA THEORY								
IV Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT402PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Mathematical Foundations								

I. COURSE OVERVIEW:

The Formal Languages and Automata Theory Course explore the mathematical and theoretical foundations of computer science, focusing on formal languages, automata, and computational models. Students learn to model and analyze computation using abstract machines and formal grammars.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.
2. To introduce the fundamental concepts of formal languages, grammars and automata theory.
3. Classify machines by their power to recognize languages.
4. Employ finite state machines to solve problems in computing.
5. To understand deterministic and non-deterministic machines.
6. To understand the differences between decidability and undecidability.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the concept of abstract machines and their power to recognize the languages.
CO2:	Employ finite state machines for modeling and solving computing problems.
CO3:	Design context free grammars for formal languages.
CO4:	Distinguish between decidability and undecidability.
CO5:	Analyze types of Turing machines, study recursive and recursively enumerable languages, and identify undecidable problems including Post Correspondence Problem and halting problem.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.

Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with Є-transitions to NFA without Є-transitions. Conversion of NFA to DFA, Moore and Melay machines

UNIT – II: (14 Hrs)

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.
Pumping Lemma for Regular Languages, Statement of the pumping lemma, Applications of the Pumping Lemma.
Closure Properties of Regular Languages: Closure properties of Regular languages, Decision Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT - III: (10 Hrs)

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Tress, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.
Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state, Acceptance by empty stack, Deterministic Pushdown Automata. From CFG to PDA, From PDA to CFG.

UNIT - IV: (12 Hrs)

Normal Forms for Context- Free Grammars: Eliminating useless symbols, Eliminating €-Productions. Chomsky Normal form Greibach Normal form.
Pumping Lemma for Context-Free Languages: Statement of pumping lemma, Applications **Closure Properties of Context-Free Languages:** Closure properties of CFL's, Decision Properties of CFL's Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine.

UNIT - V: (10 Hrs)

Types of Turing machine: Turing machines and halting
Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem that is RE, Undecidable Problems about Turing Machines, Recursive languages, Properties of recursive languages, Post's Correspondence Problem, Modified Post Correspondence problem, Other Undecidable Problems, Counter machines.

V:TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Theory of Computer Science — Automata languages and computation, Mishra and Chandrasekaran, 2nd edition, PHI.

VI: REFERENCE BOOKS:

1. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.

ELECTRONICS RESOURCES

1. https://nptel.ac.in/courses/106/103/106103118/
2. https://www.geeksforgeeks.org/regular-expression/
3. http://www.jflap.org/
4. https://www.tutorialspoint.com/automata_theory/turing_machine.htm
5. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2011/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATABASE MANAGEMENT SYSYTEMS								
IV Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT403PC	**Professional Core**	3	1	-	4	40	60	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		
Prerequisite: Data Structures								

I. COURSE OVERVIEW:

The Database Management Systems (DBMS) course introduces students to data modeling, database design, relational databases, SQL, transactions, and indexing techniques. Students learn how to design, implement, and maintain databases effectively, focusing on data integrity, concurrency, and performance.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To understand the basic concepts and the applications of database systems.
2. To master the basics of SQL and construct queries using SQL.
3. Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Gain knowledge of fundamentals of DBMS.
CO2:	Apply relational model concepts, enforce integrity constraints, use relational algebra and calculus, and design logical database schemas with views.
CO3:	Master the basics of SQL for retrieval and management of data.
CO4:	Be acquainted with the basics of transaction processing and concurrency control.
CO5:	Familiarity with database storage structures and access techniques

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

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: (14 Hrs)

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: (10 Hrs)

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: (12 Hrs)

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: (10 Hrs)

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

V:TEXT BOOKS:

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

VI: 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.

ELECTRONICS RESOURCES

1. https://nptel.ac.in/courses/106/105/106105064/
2. https://www.geeksforgeeks.org/relational-algebra/
3. https://www.tutorialspoint.com/sql/index.htm
4. https://www.w3schools.in/dbms/indexing/

COURSE CONTENT

WEB PROGRAMMING								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT404PC	**Professional Core**	3	0	-	3	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Programming for problem solving and Java Programming.								

I. COURSE OVERVIEW:

The Web Programming course introduces students to front-end and back-end web development, covering HTML, CSS, JavaScript, PHP, XML, JDBC, Servlets, JSP, and Content Management Systems (CMS).

