MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(Autonomous Institution-UGC, Govt. of India)

Accredited by NAAC with 'A+' Grade | Programmes Accredited by NBA

NIRF Institutional Ranking 2025 - Rankband(201-300), MHRD, Govt. of India

Approved by AICTE, Affiliated to JNTUH, ISO 9001:2015 Certified Institution

Maisammaguda, Dhulapally, Secunderabad 500100.

B.Tech. in COMPUTER SCIENCE AND ENGINEERING (CYBER SECURITY)

Course Structure & Syllabus (R-26 Regulations)

AY 2026 – 2027

I Year I Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	2600BS01	Matrices and Calculus	3	1	0	4
2.	2600BS04	Advanced Engineering Physics	3	0	0	3
3.	2605ES01	Programming for Problem Solving	3	1	0	4
4.	2602ES01	Basic Electrical and Electronics Engineering	3	0	0	3
5.	2603ES01	Computer Aided Engineering Graphics	2	0	2	3
6.	2600BS62	Advanced Engineering Physics Lab	0	0	2	1
7.	2605ES61	Programming for Problem Solving Lab	0	0	2	1
8.	2602ES61	Basic Electrical Engineering and Electronics Lab	0	0	2	1
9.	2603ES61	Engineering Workshop	0	0	2	1
10.		Induction Program				
		Total Credits	**14**	**2**	**10**	**21**

I Year II Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	2600BS03	Ordinary Differential Equations and Vector Calculus	3	1	0	4
2.	2600BS02	Engineering Chemistry	3	0	0	3
3.	2605ES03	Data Structures	3	0	0	3
4.	2605ES02	Python Programming	3	0	0	3
5.	2600HS01	English for Skill Enhancement	3	0	0	3
6.	2600BS61	Engineering Chemistry Lab	0	0	2	1
7.	2605ES63	Data Structures Lab	0	0	2	1
8.	2600HS61	English Language and Communication Skills Lab	0	0	2	1
9.	2605ES62	Python Programming Lab	0	0	2	1
		Total Credits	**15**	**1**	**08**	**20**

II YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1.	2605PC01	Discrete Mathematics	2	1	0	3
2.	2605PC03	Object Oriented Programming through Java	3	0	0	3
3.	2662PC01	Mathematical Foundations of Cryptography	3	0	0	3
4.	2605PC05	Database Management System	3	0	0	3
5.	2605PC06	Software Engineering	3	0	0	3
6.	2600HS04	Innovation and Entrepreneurship	2	0	0	2
7.	2605PC61	Object Oriented Programming through Java Lab	0	0	2	1
8.	2662PC61	Mathematical Foundations of Cryptography lab	0	0	2	1
9.	2605PC63	Database Management Systems Lab	0	0	2	1
10.	2662SD01	Data Visualization-Python/Power BI/Tableau/Google Chart	0	0	2	1
11.	2600MC01	Indian Knowledge System	1	0	0	0
		Total Credits	**17**	**1**	**10**	**21**

II YEAR II SEMESTER

S.No.	Course Code	Course Title	L	T	P	Credits
1.	2600BS06	Mathematical and Statistical Foundations	2	1	0	3
2.	2605PC13	Computer Organization	3	0	0	3
3.	2605PC12	Formal Language and Automata Theory	2	1	0	3
4.	2605PC07	Computer Networks	3	0	0	3
5.	2605PC04	Operating Systems	3	0	0	3
6.	2605PC62	Operating Systems lab	0	0	2	1
7.	2600BS64	Computational Mathematics Lab using Python	0	0	2	1
8.	2605PC65	Computer Networks lab	0	0	2	1
09.	2662SD02	Linux Programming, UI Design-Flutter	0	0	2	1
10.	2600IP01	Innovative Product Development-1	1	0	0	1
		Total Credits	**14**	**2**	**08**	**20**

III YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1.	2662PC02	Network Management System and Operations	3	1	0	4
2.	2600HS03	Business Economics and Financial Analysis	2	1	0	3
3.	2605PC09	Design and Analysis of Algorithms	2	1	0	3
4.	PE-I	Professional Elective-I	3	0	0	3
5.	OE-I	Open Elective-I	2	0	0	2
6.	2662PC62	Network Management System and Operations Lab	0	0	2	1
7.	2605PC67	Design and Analysis of Algorithms Lab	0	0	2	1
8.	2662PR01	Field Based Research Project	0	0	4	2
9.	2662SD03	Full Stack Development-Node JS/React JS/Angular JS/ Django	0	0	2	1
10.	2600MC02	Gender Sensitization*/Human Values and Professional Ethics *	1	0	0	0
11.	2600HS63	English for Employability Skills lab	0	0	2	1
		Total Credits	**13**	**3**	**12**	**21**

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

III YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1.	2662PC03	Cyber Security Essentials	3	0	0	3
2.	2662PC04	Information Security	3	1	0	4
3.	2600HS02	Fundamentals of Management	3	0	0	3
4.	PE-II	Professional Elective-II	3	0	0	3
5.	OE-II	Open Elective-II	2	0	0	2
6.	2662PC63	Cyber Security Essentials Lab	0	0	2	1
7.	2662PC64	Information Security Lab	0	0	2	1
8.	2605PC69	Development and Operations Lab	0	0	2	1
9.	2600HS62	Professional English	1	0	0	1
10.	2662SD04	Cyber Defense Techniques/Ethical Hacking	0	0	2	1
11.	2600ES01	Environmental Science	1	0	0	0
		Total Credits	**16**	**1**	**08**	**20**

IV YEAR I SEMESTER (25 Hours)

S. No.	Course Code	Course Title	L	T	P	Credits
1.	2662PC05	Vulnerability Assessment & Penetration Testing	3	0	0	3
2.	2662PC06	Cyber Crime Investigation & Digital Forensics	3	0	0	3
3.	2662PC07	Risk Management and Secure Coding Practices	2	1	0	3
4.	PE-III	Professional Elective-III	3	0	0	3
5.	PE-IV	Professional Elective-IV	3	0	0	3
6.	OE-III	Open Elective - III	2	0	0	2
7.	2662PC65	Vulnerability Assessment & Penetration Testing Lab	0	0	2	1
8.	2662PC66	Cyber Crime Investigation & Digital Forensics Lab	0	0	2	1
09.	2662PR02	Industry Oriented Mini Project/ Internship	0	0	4	2
10.	2600MC03	Research Methodology and IPR	1	0	0	0
		Total Credits	**17**	**1**	**08**	**21**

IV YEAR II SEMESTER (48 Hours)

S. No.	Course Code	Course Title	L	T	P	Credits
1.	PE-V	Professional Elective - V	3	0	0	3
2.	PE-VI	Professional Elective - VI	3	0	0	3
3.	2662PR03	Project Work	0	0	36	12
4.	2600IP02	Innovative startup	0	0	6	2
		Total Credits	**6**	**0**	**42**	**20**

PROFESSIONAL ELECTIVES

Professional Elective-I

2662PE01	Artificial Intelligence in Cyber Security
2612PE11	Advanced Web Technologies
2667PE19	Data Warehousing and Business Intelligence
2605PE06	Cloud Computing
2612PE09	Mobile Application Development
2612PE04	Internet of Things

Professional Elective-2

2662PE02	Machine Learning in Cyber Security
2667PE04	Fundamentals of Data Science
2605PE12	Information Retrieval Systems
2605PE15	Ad-hoc & Sensor Networks
2662PE10	Mobile Application Security
2662PE29	IOT Security

Professional Elective-3

2666PE02	Principles of Deep Learning
2662PE04	Cyber Laws and IT Security
2662PE05	Privacy Preserving Data Mining
2662PE06	Security in Computing
2662PE09	Malware Analysis
2605PE08	Big Data Analytics

Professional Elective-4

2662PE11	Digital Watermarking and Steganography
2662PE13	Authentication Techniques
2662PE20	Cyber Threat Intelligence and Security Analyst
2662PE21	Information System Audit & Assurance
2662PE22	Web and Database Security
2666PE03	Generative AI Essentials

Professional Elective-5

2666PE13	Agentic AI Essentials
2612PE05	Blockchain Technology
2666PE01	Large Language Models
2662PE23	Cloud Security
2662PE24	Data Analytics for fraud Detection
2666PE05	Natural Language Processing concepts and Applications