II. COURSE OBJECTIVES:

1. Learn web design using HTML, CSS, and JavaScript.
2. Understand PHP for server-side scripting and databases.
3. Explore XML structure and parsing techniques.
4. Study servlets, applets, and user interface components.
5. Use WordPress and Joomla for CMS development.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Build responsive web pages using HTML and CSS.
CO2:	Write PHP scripts for form handling and storage.
CO3:	Validate and parse XML with DTD and schemas.
CO4:	Implement servlets and applets with session management.
CO5:	Develop websites using CMS tools like WordPress.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

HTML- Basic Tags- List, Tables, Images, Forms, Frames, **Cascading Style Sheets (CSS)** – syntax, selectors, properties. **JAVA Script -** Web page Designing using HTML, Scripting basics- Client side and server-side scripting. Java Script Object, names, literals, operators and expressions- statements and features- events - windows - documents - frames - data types - built-in functions- Browser object model - Verifying forms- HTML5- CSS3- HTML 5 canvas - Web site creation using tools.

UNIT – II: (14 Hrs)

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies File Handling in **PHP**: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

UNIT - III: (10 Hrs)

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemes, Document Object Model, XHTML Parsing XML Data – DOM and SAX Parsers in java. **JDBC:** JDBC Overview – JDBC implementation – Connection class – Statements - Catching Database.

UNIT - IV: (12 Hrs)

Servlet – life cycle of a servlet. The Servlet API, Handling HTTP Request and Response, using Cookies, Session Tracking. Introduction to JSP. The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, Implicit objects, Using Bans in JSP Pages, Using Cookies and Session for session tracking, Connecting to database in JSP.

UNIT - V: (10 Hrs)

Database Handling and Content Management System

PHP Database Interface - Web Hosting - Content Management System - Case Study of CMS-using WordPress, Joomla - The content structure – Templates - Menu links – Components - Modules -Text editors — Permissions.

V:TEXT BOOKS:

1. Harvey Deitel, Abbey Deitel, Internet and World Wide Web: How To Program 5th Edition.
2. JavaScript: The Definitive Guide by David Flanagan, O'Reilly Media, Latest Edition.
3. Web Technologies: HTML, JavaScript, PHP, Java, JSP, ASP.NET, XML and Ajax by Uttam K. Roy, Oxford University Press
4. Michael Morrison XML Unleashed Tech media SAMS.
5. Web Technologies, Uttam K Roy, Oxford University Press

VI: REFERENCE BOOKS:

1. John Pollock, Javascript - A Beginners Guide, 3rd Edition –- Tata McGraw-Hill Edition.
2. Keyur Shah, Gateway to Java Programmer Sun Certification, Tata McGraw Hill, 2002.

ELECTRONICS RESOURCES

1. https://www.w3schools.com/html/
2. https://www.php.net/manual/en/book.filesystem.php
3. https://www.tutorialspoint.com/xml/xml_parsers.htm
4. https://docs.oracle.com/javaee/7/tutorial/servlets.htm
5. https://docs.joomla.org/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

INNOVATION AND ENTREPRENEURSHIP								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25MS406HS	**Humanities & Social Science**	2	0	-	2	40	60	100
Contact Classes: 30	**Tutorial Classes: 00**	**Practical Classes: Nil**				**Total Classes: 30**		
Prerequisite: Programming for problem solving								

I. COURSE OVERVIEW:

The Innovation and Entrepreneurship course introduces students to the fundamentals of entrepreneurship, innovation processes, and startup development. It emphasizes problem identification, opportunity assessment, prototype development, business planning, and intellectual property.

II. COURSE OBJECTIVES:

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

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand the entrepreneurship and the entrepreneurial process and its significance in economic development.
CO2:	Assess the problem from an industry perspective and generate solutions using the design thinking principles.
CO3:	Assess market competition, estimate market size, and develop a prototype.
CO4:	Analyze Business and financial planning models and Go-to-Market strategies.
CO5:	Able to build a start-up, register IP and identify funding opportunities.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Fundamentals of Innovation and Entrepreneurship

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

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

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

UNIT – II: (14 Hrs)

Intr Problem and Customer Identification

Identification of gap, problem, analyzing the problem from a industry perspective, real-world problems, market and customer segmentation, validation of customer problem fit, Iterating problem-customer fit, Competition and Industry trends mapping and assessing initial opportunity, Porter's Five Force Model. Idea generation, Ideation techniques: Brainstorming, Brain writing, Round robin, and SCAMPER, Design thinking principles, Mapping of solution to problem.

Core Teaching Tool: Several types of activities including: Class, game, Gen AI, 'Get out of the Building' and Venture Activity.

UNIT - III: (10 Hrs)

Opportunity assessment and Prototype development

Identify and map global competitors, review industry trends, and understand market sizing: TAM, SAM, and SOM. Assessing scope and potential scale for the opportunity.

Understanding prototyping and Minimum Viable Product (MVP). Developing a prototype: Testing, and validation.

Core Teaching Tool: Venture Activity, no-code Innovation tools, Class activity

UNIT - IV: (12 Hrs)

Business & Financial Models

Introduction to Business Model and types, Lean Canvas Approach: 9-block lean canvas model, building lean canvas for your startup. Business planning: components of Business plan- Sales plan, People plan and financial plan, Financial Planning: Types of costs, preparing a financial plan for profitability using a financial template, understanding the basics of Unit economics, Economies of Scale and analyzing financial performance. Go-To-Market (GTM) approach — Selecting the Right Channel, creating digital presence, and building customer acquisition strategy.

Core Teaching Tool: Founder Case Studies – Sama and Securely Share; Class activity and discussions; Venture Activities.

UNIT - V: (10 Hrs)

Startups and IPR

Startup requirements, building founding team members and mentors, pitch preparation, start-up registration process, funding opportunities and schemes, institutional support to entrepreneurs, startup lifecycle, documentation, legal aspects in startup, venture pitching readiness, National Innovation Startup Policy (NISP) and its features.

Patents, Designs, Patentability, Procedure for grants of patents. Indian Scenario of Patenting, International Scenario: International cooperation on Intellectual Property. Patent Rights: Scope of Patent Rights. Copyright, trademark, and GI. Licensing and transfer of technology.

Core Teaching Tool: Expert talks; Cases; Class activity and discussions; Venture Activities.

V:TEXT BOOKS:

1. John R Bessant, Joe Tidd, Innovation and Entrepreneurship, 4E, Wiley, Latest Edition.
2. Ajay Batra, The Stratup Launch Book- A Practical Guide for Launching Customer Centric Ventures, Wiley, 2020. (For Core Teaching Tool).
3. Entrepreneurship Development and Small Business Enterprises, Poornima M Charantimath, 3E, Pearson, 2018.

VI: REFERENCE BOOKS:

1. D.F. Kuratko and T.V. Rao, Entrepreneurship: A South-Asian Perspective, Cengage Learning, 2013.
2. Robert D. Hisrich, Michael P. Peters, Dean A. Shepherd, Sabyasachi Sinha (2020). Entrepreneurship, McGrawHill, 11th Edition.

ELECTRONICS RESOURCES

1. NISP -Brochure inside pages - startup_policy_2019.pdf
2. https://nptel.ac.in/courses/110/104/110104083/
3. https://www.ideo.com/post/design-kit
4. https://theleanstartup.com/principles
5. https://blog.hubspot.com/sales/go-to-market-strategy
6. https://www.startupindia.gov.in/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

COMPUTER NETWORKS LAB								
IV Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT405PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Operating Systems and Digital Logic.								

I. COURSE OVERVIEW:

The Computer Networks Lab is designed to provide hands-on experience with fundamental concepts of computer networks through practical implementation and simulation. The lab enables students to understand the working principles of communication protocols, analyze traffic flow, and study network performance using real tools and simulators.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To understand the working principle of various communication protocols.
2. To understand the network simulator environment and visualize a network topology and observe its performance.
3. To analyze the traffic flow and the contents of protocol frames.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Implement data link layer farming methods.
CO2:	Analyze error detection and error correction codes.
CO3:	Implement and analyze routing and congestion issues in network design.
CO4:	Implement Encoding and Decoding techniques used in presentation layer.
CO5:	To be able to work with different network tools.

IV: COURSE CONTENT:

List of Experiment

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.