Professional Elective-6

2662PE03	Security Incident Response Management (SOC)
2662PE25	Social Media Security
2662PE26	Quantum Cryptography
2662PE27	Green Computing
2662PE28	Data Privacy
2666PE15	Retrieval-Augmented Generation (RAG)

OPEN ELECTIVES

Open Elective-I:

2662OE01	Cyber Security
2662OE02	Threat Intelligence
2662OE03	Security Incident Response Management (SOC)
2662OE04	Authorization and Authentication Techniques

Open Elective-II:

2662OE05	Cyber Laws and Digital Security
2662OE06	Basics of Ethical Hacking
2662OE07	Social Media Security
2662OE08	Web Application Security

Open Elective-III:

2662OE09	Intrusion Detection System
2662OE10	Malware Analysis
2662OE11	Ad-hoc and Sensor Networks
2662OE12	Web and Database Security

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600BS01) MATRICES AND CALCULUS

B. TECH I YEAR I SEMESTER L T P C

CSE, CSE(AI&ML), CSE(DS), CSE(CS), IT, ECE & EEE 3 1 0 4

Course Objectives:

- To learn the concepts of matrix rank, matrix inverse, and methods for solving homogeneous and non-homogeneous systems of linear equations.
- To study the concepts of eigenvalues, eigenvectors, diagonalization, the Cayley–Hamilton theorem, and quadratic forms.
- To understand the concepts of limits, continuity, Mean Value Theorems, and Taylor's series.
- To study the concepts of partial differentiation for determining the maxima and minima of functions of two variables.
- To apply multiple integration techniques and coordinate transformations for evaluating double and triple integrals.

Course Outcomes: After learning the contents of this course, the student will be able to:

- Solve systems of linear equations and determine the rank and inverse of matrices using appropriate matrix methods.
- Apply the concepts of eigenvalues, eigenvectors, diagonalization, the Cayley–Hamilton theorem, and quadratic forms to solve matrix-related problems.
- Analyze the properties of continuous functions using Mean Value Theorems to solve mathematical problems.
- Apply Taylor's series to approximate functions and solve mathematical problems.
- Determine the maxima and minima of functions of two variables using partial differentiation.
- Evaluate double and triple integrals using appropriate integration techniques and coordinate transformations.

UNIT-I: Matrices ***8 L***

Rank of a matrix by Echelon form and Normal form - Inverse of Non-singular matrices by Gauss-Jordan method. System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations. Gauss Seidel Iteration Method.

UNIT-II: Eigen values and Eigen vectors ***10 L***

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

UNIT-III: Single Variable Calculus *10 L*

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).

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

Definitions of Limit and continuity - Partial Differentiation: Euler's Theorem – Total derivative - Jacobian - Functional dependence & independence. Applications: Maxima and minima of functions of two variables.

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

Evaluation of Double Integrals (Cartesian and polar coordinates) - change of order of integration (only Cartesian form) - Change of variables for double integrals (Cartesian to polar). Evaluation of Triple Integrals- Change of variables for triple integrals (Cartesian to Spherical and Cylindrical polar coordinates).

TEXTBOOKS:

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

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 9thEdition,2013.
2. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi,38th Reprint, 2022.
3. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson, 9thEdition, Reprint, 2002.
4. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
5. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

ADVANCED ENGINEERING PHYSICS

Course Structure & Syllabus (R-26 Regulations) -AY 2026-2027

	L	T	P	C
I B.Tech. I- SEMESTER: **CSE (AIML, DS), CS, IT & EEE**				
II- SEMESTER: **ECE & CSE**	3	0	0	3

Pre-requisites: 10+2 Physics

Course Objectives:

1. To study crystal structures, defects, and material characterization techniques like XRD.
2. To familiarize students with demonstrating skills in scientific enquiry, problem solving techniques and to develop competency and understanding the concepts found in quantum mechanics.
3. To introduce quantum computing principles, quantum gates, and basic quantum algorithms.
4. To solve non-traditional problems that potential drawn on knowledge in multiple areas of physics.

Course Outcomes:

1. **CO1**: To understand the crystal structures, identify defects, and apply XRD techniques for material characterization.
2. **CO2**: To apply quantum mechanical principles to explain particle behaviour and energy band formation in solids.
3. **CO3**: To analyse the advance concepts linear algebra for quantum computation along operators', Dirac's Bra and Ket notations and their properties.
4. **CO4**: To understand quantum computing concepts, use quantum gates and explain basic quantum algorithms.
5. **CO5**: To classify magnetic and dielectric materials and explain their properties, synthesis, and applications.
6. **CO6**: To explain the principles of lasers and fibre optics and their applications in communication and sensing.

UNIT - I: 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.

UNIT - II: 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, formation of energy bands, origin of bandgap, classification of solids, concept of discrete energy levels and quantum confinement in nanomaterials.

UNIT - III: Quantum Computing

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

UNIT - IV: Magnetic and Dielectric Materials

Introduction to magnetic materials, origin of magnetic moment-classification of magnetic materials, hysteresis, Weiss domain theory of ferromagnetism, soft and hard magnetic materials, synthesis of 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: 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.

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
4. *A Textbook of Advanced Engineering Physics,* Dr. T.V.S Arun Murthy, Dr. M. N. Avadhanulu, S. Chand, First edition, 2026.
5. *Engineering Physics* by P.K. Palanisamy, SciTech Publications.

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.

USEFUL LINKS

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

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600ES01) PROGRAMMING FOR PROBLEM SOLVING

B.Tech I Year I Sem L T P C

3 0 0 3

Course Objectives:

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

Course Outcomes: The student will learn

1. To write algorithms and to draw flowcharts for solving problems.
2. To convert the algorithms/flowcharts to C programs.
3. To code and test a given logic in the C programming language.
4. To decompose a problem into functions and to develop modular reusable code.
5. To use arrays, pointers, strings and structures to write C programs.
6. Searching and sorting problems.

UNIT - I: Overview of C:C Language Elements, Variable Declarations and Data Types, Executable Statements, General Form of a C Program, Arithmetic Expressions.

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: 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, Formal Output Parameters as Actual Arguments.

UNIT- III: 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, Enumerated Types, Multidimensional Arrays.

Strings: String Basics, String Library Functions: Assignment and Substrings, String Comparison.

UNIT - IV: Recursion: The Nature of Recursion, Tracing a Recursive Function,

Structure and Union Types: User-Defined Structure Types, Structure Type Data as Input and Output Parameters, Union Types.

UNIT - V: 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 algorithm

TEXTBOOKS:

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 Leaming, (3rd Edition).

REFERENCEBOOKS:

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.

B.Tech. I Year -I Sem.

L/T/P/C
3/0/0/3

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

2402ES01: BASIC ELECTRICAL AND ELECTRONICS ENGINEERING

(Common to CSE, AIML, DS, CS, IT)

UNIT I:

Diode Characteristics:

Semiconductors –classification -PN Junction Diode, Avalanche breakdown, zener break down, forward bias and reverse bias characteristics, diode resistance an capacitance, diode models (Ideal, simplified, piecewise linear) Zener diode-I-V characteristics and zener diode as voltage regulator.

UNIT II:

Bipolar Junction Transistor (BJT)

Structure and working principle of BJT, Transistor as switch, configuration: Common base (CB), Common emitter (CE), Common collector(CC), Common Base(CB), input and output characteristics. Applications of transistor.

UNIT III :

Number Systems and Boolean algebra

Overview of Number Systems, 1's complement and 2's Complement, addition, subtraction using 1's complement and 2's Complement, BCD codes, Excess-3 codes, HEX codes, parity bit generator. Boolean Algebra-Basic Theorems and properties, Minimization of Boolean function using K-Maps up to four variables.

UNIT IV:

DC and AC Circuits

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

A.C. Fundamentals: Equation of AC Voltage and current, waveform, time period, frequency, amplitude, phase, phase difference, average value, RMS value, form factor, and peak factor, (Simple Numerical problems).

UNIT V Electrical Machines

Construction and working principle of DC Motor, DC Generator, Single Phase Transformer, Three Phase Induction Motors and Alternator.

TEXT BOOKS:

1. Basic Electrical Engineering - D.P. Kothari and I.J. Nagrath, Tata McGraw Hill.
2. A text book of Electrical Technology Vol- 1, 2 by B.L.Theraja and A K Theraja- S Chand Publications.
3. Electronic Devices and Circuits, S. Salivahanan, N. Suresh Kumar, A. Vallavaraj, Tata McGraw-Hill companies.
4. Switching Theory and Logic Design by A. Anand Kumar, PHI Publications.

REFERENCE BOOKS:

1 Basic Electrical Engineering, T.K. Naga sarkar and M.S. Sukhija, Oxford University Press

2. M. Morris Mano, Digital Design, 3rd Edition, Prentice Hall of India Pvt. Ltd., 2003

COMPUTER AIDED ENGINEERING GRAPHICS (2603ES01)

B.Tech. I Year. I Sem **L / T/ P/ C**

2/ 0 / 2/ 3

Course Objectives:

- To impart conceptual knowledge about AutoCAD software in creating, editing and manipulating 2D engineering drawings.
- To inculcate the required skills to apply engineering drawing principles effectively for accurately sketching conics, scales and cycloids.
- To enhance visualization skills for better interpreting the position of points and lines, enabling accurate representation of their orthographic projections.
- To familiarize drafting skills for constructing orthographic views of planes, solids and development of surfaces in diverse orientation.
- To develop the competency in creating isometric views from orthographic views and vice-versa.

Course Outcomes:

Upon the completion of the course students should be able to

- Demonstrate proficiency in using AutoCAD software to create, edit, and manipulate 2D engineering drawings.
- Apply the concepts of engineering drawing for sketching conic sections, scales and cycloids.
- Analyze position of points and lines for representing their orthographic projections.
- Sketch orthographic projections of planes, solids and development of surfaces to analyze their different orientations
- Apply the principle of projections for sketching the isometric views.
- Interpret the given orthographic projections to convert isometric view and vice versa

UNIT-I

Introduction to AutoCAD Software: Menu Bar, Ribbon – Draw and Modify Toolbar, Dimension, Status Bar, Command Bar, etc., Basic geometrical constructions using AutoCAD commands.

Introduction to Engineering Drawing: Principles of Engineering drawing and their significance, Conventions.

Conic Sections: Construction of Ellipse, Parabola, Hyperbola - General method only. Scales, Plain and Diagonal.

Cycloidal Curves: Cycloid, Epicycloid and Hypocycloid.

UNIT-II

Principles of Orthographic Projections: Conventions; Projections of Points; Projections of lines - Parallel to one Plane and Perpendicular to other Plane, Inclined to one Plane and Parallel to other Plane, inclined to both the planes.

UNIT-III

Projections of Planes: Surface Parallel to one Plane and Perpendicular to other Plane, Surface Inclined to one Plane and Perpendicular to other Plane, Surface Inclined to both the Planes.

UNIT-IV

Projection of Regular Solids: Prisms, Pyramids, Cylinder and Cone – Axis parallel to one plane and perpendicular to other plane, Axis Inclined to one plane and parallel to other plane.

Development of Surfaces: Prism, Cylinder, Pyramid and Cone.

UNIT-V

Principles of Isometric projection: Isometric Scale, Isometric Views, Conventions; Isometric Views of lines, Planes, Simple Solids only. Conversion of Isometric Views to Orthographic View and Vice-versa, Conventions. Introduction to Solid Modeling: Creation of simple solid models relevant to the domain.

TEXT BOOKS

1. Engineering Drawing, N.D. Bhatt – N.D. Bhatt & V.M Panchal, 48th Edition, 2005 Charotar Publishing House, Gujarat.
2. Engineering Drawing / Basant Agarwal and MC Agarwal / McGraw Hill

REFERENCES

1. Engineering drawing – P.J. Shah .S.Chand Publishers.
2. Engineering Drawing- Johle/Tata Macgraw Hill Book Publisher.
3. Computer Aided Engineering Drawing – S. Trymbaka Murthy, - I.K. International Publishing House Pvt. Ltd., New Delhi, 3rd revised edition-2006.
4. "Computer Aided Engineering Drawing"by Dr. M H Annaiah, Dr C N Chandrappa and Dr B Sudheer Premkumar Fifth edition, New Age International Publishers.
5. Engineering Drawing by K.Venu Gopal& V.Prabu Raja New Age Publications.

ADVANCED ENGINEERING PHYSICS LABORATORY

I B.Tech. I- SEMESTER: CSE (AIML, DS), CS, IT & EEE

II- SEMESTER: ECE & CSE

L	T	P	C
0	0	2	1

Course Objectives:

1. To provide practical exposure to advanced concepts in solid-state and modem 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 oflasers and optical fibers through hands-on experiments.
5. To develop skills in data analysis, interpretation, and scientific reporting.

Course Outcomes:

1 COl: To synthesize and analyze nanomaterials such as magnetite (Fe3Q4) using chemical methods.
2. CO2: To determine key electrical, magnetic, and optical properties ofsemiconductors and other functional materials.
3. C03: To characterize semiconductors using Hall effect and energy gap measurement techniques.
4. C04: To demonstrate working knowledge oflaser systems and optical fiber parameters through experimental study.
5. COS: To apply scientific methods for accurate data collection, analysis, and technical report writing.
6. C06: To calculate the dielectric constant ofa given material and its applications.

List of Experiments:

1. Synthesis ofmagnetite (Fe3Q4) powder using sol-gel method.
2. Determination ofenergy gap ofa semiconductor.
3. Determination ofHall coefficient and carrier concertation ofa given semiconductor.
4. To determine the magnetic induction at the centre and at several points on the axis ofa circular coil
5. Study ofB-H curve ofa ferro magnetic material.
6. Study ofP-E loop ofa given ferroelectric crystal.
7. Determination ofdielectric constant ofa given material.
8. Determination of Curie's temperature ofa given ferroelectric material.
9. Study ofV-1 & L-1 characteristics ofa given laser diode.
10. a. Determination ofnumerical aperture ofa given optical fibre.
    b. Determination ofbending losses ofa given optical fibre.

Note: Any 8 experiments are to be performed.

2605ES61: PROGRAMMING FOR PROBLEM SOLVING LAB

B.Tech. I Year I Sem.	L	T	P	C
[Note:The programs may be executed using any available Open Source/Freely available IDE Some of the Tools available are:	*0*	*0*	*2*	*1*

CodeLite: https://codelite.org/ Code::Blocks:

http://www.codeblocks.org/

DevCpp : http://www.bloodshed.net/devcpp.html Eclipse:
http:/lwww.eclipse.org

This list is not exhaustive and is NOT in any order of preference]

Course Objectives: The students will learn the following:

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

Course Outcomes: The candidate is expected to be able to:

1. Formulate the algorithms for simple problems
2. Translate given algorithms to a working and correct program
3. Correct syntax errors as reported by the compilers
4. Identify and correct logical errors encountered during execution
5. Represent and manipulate data with arrays, strings and structures
6. Use pointers of different types
7. Create, read and write to and from simple text and binary files
8. Modularize the code with functions so that they can be reused

PRACTICE SESSIONS:

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.

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 5and rows = 3, the output should be: 5xl=5

5x2=10

5 x 3 = 15

d) Write a program that shows the binary equivalent ofa given positive number between Oto255.

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 ofTwo Matrices

II. Multiplication ofTwo 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 ofthe 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 ifthe 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 ofintegers 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 ofintegers using selection sort in descending order

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

f) Write a C program that sorts a given array ofnames.

TEXTBOOKS:

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

B.Tech I -Year I -Sem. **L/T/P/C**
0/0/2/1

2602PC05: Basic Electrical Engineering and Electronics Lab

The following experiments are required to be conducted compulsory experiments:

1. Magnetization characteristics of DC shunt generator (Determination of critical field resistance and critical speed)
2. Load test on DC shunt generator (Determination of characteristics)
3. Load test on DC series generator (Determination of characteristics)
4. Hopkinson's test on DC shunt machines (Predetermination of efficiency)
5. Swinburne's test (Predetermination of efficiency)
6. Brake test on DC compound motor (Determination of performance curves)
7. OC and SC Test on Single Phase Transformer
8. Three Phase Transformer: Verification of Relationship between Voltages and Currents (Star-Delta, Delta- Delta, Delta-star, Star-Star)

In addition to the above eight experiments, at least any two of the experiments from the following listare required to be conducted:

1. Brake Test on DC shunt motor (Determination of performance curves)
2. Load Test on DC compound generator (Determination of characteristics).
3. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single-Phase Transformer
4. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
5. Speed control of DC shunt motor
6. Modeling of DC Machine using simulation tools.
7. Equivalent circuit of Transformer using simulation tools.