V: TEXT BOOKS:

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

VI: 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.

ELECTRONICS RESOURCES:

1. https://www.geeksforgeeks.org/computer-network-tutorials/
2. https://www.tutorialspoint.com/computer_networks/index.htm
3. https://nmap.org/book/man.html
4. https://www.isi.edu/nsnam/ns/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

DATABASE MANAGEMENT SYSYTEMS LAB								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT406PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Data Structures								

I. COURSE OVERVIEW:

The Database Management Systems Lab provides hands-on practical exposure to the concepts of database design and implementation. This laboratory course complements the DBMS theory course by enabling students to apply conceptual, logical, and physical database design principles through structured experiments.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Design conceptual and logical database schemas using ER modeling and relational modeling techniques for real-world applications.
CO2:	Apply normalization techniques to database schemas to eliminate redundancy and ensure data integrity.
CO3:	Acquire skills in using SQL commands for data definition and data manipulation.
CO4:	Write advanced SQL queries using joins, subqueries, aggregate functions, views, and constraints.
CO5:	Develop solutions for database applications using procedures, cursors and triggers

IV: COURSE CONTENT:

list of Experiments:

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A) Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.) B) Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

V:TEXT BOOKS:

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

VI: 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.

ELECTRONICS RESOURCES

1. https://nptel.ac.in/courses/106/105/106105064/
2. https://www.geeksforgeeks.org/relational-algebra/
3. https://www.tutorialspoint.com/sql/index.htm
4. https://www.w3schools.in/dbms/indexing/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

WEB PROGRAMMING LAB								
IV Semester: IT					**Regulations: R25**			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT407PC	**Professional Core**	0	0	2	1	40	60	100
Contact Classes: 45	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Programming for problem solving and Java Programming.								

I. COURSE OVERVIEW:

The Web Technologies Lab provides practical exposure to the design and development of dynamic and interactive web applications using modern web technologies. This laboratory course complements the Web Technologies theory course by enabling students to apply concepts of client-side and server-side scripting, database connectivity, and web application deployment.

II. COURSE OBJECTIVES:

1. Learn HTML, CSS, JavaScript for building responsive web interfaces.
2. Understand PHP for handling forms, sessions, files, and databases.
3. Explore XML structure, validation, and parsing using different techniques.
4. Gain skills in database access using JDBC and SQL queries.
5. Implement CMS platforms like WordPress and Joomla for dynamic websites.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Design responsive websites using HTML, CSS, and JavaScript effectively.
CO2:	Develop PHP applications for data input, validation, and storage.
CO3:	Manipulate and parse XML documents in web-based applications.
CO4:	Connect web applications to databases using PHP or JDBC.
CO5:	Create and manage CMS-based websites using WordPress and Joomla.

IV: COURSE CONTENT:

list of Experiments:

1. HTML

A. Create a webpage with HTML describing your department. Use paragraph and list tags.
B. Apply various colors to suitably distinguish keywords. Also apply font styling like italics, underline and two other fonts to words you find appropriate. Also use header tags.
C. Create links on the words e.g. “Wi-Fi” and “LAN” to link them to Wikipedia pages.
D. Insert an image and create a link such that clicking on image takes user to another page.
E. Change the background color of the page. At the bottom create a link to take user to the top of the page.

2. HTML Tables

A. Create at able to show your class time-table.
B. Use tables to provide layout to your HTML page describing your university infrastructure.
C. Use and tags to provide a layout to the above page instead of a table layout.
D. Use frames such that page is divided into 3 frames 20% on left to show contents of pages, 60% in center to show body of page, remaining on right to show remarks.
E. Embed Audio and Video into your HTML webpage.

3. CSS

A. Apply in-line CSS to change colors of certain text portion, bold, underline and italics certain words in your HTML web page. Also change background color of each paragraph using in- line CSS.
B. Write all the above styling in CSS indifferent file (.css) and link it to your webpage such that changes made in CSS file are immediately reflected on the page. Group paragraphs into single class and add styling information to the class in CSS.
C. Create a simple form to submit user input like his name, age, address and favourite subject, movie and singer.
D. Add few form elements such as radio buttons, check boxes and password field. Add a submit Button at last.