TEXT BOOKS:

1. P. S. Bimbhra, "Electrical Machinery", Khanna Publishers, 2011.
2. I.J. Nagrath and D. P. Kothari, "Electric Machines", McGraw Hill Education, 2010.

REFERENCE BOOKS:

1. PrithwirajPurkait, IndrayudhBandyopadhyay, "Electrical Machines", Oxford, 2017.
2. M. G. Say, "Performance and design of AC machines", CBS Publishers, 2002.
3. A. E. Fitzgerald and C. Kingsley, "Electric Machinery", New York, McGraw Hill Education, 2013.
4. A. E. Clayton and N. N. Hancock, "Performance and design of DC machines", CBS Publishers, 2004.

B.Tech. I Year | L / T/ P/ C

Course Code: 2603ES61 | 0/ 0 / 2/ 1

ENGINEERING WORKSHOP

Course Objectives:

- To Study different hand operated power tools, uses and their demonstration.
- To gain a good basic working knowledge required for the production of various engineering products.
- To provide hands on experience about use of different engineering materials, tools, equipment's and processes those are common in the engineering field.
- To develop a right attitude, team working, precision and safety at work place.
- To study commonly used carpentry joints and to have practical exposure to various welding and joining processes.

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

- Identify and make use of various tools to perform a range of basic manufacturing operations in different trades to make/repair engineering components with workshop safety regulations.
- Illustrate knowledge of various trade operations based on requirements of the job.
- Illustrate knowledge of various trade tools based on requirements of the job.
- Interpret and establish residential wiring circuits according to given specifications and circuit diagram.
- Demonstrate working principles of power tools in different trades to use and to make with them engineering components.
- Develop model various basic prototypes to explore its functions and features of a innovative system.

TRADES FOR EXERCISES:

I. *Carpentry*

1. T- Lap Joint
2. Dovetail Lap Joint

II. *Fitting*

1. Semi-circular Fitting
2. V- Fitting

III. *Tin Smithy*

1. Rectangular Tray
2. Square Tin

IV. *House Wiring*

1. Two bulbs controlled by two independent switches (Series and parallel connection)
2. One bulb controlled by two-two way switches (Stair case connection).

V. *Foundry*

1. Single piece pattern
2. Split piece pattern

VI. *Black Smithy*

1. Round to Square
2. S - Hook

Trades for Demonstration:

1. Plumbing
2. Welding
3. Machine Shop
4. Power tools

TEXT BOOKS:

1. Workshop Manual, P. Kannaiah and K. L. Narayana, 3rd Edition, Scitech, 2015
2. Elements of Workshop Technology Vol.1 & 2, S. K. Hajra Choudhury, A. K. Hajra Choudhury and Nirjhar Roy, 13th Edition, Media Promoters & Publishers Pvt. Ltd., 2010.

REFERENCE BOOKS:

1. Workshop Manual / Venkat Reddy/ BSP
2. Workshop Manual / K Venu Gopal / Anuradha

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600BS02) ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS

B.TECH I YEAR II SEMESTER *L T P C*

CSE, CSE(AI&ML), CSE(DS), CSE(CS), IT, ECE & EEE 3 0 0 3

Course Objectives:

- To learn the methods of solving first-order and higher-order differential equations.
- To learn the concepts and properties of Laplace transforms.
- To learn the techniques of solving ordinary differential equations using Laplace transforms.
- To understand the concepts of vector-valued functions and vector differential operators used in engineering applications.
- To understand the properties of vector integration and apply Green's, Gauss's, and Stokes' theorems to line, surface, and volume integrals.

Course Outcomes: After learning the contents of this course, the student must be able to

- Identify whether a given first-order differential equation is exact or reducible to an exact equation and solve it using appropriate methods.
- Solve higher-order linear differential equations with constant coefficients using appropriate analytical techniques.
- Compute Laplace transforms of standard functions using the properties and theorems of Laplace transforms.
- Apply Laplace transform techniques, including the convolution theorem, to solve initial value problems involving linear ordinary differential equations.
- Apply the concepts of gradient, directional derivative, divergence, curl, and scalar potential to analyze vector fields.
- Evaluate line, surface, and volume integrals and apply Green's, Gauss's, and Stokes' theorems to solve vector calculus problems.

UNIT-I: First Order ODE **8 L**

Exact differential equations - Equations reducible to exact differential equations - linear and Bernoulli's equations - Orthogonal Trajectories (only in Cartesian Coordinates). Newton's law of cooling.

UNIT-II: Ordinary Differential Equations of Higher Order **10 L**

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

UNIT-III: Laplace Transforms 10L

Laplace Transform of standard functions - First shifting theorem - Laplace transforms of functions multiplied by 't' and divided by 't' - Laplace transforms of derivatives and integrals of function - Evaluation of integrals by Laplace transforms - Inverse Laplace transform, convolution theorem (All theorems without proof). Applications: solving Initial value problems (Linear ODEs with constant coefficients) by Laplace Transform.

UNIT-IV: Vector Differentiation 10 L

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

UNIT-V: Vector Integration 10 L

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

TEXTBOOKS:

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

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 9thEdition,2013.
2. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi,38th Reprint, 2022.
3. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson, 9thEdition, Reprint, 2002.
4. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
5. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

2600BS02: ENGINEERING CHEMISTRY

B. TECH I YEAR I SEMESTER (CSE & ECE)	L T P C
B. TECH I YEAR II SEMESTER (AIML, DS, CS, IT & EEE)	3 0 0 3

Course Objectives:

1. To acquire the knowledge of water treatment which are essential for the Engineers and in industry.
2. To include the importance of fundamental aspects of electrochemistry, significance of corrosion and it's control to protect the structures.
3. To impart foundational knowledge of various energy sources and their practical applications in engineering.
4. To account the applications of different kinds of polymers in various fields.
5. To equip students with an understanding of smart materials, biosensors, and analytical techniques applicable in engineering, industrial, environmental, and biomedical fields.

Course Outcomes:

1. Students will be able to understand the fundamental properties of water and its applications in both domestic and industrial purposes.
2. Students will gain basic knowledge of electrochemical processes to understand the functioning of electrodes.
3. Students will be able to assess the underlying causes and consequences of corrosion, distinguishing between various corrosion types and evaluate advanced corrosion control strategies.
4. Students will comprehend the significance and practical applications of batteries and various energy sources, enhancing their potential as future engineers and entrepreneurs.
5. Students will learn the basic concepts and properties of polymers and other engineering materials.
6. Students will be able to apply the principles of UV-Visible, IR spectroscopy and Raman spectroscopy in analyzing pollutants in dye industries and biomedical applications.

UNIT-I: Water and its treatment: [8]

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 breakpoint chlorination. Defluoridation - Nalgonda technique.

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

UNIT-II: Electrochemistry and Corrosion: [8]

Introduction- Electrode potential, standard electrode potential, Nernst equation (no derivation), electrochemical cell - Galvanic cell, cell representation, 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, **Glass electrode-Construction, working and determination of pH of unknown solution using glass 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: Energy sources: [8]

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, **Solid fuels-classification, analysis of coal-proximate and ultimate analysis and their significance.** Fossil fuels: Introduction, Classification, Petroleum - Refining of Crude oil, Cracking - Types of cracking - Moving bed catalytic cracking. LPG and CNG composition and uses.

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

UNIT - IV: Polymers: [8]

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 themoplastics and thermo setting plastics, Fiber reinforced plastics (FRP). **Conducting polymers:** Definition and Classification with examples - Mechanism of conduction in transpoly-acetylene and applications of conducting polymers. Biodegradable polymers: Polylactic acid and its applications.

UNIT-V- Advanced Functional Materials: [8]

UNIT-V- Engineering Applications of Chemistry: [8]

Smart materials: Introduction, Classification with examples - Shape Memory Alloys – Nitinol,

Piezoelectric materials – quartz and their engineering applications.

Biosensor - Definition, Amperometric Glucose monitor sensor.

Interpretative spectroscopic applications 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.

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.

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

2605ES03: DATA STRUCTURES

B.Tech. I Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites: A course on "Programming for Problem Solving

Course Objectives

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

Course Outcomes

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.
- Analyze and implement advanced data structures, searching, sorting, hashing, and graph algorithms to solve computational problems efficiently.
- Evaluate the performance of different data structures and algorithms using complexity analysis, and develop optimized solutions for real-world applications involving file organization and data management.

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.

UNIT-iII

Multi way Search Trees: Introduction, B Trees, B Trees ADT, 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

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 File Operations, File Organization, Indexing.

TEXTBOOKS:

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

REFERENCE:

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

2605ES02: **PYTHON PROGRAMMING**

B.Tech. I Year I Sem.

L	T	P	C
3	***0***	***0***	***3***

Prerequisites: Basic knowledge of computer fundamentals, C programming.

Course Objectives:

Introduce the fundamentals of Python programming for problem-solving.

1. Develop skills to write structured, modular, and efficient Python code.
2. Enable students to use Python's built-in data structures and libraries effectively.
3. Provide knowledge on file handling, exception handling, and object-oriented programming in Python.
4. Equip students with the ability to apply Python for real-world applications including data processing and automation.

Course Outcomes:

1. Write Python programs using variables, operators, expressions, and control structures.
2. Implement Python programs using built-in data structures like lists, tuples, sets, and dictionaries.
3. Apply modular and object-oriented programming principles in Python.
4. Handle files, exceptions, and apply Python libraries for problem-solving.
5. Develop small-scale applications in Python for automation and data manipulation.
6. Demonstrate problem-solving skills by designing, implementing, testing, and debugging Python programs for real-world applications.

UNIT-1 - Introduction to Python and Basics of Programming

Introduction to Python: Features, Applications, Installation, IDEs, Python Syntax, Indentation, Comments, Variables, Data Types, Type Casting, Operators: Arithmetic, Relational, Logical, Assignment, Membership, Identity, Bitwise, Input/Output functions (input(), print()), Control Structures: if, if-else, if-elif-else, Nested Conditions, Looping: for, while, Nested Loops, break, continue, pass.

UNIT-2 - Data Structures in Python

Strings: Creation, Indexing, Slicing, Methods, String Formatting, Lists: Creation, Indexing, Slicing, List Comprehension, Methods, Tuples: Properties, Indexing, Methods, Sets: Creation, Operations, Methods, Dictionaries: Creation, Access, Methods, Dictionary Comprehension, Iterating over data structures.

UNIT-3 - Functions and Modules

Functions: Defining, Calling, Parameters, Return Values, Types of Arguments: Positional, Keyword, Default, Variable Length, Scope of Variables: Local and Global, Lambda Functions, Map, Filter, Reduce, Recursion in Python, Modules: Importing, Creating User-defined Modules, Standard Modules (math, random, datetime), Packages in Python.

UNIT-4 - File Handling and Exception Handling

File Handling: Opening, Reading, Writing, Appending, File Modes, File Methods, Working with CSV and JSON Files, Exception Handling: try, except, else, finally, Built-in Exceptions, Raising Exceptions.

UNIT-5 - Object-Oriented Programming and Applications

OOP Basics: Classes, Objects, Attributes, Methods, Constructor (_init_), self-keyword, Inheritance: Single, Multiple, Multilevel, Hierarchical, Method Overriding, Method Overloading (conceptual), Encapsulation and Polymorphism, Application Development: Data Processing Script, Basic Calculator.

TEXTBOOKS:

1. Python Programming: Using Problem Solving Approach by Reema Thareja.
2. Python Crash Course by Eric Matthes, Leaming Python by Mark Lutz.

REFERENCEBOOKS:

1. Introduction to Python Programming by Gowrishankar S., Veena A.
2. Python Cookbook by David Beazley and Brian K. Jones.
3. Fluent Python by Luciano Ramalho, Automate the Boring Stuff with Python by Al Sweigart.

(2600HS01): ENGLISH FOR SKILL ENHANCEMENT

B.Tech. I Year II Sem. ***L TP C***

3 0 0 3

INTRODUCTION

National Education Policy-2020 aims at preparing students with knowledge, skills, values, leadership qualities and initiates them for lifelong learning. It also emphasizes language study and promotion of languages through understanding and proper interpretation. English language is central to the educational eco system. The importance oflanguage as medium ofcommunication for personal, social, official and professional needs to be emphasized for clear and concise expression. Teaching and learning of receptive and productive skills viz., Listening, Speaking, Reading and Writing (LSRW) are to be taught and learnt effectively in the undergraduate Engineering programs. Learners should be encouraged to engage in a rigorous process oflearning to become proficient users of English language by adopting a deeply focused and yet flexible approach as opposed to rote learning.

In this connection, suitable syllabus, effective pedagogy, continuous assessments and students' involvement result in productive learning. This course supports the latest knowledge and skill requirements and shall meet specified learning outcomes. The main objectives of English language teaching and learning as medium ofcommunication and for promotion ofcultural values are embedded in this syllabus. Efforts are being made in providing a holistic approach towards value-based language learning which equips the learner with receptive as well as productive skills.

The focus in this syllabus is on skill development, fostering ideas and practice of language skills in various contexts and cultures in the areas of vocabulary, grammar, reading and writing. For this, the teachers should use the prescribed textbook for detailed study. The students should be encouraged to read the texts leading to reading comprehension. The time should be utilized for working out the exercises given after each excerpt, and also for supplementing the exercises with authentic materials of a similar kind, for example, newspaper articles, advertisements, promotional material.

LEARNING OBJECTIVES: This course will enable the students to:

a. Improve their vocabulary.
b. Use appropriate sentence structures in their oral and written communication.
c. Develop their reading and study skills.
d. Equip students to write paragraphs, essays, precis and draft letters.
e. Acquire skills for Technical report writing.

COURSE OUTCOMES: Students will be able to:

a. Choose appropriate vocabulary in their oral and written communication.
b. Demonstrate their understanding ofthe rules offunctional grammar and sentence structures.
c. Develop comprehension skills from known and unknown passages.
d. Write paragraphs, essays, precis and draft letters.
e. Write abstracts and reports in various contexts.

UNIT-I

Theme: ***Perspectives***

Lesson on 'The Generation Gap' by Benjamin M. Spock from the prescribed textbook titled English for the Young in the Digital World published by Orient Black SwanPvt. Ltd

and Prepositions- Degrees of Comparison

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

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

UNIT-II

Theme: **Digital Transformation**

Lesson on *'Emerging Technologies'* from the prescribed 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-ID

Theme: **Attitude and Gratitude**

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

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

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

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

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

UNIT-IV

Theme: **Entrepreneurship**

Lesson on *'Why a Start-Up Needs to Find its Customers First'* by Pranav Jain from the prescribed 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 Cliches 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-Precis Writing.

UNIT-V

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 ofReports Formats-Structure ofReports (Manuscript Format) -Types of Reports - Writing a Technical Report***

Prescribed Textbook:

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

REFERENCE BOOKS:

1. Swan, Michael. (2016). *Practical English Usage.* Oxford University Press. New Edition.
2. *Kara!, Rajeevan. 2023. English Grammar Justfor 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). *Englishfor Technical Communicationfor Engineering Students.* Mc Graw-Hill Education India Pvt. Ltd.

2600BS61: ENGINEERING CHEMISTRY LAB

B. TECH I YEAR I SEMESTER (CSE & ECE)	L T P C
B. TECH I YEAR II SEMESTER (AIML, DS, CS, IT & EEE)	0 0 2 1

Course Objectives:

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.

Course Outcomes:

1. Students will develop practical skills through hands-on chemistry experiments relevant to engineering.
2. Students will learn to determine important parameters such as water hardness and the corrosion rate of mild steel under various conditions.
3. Students will be able to apply techniques like conductometry, potentiometry, and pH metry to determine concentrations or equivalence points in acid-base reactions.
4. Students will gain experience in synthesizing polymers such as Bakelite and Nylon-6,6.
5. Students will understand the working principle of colorimetry and the relationship between absorbance and concentration (Beer-Lambert Law).
6. Students will gain the skills in resolving the societal issues and are able to understand the real-world scenario.

List of Experiments:

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

II. Conductometry: Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.

III. Potentiometry: Estimation of concentration of Fe^{+2}ion by Potentiometry using $K_2Cr_2O_7$.

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

V. Colorimetry: Verification of Lambert-Beer's law using KMnO4.

VI. Preparations:

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

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

VIII. Lubricants:

1. Determination of surface tension of a given liquid using stalagmometer.

2. Determination of viscosity of lubricant oil by using Ostwald's viscometer.

IX. 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.

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.
3. Inorganic Quantitative analysis by A.I. Vogel, ELBS Publications.
4. College Practical Chemistry by V.K. Ahluwalia, Narosa Publications Ltd. New Delhi (2007).