4. JavaScript

A. Create a form similar to the one in previous experiment. Put validation checks on values entered by the user using JavaScript (such as age should be a value between 1and 150). Write a Java Script program to display information box as soon as page loads.
B. Write a Java Script program to change background color after 5 seconds of page load.
C. Write a JavaScript program to dynamically bold, italic and underline words and phrases based on user actions.
D. Write a JavaScript program to display a hidden div (e.g. showing stats of a player when user clicks on his name).

5. PHP

A. Design a PHP script to read user input from form controls (text boxes, radio buttons, dropdowns) and display the submitted data.
B. Create a PHP program to handle file upload and store the file information in a specific directory.
C. Develop a PHP application to connect to a MySQL database, insert form data, and display retrieved records
D. Implement a PHP program to open, read, write, append, and delete text files and list all files in a directory.

6. XML

A. Create an XML document with custom tags, attributes, and define a DTD to validate its structure.
B. Write a PHP program to read and parse XML data using SimpleXML and display it in tabular format..
C. Develop a PHP script to connect to a MySQL database and insert XML data into corresponding database tables.

7. JDBC

A. Establishing JDBC connection to any database (oracle/mySql, H2)
B. Search the contents of Employee table/students table in JDBC
C. Perform database CRED operations
D. using simple statements, prepared statements and callable statements

8. Servlets

A. Create user registration web Application.
B. At the server end, write code to retrieve contents of request object and show them to the user. Match user input password with predefined password and show "Valid User" or "Invalid User"
C. A web Application name as input and on submits it should show hello <name>it shows start time at the right top corner of the page and provide a logout button. On clicking logout button should go to logout page shows Thank you <name> with duration of usage.
D. A web application that takes name and age from html page. If age less than 18 it should show Hello <name> You are not authorized to visit the site. Otherwise, welcome <name> to website.
E. A web Application name as input and on submit it should show hello <name> it shows start time at the right top corner of the page and provide a logout button. On clicking logout button should go to logout page shows Thank you <name> with duration of usage.
F. A web application that takes name and age from html page. If age less than 18 it should show Hello <name> You are not authorized to visit the site. Otherwise, welcome <name> to website.
G. Create XML Document that contain 10 users' information writes a java program user Id as input and returns user details from XML file by using DOM Parser.
H. Create XML Document that contain 10 users' information write a java program user Id as input and returns user details from XML file by using SAX Parser.

9. APPLETS

A. Create a web page using JavaScript to display an image and play a sound with button-triggered events.
B. Develop a PHP script to process HTTP form requests (GET/POST), display submitted data, and manage sessions.
C. Create a PHP-based login page that uses cookies and session tracking to display a welcome message to the user

10. DATABASE HANDLING

A. Develop a PHP-based login and registration system using sessions and database validation
B. Install and configure a WordPress CMS locally or on a web host, and create pages using templates, menus, and plugin.
C. Create a simple blog using Joomla CMS by adding components, setting permissions, and customizing modules and menus.

V:TEXT BOOKS:

1. Harvey Deitel, Abbey Deitel, Internet and World Wide Web: How To Program 5th Edition.
2. JavaScript: The Definitive Guide by David Flanagan, O'Reilly Media, Latest Edition.
3. Web Technologies: HTML, JavaScript, PHP, Java, JSP, ASP.NET, XML and Ajax by Uttam K. Roy, Oxford University Press
4. Michael Morrison XML Unleashed Tech media SAMS.
5. Web Technologies, Uttam K Roy, Oxford University Press

VI: REFERENCE BOOKS:

1. John Pollock, Javascript - A Beginners Guide, 3rd Edition — Tata McGraw-Hill Edition.
2. Keyur Shah, Gateway to Java Programmer Sun Certification, Tata McGraw Hill, 2002.

ELECTRONICS RESOURCES

1. https://www.w3schools.com/html/
2. https://www.php.net/manual/en/book.filesystem.php
3. https://www.tutorialspoint.com/xml/xml_parsers.htm
4. https://docs.oracle.com/javaee/7/tutorial/servlets.htm
5. https://docs.joomla.org/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

NODE JS/ REACT JS/ DJANGO, UI DESIGN - FLUTTER								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25IT408SD	**Skill Development**	0	0	2	1	40	60	100
Contact Classes: 00	**Tutorial Classes: 00**	**Practical Classes: 30**				**Total Classes: 30**		
Prerequisite: Object Oriented Programming through Java, HTML Basics.								