2605ES63: DATA STRUCTURES LAB

B.Tech. I Year II Sem.

L	T	P	C
0	0	2	1

Prerequisites: 1. A Course on "Programming for problem solving".

Course Objectives:

1. It covers various concepts of C programming language
2. It introduces searching and sorting algorithms
3. It provides an understanding of data structures such as stacks and queues.

Course Outcomes:

1. Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
2. Ability to Implement searching and sorting algorithms.
3. Implement and analyze linear and non-linear data structures, including linked lists, stacks, queues, trees, graphs, and hash tables using C programming.
4. Apply tree traversal, graph traversal, and hashing techniques to efficiently organize, search, and retrieve data.
5. Design and implement efficient solutions using advanced data structures such as Binary Search Trees, AVL Trees, B-Trees, and B+ Trees for real-world applications.
6. Evaluate the performance of different data structures and algorithms based on time and space complexity to develop optimized software solutions.

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) Heap sort, ii) Merge Sort, iii) Quick 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

9. Write a program to implement the graph traversal methods.

   Write a program to implement the following Hash Functions: i) Division

10. 10. Method, ii) Multiplication Method, iii) Mid-square Method, iv) Folding Method

TEXTBOOKS:

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

REFERENCE BOOK:

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

(2600HS61): ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB

B.Tech. I Year II Sem.

L	T	p	C
0	0	2	1

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.

Course Objectives:

- To facilitate computer-assisted multi-media instruction enabling individualized and independent language learning
- To sensitize students to the nuances of English speech sounds, word accent, intonation and rhythm
- To bring about a consistent accent and intelligibility in students' pronunciation of English by providing an opportunity for practice in speaking
- To improve the fluency of students in spoken English and neutralize their mother tongue influence
- To train students to use language appropriately for public speaking and interviews
- To foster better understanding of nuances of English language through audio- visual experience and group activities
- To inculcate Neutralization of accent for intelligibility
- To enhance students' speaking skills with clarity and confidence which in turn enhances their employability skills

Learning Outcomes: Students will be able to:

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

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

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

Listening Skills:

Objectives

1. To enable students develop their active listening skills
2. To equip students with necessary training in listening, so that they can comprehend the speech of people from different linguistic backgrounds

Speaking Skills:

3. To improve their pronunciation and neutralize accent
4. To enable students express themselves fluently and appropriately
5. To practice speaking in social and professional contexts

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- ActiveListening

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

❖ **Diagnostic Test-Activit y 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.)*

ICSLab:

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- TakingLeave - 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*

ICSLab:

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 IHandouts 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.)* **ICSLab:**

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.)

ICSLab:

Instruction: Understanding Non-Verbal Communication

Practice: Silent Speech - Dumb Charades Activity

Minimum Requirement of infrastructural facilities for ELCSLab:

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 Requirement (Hardware component):

Computer network with LAN facility (minimum 40 systems with multimedia) with the following specifications:

i) Computers with Suitable Configuration

ii) High Fidelity Headphones

with a Public Address System, a T. V. or LCD, a digital stereo - audio & video system and camcorder etc.

Suggested Software:

- Cambridge Advanced Learners' English Dictionary with CD.
- Grammar Made Easy by Darling Kindersley.
- Punctuation Made Easy by Darling Kindersley.
- Oxford Advanced Learner's Compass, 10th Edition.
- English in Mind (Series 1-4), Herbert Puchta and JeffStranks with Meredith Levy, Cambridge.
- English Pronunciation in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- English Vocabulary in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS).

REFERENCE BOOKS:

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

2605ES62: PYTHON PROGRAMMING LAB

B.Tech. I Year I Sem.

Course Objectives:

L	T	P	C
0	*0*	*2*	*1*

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

Course Outcomes:

After completion of the course, the student should be able to

- Develop the application specific codes using python.
- Understand Strings, Lists, Tuples and Dictionaries in Python
- Verify programs using modular approach, file 1/0, Python standard library
- Implement Digital Systems using Python
- Apply object-oriented programming, exception handling, and external Python libraries to develop robust and reusable applications.
- Design, test, debug, and develop GUI-based and real-world Python applications by applying problem-solving techniques and software development practices.

Note: The lab experiments will be like the following experiment examples.

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 ofa person through the keyboard and print the details.
5. Print the below triangle using for loop. 5

333
2222
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

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 *file]* 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 NumPy package 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.

TEXTBOOKS:

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

REFERENCE BOOKS:

1. Python Programming: A Modem 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, AUser'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

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC01) DISCRETE MATHEMATICS

B.Tech. II Year I Sem **L T P C**
3 0 0 3

Course Objectives:

1. Use mathematically correct terminology and notation.
2. Construct correct direct and indirect proofs.
3. Use division into cases in a proof.
4. Use counter examples.
5. Apply logical reasoning to solve a variety of problems.

Course Outcomes:

At the end of the course the students can:

1. For a given logic sentence express it in terms of predicates, quantifiers, and logical connectives.
2. For a given a problem, derive the solution using deductive logic and prove the solution based on logical inference.
3. For a given a mathematical problem, classify its algebraic structure Evaluate Boolean functions and simplify expressions using the properties of Boolean algebra.
4. Develop the given problem as graph networks and solve with techniques of graph theory.
5. Analyse elementary combinatorics, including basic counting techniques, the inclusion-exclusion principle, and permutations and combinations, both with and without repetition, to solve combinatorial problems.
6. Understand and apply graph theory concepts such as connectivity, paths, cycles, sub graphs, isomorphism, Eulerian and Hamiltonian walks, graph coloring, and tree structures to analyze and solve real-world problems, including shortest path determination and articulation points.

UNIT-I

Propositional Logic: Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, Normal Forms, Disjunctive and Conjunctive Normal Form, The use of Quantifiers.

UNIT-II

Sets, Relation, and Function: Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets. Mathematical Induction: The Well-Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers,

UNIT-III

Algebraic Structures and Morphism: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Abelian Group, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings,Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function.

UNIT-IV

Elementary Combinatorics: Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutations, and combinations With Repetition and Without Repetition.

UNIT-V

Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph, definition properties and Example, rooted trees, trees and sorting, weighted trees and prefix codes, Bi- connected component and Articulation Points, Shortest distances.

TEXT BOOKS:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw–Hill
2. Susanna S. Epp, Discrete Mathematics with Applications,4th edition, WadsworthPublishing Co. Inc.
3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw–Hill.

REFERENCE BOOKS:

1. J.P. Tremblay and R. Manohar, "Discrete Mathematical Structure and It's Application to Computer Science", TMG Edition, Tata McGraw-Hill.
2. Norman L. Biggs, Discrete Mathematics, 2nd Edition, Oxford University Press. Schaum's Outlines Series.
3. Seymour Lipschutz, Marc Lipson, Discrete Mathematics, Tata McGraw–Hill

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC03) OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B.Tech. II Year I Sem **L T P C**

3 0 0 3

Course Objectives

1. The objective of this course is to provide object-oriented concepts through which robust, secure, and reusable software can be developed.
2. To understand object-oriented principles like abstraction, encapsulation, inheritance, and polymorphism and apply them in solving problems.
3. To understand the principles of inheritance and polymorphism and demonstrate how they relate to the design of abstract classes.
4. To understand the implementation of packages and interfaces.
5. To understand the concepts of exception handling, multithreading, and collection classes.
6. To understand the design of Graphical User Interface using applets and Swing controls.

Course Outcomes

1. Explain object-oriented principles and implement them using Java.
2. Apply inheritance, polymorphism, and interfaces in Java programs.
3. Handle exceptions and develop multithreaded applications.
4. Utilize Java collections, files, and built-in utility classes.
5. Develop simple applets and Swing-based GUI applications.
6. Implement event handling for interactive Java programs

UNIT I

Object-oriented thinking – A way of viewing the world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies – Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. An Overview of Java – History of Java, comments, Data types, Variables, Constants, Scope and Lifetime of variables, Operators, Type conversion and casting, Enumeration, Control flow – block scope, conditional statements, loops, break and continue statements, simple Java standalone programs, arrays, console input and output, classes, methods, constructors, static, this keyword, recursion, exploring string classes and garbage collection.

UNIT II

Inheritance – Inheritance hierarchy, super keyword, preventing inheritance: final classes and methods, the Object class and its methods. Polymorphism – dynamic binding, Constructor and method overloading, method overriding, abstract classes. Interfaces – Interfaces vs Abstract Classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interfaces, inner classes. Packages – Defining, creating and accessing a package, CLASSPATH, Access modifiers, importing packages.

UNIT III

Exception Handling – Dealing with errors, benefits of exception handling, the classification of exceptions – exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, creating own exception subclasses. Multithreading – Differences between multiple processes and multiple threads, thread lifecycle, creating threads, interrupting threads, thread priorities, synchronizing threads, inter-thread communication – producer-consumer problem.

UNIT IV

Collection Framework in Java – Introduction to Java collections, Overview of Java collection framework, Commonly used collection classes – ArrayList, LinkedList, HashSet, TreeSet, Map –

HashMap, TreeMap, Legacy Classes – Vector, Stack, Hashtable. Other Utilities – Scanner, StringTokenizer, Random, Date. Files – Streams – Byte Streams, Character Streams, Text input/output, Binary input/output, File Management using File class.

UNIT V

Applets – Inheritance hierarchy for applets, differences between applets and applications, Lifecycle of an applet and Passing parameters to applets. GUI Programming – Swing – The AWT class hierarchy, Introduction to Swing, Swing vs AWT, Hierarchy for Swing components, Overview of Swing components – JButton, JLabel, JTextField, JCheckBox, RadioButton, JTextArea, etc., simple Swing applications, Layout managers – FlowLayout, BorderLayout, GridLayout, and GridBagLayout. Event Handling – Events, Event sources, Event classes, Event Listeners, Delegation event model, Handling Mouse and Key events, Adapter classes.

Textbooks

1. *Java Fundamentals – A Comprehensive Introduction*, Herbert Schildt and Dale Skrien, TMH
2. *Understanding Object-Oriented Programming with Java*, Updated Edition, T. Budd, Pearson Education

Reference Books

1. *Java for Programmers*, P.J. Deitel and H.M. Deitel, PEA (or) *Java: How to Program*, P.J. Deitel and H.M. Deitel, PHI
2. *Object-Oriented Programming through Java*, P. Radha Krishna, Universities Press
3. *Thinking in Java*, Bruce Eckel, PE
4. *Programming in Java*, S. Malhotra and S. Choudhary, Oxford University Press
5. *Design Patterns*, Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2662PC01) MATHEMATICAL FOUNDATIONS OF CRYPTOGRAPHY

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Course Objectives

1. Build a solid mathematical basis to understand foundations of cryptography
2. Formally understand the notions related to security authentication and privacy.
3. Provide a rigorous treatment of the emerging and key subject subarea of CSE - security.

Course Outcomes

1. Students will gain an understanding of cryptosystems widely used to protect data security on the internet, and be able to apply the ideas in new situations as needed.

UNIT - I

Basic functions of cryptography - Encryption Schemes, Digital Signatures, Fault Tolerant Protocols and Zero-Knowledge Proofs.

The Computational Model: P, NP, and NP- Completeness, Probabilistic Polynomial Time, Non-Uniform Polynomial Time

UNIT - II

Computational Difficulty

One-Way Functions Definitions, Strong One- Way Functions, Weak One-Way Functions, Universal One-Way Function, Trapdoor One-Way Permutations Computational Indistinguishability: Definition, Relation to Statistical Closeness, Indistinguishability by Repeated Experiments, Indistinguishability by Circuits

UNIT - III

Zero-Knowledge Proof Systems

Zero-Knowledge Proofs, Perfect and Computational Zero-Knowledge, An Example (Graph Isomorphism in PZK) Zero-Knowledge with Respect to Auxiliary Inputs

UNIT - IV

Encryption Schemes

Private-Key versus Public-Key Schemes, The Syntax of Encryption Schemes, Semantic Security, Indistinguishability of Encryptions, Stream--Ciphers, Preliminaries: Block--Ciphers

UNIT - V

Digital Signatures and Message Authentication: Attacks and security, Variants

Constructions of Message Authentication Schemes: Applying a pseudorandom function to the document

TEXTBOOK:

1. Foundations of Cryptography (two volumes), Oded Goldreich, Cambridge university Press, 2004. (Indian print available).

REFERENCES:

1. Introduction to Modern Cryptography, J. Katz, Y. Lindell, Chapman Hall, USA 2007.
2. Modern cryptography - Theory and practice, Wen Bo Mao, Prentice Hall, USA, 2003 (Indian edition available)

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC05) DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year I Sem **L T P C**
3 0 0 3

Course Objectives:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

Course Outcomes:

- Demonstrate the basic elements of a relational database management system and Ability to identify the data models for relevant problems.
- Ability to design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data.
- Analyze the architecture and purpose of database systems, including database languages and data retrieval methods, to understand their role in data management and application domains.
- Design entity-relationship (ER) models and develop conceptual schemas for database systems, effectively representing entities, relationships, and attributes in complex, large-scale enterprise databases.
- Construct relational database schemas by transforming ER models into the relational model, enforcing integrity constraints, and creating views to ensure logical data consistency and accessibility.
- Develop complex SQL queries, including nested queries and triggers, and apply normalization techniques to refine schema design, ensuring data integrity and minimizing redundancy in database applications.

UNIT – I:

Introduction: Database System Applications, Purpose of Database Systems, View of Data, Database Languages – DDL, DML, Relational Databases, Database Design, Database Architecture, Data Mining and Information Retrieval, Database Users and Administrators, History of Database Systems.

Introduction to Data base design: Database Design and ER diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises.

UNIT – II:

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational, Introduction to Views, Destroying /Altering Tables and Views.

Relational Algebra and Calculus: Preliminaries, Relational Algebra, Relational calculus– Tuple relational Calculus, Domain relational calculus.

UNIT – III:

SQL: Queries, Constraints, Triggers: Form of Basic SQL Query, UNION, INTERSECT, and EXCEPT, Nested Queries, Aggregate Operators, NULL values, Natural JOINS, Complex Integrity Constraints in SQL, Triggers and Active Data bases..

Schema Refinement and Normal Forms: Introduction to Schema Refinement, Functional Dependencies - Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization, Schema Refinement in Database Design, Other Kinds of Dependencies.

UNIT – IV:

Transaction Management: Transactions, Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability.

UNIT V:

Concurrency Control: Lock – Based Protocols, Multiple Granularity, Timestamp - Based Protocols, Validation - Based Protocols.

Recovery System-Failure Classification, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of nonvolatile storage, Remote Backup systems. **Storage and Indexing:** Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing, Index Data Structures, Comparison of File Organizations. Tree-Structured Indexing: Intuition for tree Indexes, Indexed Sequential Access Method (ISAM), **B+ Trees**: A Dynamic Index Structure, Search, Insert, Delete.

TEXTBOOKS:

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGrawHill Education (India) Private Limited, 3rd Edition. (Part of UNIT-I, UNIT-II,UNIT-III, UNIT- V)
2. Data base System Concepts, A. Silberschatz, Henry. F. Korth, S. Sudarshan, Mc Graw Hill Education(India) Private Limited l, 6th edition.(Part of UNIT-I,UNIT-IV)

REFERENCE BOOKS:

1. Database Systems, 6th edition, R Elmasri, Shamkant B.Navathe, Pearson Education.
2. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning.
3. Introduction to Database Management, M. L. Gillenson and others, Wiley Student Edition.
4. Database Development and Management, Lee Chao, Auerbach publications, Taylor& Francis Group.
5. Introduction to Database Systems, C.J.Date, Pearson Education

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN
(2605PC06) SOFTWARE ENGINEERING

B.Tech. II Year I Sem **L T P C**

3 0 0 3

Course Objectives: The students will be able

:

- To comprehend the various software process models.
- To understand the types of software requirements and SRS document.
- To know the different software design and architectural styles.
- To learn the software testing approaches and metrics used in software development.
- To know about quality control and risk management.

Course Outcomes: Students will have the ability:

- To compare and select a process model for a business system.
- To identify and specify the requirements for the development of an application.
- To develop and maintain efficient, reliable and cost effective software solutions.
- To critically think and evaluate assumptions and arguments
- Identify and apply the Framework activities for a Given Project.
- Illustrate a Process Model to apply for given Project requirements.
- Implement and Solve Various System Models for a Given Scenario.
- Analyze Various Testing techniques for a Given Project.