I. COURSE OVERVIEW:

The UI Design – Flutter Lab is a hands-on course that focuses on building modern, responsive, and interactive user interfaces for both web and mobile applications. This course integrates frontend web technologies, backend development, RESTful services, and cross-platform mobile app development using Flutter.

II. COURSE OBJECTIVES:

The students will try to learn:

1. To implement responsive web pages using HTML, CSS3, Bootstrap, and JavaScript validation.
2. To develop full-stack applications with Node.js, Express, React.js, and secure them using JWT.
3. To build and integrate RESTful APIs using Node.js and Django with frontend CRUD operations.
4. To create Flutter apps with responsive layouts, state management, and custom widgets.
5. To enhance Flutter apps by integrating REST APIs and adding animations for better user experience.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Design responsive web pages using HTML, CSS, Bootstrap, and JavaScript.
CO2:	Develop backend applications using Java and Node.js.
CO3:	Build single-page applications with React.js.
CO4:	Create Flutter apps with custom widgets, layouts, and forms.
CO5:	Integrate APIs, add animations, and perform UI testing in Flutter apps.

IV: COURSE CONTENT:

List of Experiments:

1. Build a responsive website with registration, login, catalog, and cart pages using HTML, CSS3, Bootstrap, and JavaScript validation.
2. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages
3. 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
4. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman).

5. Implement JWT authentication in Node.js to create secure endpoints.
6. Create a TODO application in react with necessary components and deploy it into github.
7. Integrate a React frontend with Node.js backend to perform CRUD operations on a shared dataset (e.g., MongoDB, MySQL.)
8. Build a Django REST API for managing student data, tested using Postman.
9. Install Flutter and Dart SDK; Write Dart programs on data types, control flow, Functions, Class & Objects and collections to understand syntax and features.
10. Create a Flutter app showcasing common widgets (Text, Image, Container, Card, ListView).
11. Design a responsive UI using Row, Column, Stack, media queries, and breakpoints.
12. Implement screen navigation using Navigator and named routes.
13. Use stateful & stateless widgets; manage state using Provider or setState.
14. Design a form in Flutter with input fields, validation, and error handling (e.g., student registration form).
15. Fetch and display data from a REST API (e.g., weather or student info) in Flutter.
16. Add basic animations (fade or slide) to a Flutter app for enhancing UI interactions.

IV. TEXT BOOKS:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.

V. 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, A Press.

VI. ELECTRONICS RESOURCES:

1. https://www.w3schools.com/
2. https://react.dev/
3. https://expressjs.com/
4. https://docs.djangoproject.com/en/stable/
5. https://docs.flutter.dev/development/ui/animation/

TEEGALA KRISHNA REDDY ENGINEERING COLLEGE

(UGC-Autonomous)

Medbowli, Meerpet, Balapur(M), Hyderabad, Telangana- 500097

COURSE CONTENT

INDIAN KNOWLEDGE SYSTEM								
IV Semester: IT					**Regulations:** R25			
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIE**	**SEE**	**Total**
25VA400HS	**Humanities & Social Science**	1	0	-	1	40	60	100
Contact Classes: 15	**Tutorial Classes: 00**	**Practical Classes: Nil**				**Total Classes: 15**		
Prerequisite: Nil								

I. COURSE OVERVIEW:

Bharat is considered one of the oldest civilizations of the world. Some of the archaeological evidences proved the existence of Indus Valley Civilization in 7000 B.C. Bhartiya traditions, culture, cultural activities, rituals, sacraments, painting, art of dancing, art of singing etc. is being practised till the modern times without knowing scientific approaches behind that. Eternity of Indian knowledge system proved itself that not only many rituals but also many traditions, many streams of knowledge like astrology, mathematics, physics, chemistry, biology, language studies, yoga and meditation had been following from the starting till now with some changes, in the form of traditions.

This course is for undergraduate students to inculcate Indian values. It will promote advance study and inter disciplinary research on all aspects of the Indian knowledge system.

II. COURSE OBJECTIVES:

1. To provide a tribune of the rich culture and traditions of Indian knowledge system to students of various disciplines.
2. To introduce historical account on the education and scientific literature available in ancient Indian traditions and its connections with ancient Indian Philosophy.
3. To give insights about the applications of Bharatiya Jnana Parampara.
4. To introduce Indian approach towards health and wellbeing.
5. To elaborate vast contribution of ancient Indian researchers, engineers, scientists and architects to the modern world.

III. COURSE OUTCOMES:

At the end of the course students should be able to:	
CO1:	Understand nature, scope and related fields of Indian knowledge system.
CO2:	Demonstrate the scientific literature available in ancient Indian traditions
CO3:	Understanding the application of Bharatiya Jnana Parampara
CO4:	Understand Indian approach towards Wellbeing
CO5:	Appreciate vast contribution of ancient Indian researchers, engineers, scientists and architects to the modern world.

IV: COURSE CONTENT:

UNIT- I: (14 Hrs)

Introduction to Indian Knowledge Systems

Meaning, Nature, Scope and Salient Aspects of Bharatiya Jnana Parampara - Introduction to Vedas, Upanishads,Vidya, Kala, Jnana, Shastra - Practices and Continuity of Tradition.

UNIT – II: (14 Hrs)

Overview of History of Indian Education and Scientific Literature
Gurukul System - Role of Sanskrit in Natural Language Processing - Scientific Literature - Vedic Literature - Available Scientific Treatises – Interlinkings.

UNIT - III: (10 Hrs)

Introduction to Scientific Theories from Pure Sciences from Ancient Indian Knowledge Systems
Overview of theories from available ancient Indian Literature about Physics, Chemistry and Mathematics - Interlinkings and applications.

UNIT - IV: (12 Hrs)

Introduction to Ancient Indian Wellness Systems
Concept of Wellness – Yoga System - Ayurveda System - Ancient Indian Aesthetics.

UNIT - V: (10 Hrs)

Development of Engineering, Science, Technology & Fine Arts in India Various Industries - Silk, Cotton and Ship Building - Evolution of Indian Fine Arts — Cave and Temple Architecture, Vastu - Vidya, Sculpture, Forts and Stepwells, Observatories and Paintings - Music and Natyakala - Cultural Traditions & Folk Arts.

V:TEXT BOOKS:

1. B. Mahadevan, Bhat Vinayak and Nagendra Pavan R.N., (2022) '*Introduction to Indian Knowledge Systems: Concepts and Applications*' PHI learning PVT, New Delhi ISBN [9789391818203]
2. Dharmapal (1971) '*Indian Science and Technology in the Eighteenth Century*'. Other India Press, Goa.
3. Kapil Kapoor, Singh Avdhesh Kumar, (2005) '*Indian Knowledge Systems*' D.K. Printworld (P) Ltd. ISBN 10: 8124603367 / ISBN 13: 9788124603369
4. Chakradeo, Ujwala, Temples of Bharat, Aayu Publications, New Delhi, 2024.
5. D.N. Bose, S.N. Sen and B. V. Subbarayappa, *A Concise History of Science in India*, Indian National Science Academy, New Delhi, 2009.

VI: REFERENCE BOOKS:

1. Datta B. and A. N. Singh, *History of Hindu Mathematics: Parts I and II*, Asia Publishing House, Bombay, 1962.
2. Kapoor, K. (2021), *Indian Knowledge System: Nature, Philosophy, Character in Indian Knowledge System*, vol. 1, Pub. Indian Institute of Advanced Studies, Shimla
3. Mahadevan, B., Bhat, V.R., Pavana, N. (2022), Philosophical Systems, in Introduction to Indian Knowledge System, Pub. PHI Learning, New Delhi.
4. Mahadevan, B., Bhat, V.R., Pavana, N. (2022), Knowledge: Framework and Classification, in Introduction to Indian Knowledge System, Pub. PHI Learning, New Delhi.

ELECTRONICS RESOURCES

1. https://iksin dia.org/index.php
2. Official Website of IKS- Indian Knowledge System
3. https://www.youtube.com/watch?v=uKcf-hSlcUE
4. Address by Prof Kapil Kapoor | Indian Institute of Advanced Study (FDP 2021)
5. https://www.youtube.com/watch?v=MDJTXNiH2_A
6. Mukul Kanitkar on Bharatiya Knowledge System.