UNIT - I:

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths. A Generic view of process: Software engineering- A layered technology, a process framework, Process patterns, process assessment. Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process, Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools

UNIT - II:

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management. System models: Context Models, Behavioral models, Data models, Object models, structured methods. UML Diagrams.

UNIT - III:

Design Engineering: Design process and Design quality, Design concepts, the design model. Creating an architectural design: Software architecture, Data design, Architectural styles and patterns, Architectural Design. Object-Oriented Design: Objects and object classes, An Object-Oriented design process, Design evolution. Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation**.**

UNIT - IV:

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black- Box and White-Box testing, Validation testing, System testing, the art of Debugging. Product metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Metrics for Process and Products: Software Measurement, Metrics for software quality.

UNIT-V: Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan. Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, The Capability Maturity Model Integration (CMMI), Software reliability, The ISO 9000 quality standards.

TEXT BOOKS :

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

REFERENCE BOOKS :

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
2. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering1: Abstraction and modelling, Diner Bjorner, Springer International edition, 2006.
5. Software Engineering2: Specification of systems and languages, Diner Bjorner, Springer International edition 2006.
6. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.
7. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjorner, Springer International Edition.
8. Introduction to Software Engineering, R. J. Leach, CRC Press. Course.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600IN01) INNOVATION AND ENTREPRENEURSHIP

B.Tech. II Year I Sem.

L	T	P	C
2	0	0	2

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

Course Outcomes:

1. Understand the entrepreneurship and the entrepreneurial process and its significance in economic development.
2. Assess the problem from an industry perspective and generate solutions using the design thinking principles.
3. Assess market competition, estimate market size, and develop a prototype.
4. Analyze Business and financial planning models and Go-to-Market strategies.
5. Able to build a start-up, register IP and identify funding opportunities.

Unit I: Fundamentals of Innovation and Entrepreneurship

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

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

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

Unit II: Problem and Customer Identification

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

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

Unit III: Opportunity assessment and Prototype development

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

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

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

Unit IV: Business & Financial Models

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

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

Unit V: Startups and IPR

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

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

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

Suggested Readings:

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.
4. D.F. Kuratko and T.V. Rao, Entrepreneurship: A South-Asian Perspective, Cengage Learning, 2013.
5. Robert D. Hisrich, Michael P. Peters, Dean A. Shepherd, Sabyasachi Sinha (2020). Entrepreneurship, McGrawHill, 11th Edition.
6. NISP -Brochure inside pages - startup_policy_2019.pdf

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC61) OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

B.Tech. II Year I Sem **L T P C**

0 0 2 1

Course Objective:

1. To develop practical proficiency in solving computational problems using fundamental Java programming constructs, recursion, and control structures.
2. To strengthen students' abilities in designing object-oriented programs employing concepts like overloading, abstraction, inheritance, interfaces, and inner classes.
3. To enable learners to apply robust exception handling techniques, including user-defined exceptions and multiple catch mechanisms, for fault-tolerant code.
4. To build skills for developing concurrent applications using multithreading concepts, synchronization, and inter-thread communication.
5. To provide hands-on experience with Java Collections, file management, and directory operations to manipulate and manage data efficiently.
6. To cultivate competence in designing interactive GUI-based applications, applets, and implementing event-driven programming using AWT and Swing components.

Course Outcomes:

1. Apply iterative and recursive programming constructs to develop solutions for mathematical and logical problems using Java.
2. Demonstrate object-oriented programming principles by implementing method overloading, constructor overloading, abstract classes, interfaces, and inner classes.
3. Design user-defined packages and employ robust exception handling strategies, including custom exceptions and multi-catch blocks.
4. Construct concurrent Java applications by creating and managing multiple threads, synchronizing processes, and implementing producer-consumer solutions.
5. Execute file operations, directory traversal, and utilize Java Collection Framework classes such as ArrayList and HashTable for efficient data processing.
6. Develop interactive applets, handle user inputs and events, and build GUI-based applications using layout managers and Swing/ AWT components.

Week 1

a) Java program to find the Fibonacci series using recursive and non-recursive functions.

b) Java program to multiply two matrices.

Week 2

a) Java program demonstrating method overloading and constructor overloading.

b) Java program to display employee details using Scanner class.

c) Java program to check whether a given string is a palindrome.

Week 3

a) Java program to demonstrate abstract class usage.

b) Java program to implement interface using extends keyword (likely intended to use implements for interfaces).

Week 4

a) Java program to create a user-defined package.

Week 5

a) Java program to create inner classes.

b) Java program demonstrating multiple catch blocks.

c) Java program for creating a user-defined exception.

Week 6

a) Java program for producer-consumer problem using threads.

b) Java program that implements a multi-threaded application with three threads.

Week 7

a) Java program to implement file operations (read/write).

b) Java program to list all files in a directory, including its subdirectories.

Week 8

a) Java program using the ArrayList class.

b) Java program to load phone numbers and names from a text file using Hashtable.

Week 9

a) Applet program to display a simple message.

b) Applet program to compute factorial value.

c) Program for passing parameters to an applet.

Week 10

Java program for handling mouse events and key events.

Week 11

Java program for a simple calculator using GridLayout, buttons for digits and operations (+, -, *, %, etc.), and a text field for displaying results.

Textbooks / Reference Books

1. Java Fundamentals – A Comprehensive Introduction, Herbert Schildt and Dale Skrien, TMH.
2. Java for Programmers, P.J. Deitel and H.M. Deitel, PEA (or) Java: How to Program, P.J. Deitel and H.M. Deitel, Pearson.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2662PC61) MATHEMATICAL FOUNDATIONS OF CRYPTOGRAPHY LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	2	1

1. Implement Basic Encryption/Decryption Schemes

- Implement Caesar cipher, Vigenère cipher, and XOR cipher.
- Test encryption and decryption on text input.

2. Demonstrate Digital Signature Generation and Verification

- Use Python libraries (e.g., cryptography or pycryptodome) to sign a message with a private key and verify it using a public key.

3. Simulate a Fault-Tolerant Protocol

- Simulate a basic consensus protocol (like 2-phase commit) handling failure scenarios.

4. Implement Zero-Knowledge Proof (ZKP) Concept

- Simulate a ZKP for Graph Isomorphism using a challenge-response protocol.

5. Implement a Simple One-Way Hash Function

- Build a toy hash function and demonstrate its one-way property. Compare it to SHA-256.

6. Test for Strong and Weak One-Way Functions

- Write a program to simulate strong vs weak one-way functions. Use time complexity analysis to show difficulty of inversion.

7. Demonstrate Trapdoor One-Way Permutations

- Use RSA key generation to demonstrate how trapdoor functions work in public-key encryption.

8. Demonstrate Computational Indistinguishability

- Create two similar distributions and test whether a computational process can distinguish between them (simulate using coin tosses).

9. Implement a Zero-Knowledge Proof for Discrete Logarithms

- Given a number $y = g^x \mod p$, simulate ZKP to prove knowledge of xxx without revealing it.

10. Implement Perfect and Computational ZKP

- Demonstrate perfect ZKP using small problems like Sudoku validation without revealing the solution.

11. Auxiliary Inputs in ZKP

- Simulate a scenario where an adversary has auxiliary information and test how ZKP

maintains secrecy.

12. Compare Private-Key vs Public-Key Encryption

- Implement AES (symmetric) and RSA (asymmetric) and compare their usage on messages.

13. Demonstrate Block Cipher Modes

- Implement ECB, CBC, and CFB modes for AES and show their impact on ciphertext for patterned plaintext.

14. Implement a Digital Signature Scheme Using RSA

- Generate keys, sign a message, and verify using the RSA algorithm.

15. Construct a Message Authentication Code (MAC)

- Use a pseudorandom function like HMAC-SHA256 to generate and verify MACs for file integrity.

TEXT BOOK:

1. Foundations of Cryptography (two volumes), Oded Goldreich, Cambridge university Press, 2004. (Indian print available).

REFERENCE BOOKS:

1. Introduction to Modern Cryptography, J. Katz, Y. Lindell, Chapman Hall, USA 2007.
2. Modern cryptography - Theory and practice, Wen Bo Mao, Prentice Hall, USA, 2003 (Indian edition available)

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC63) DATABASE MANAGEMENT SYSTEMS LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	2	1

Course Objectives:

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Course Outcomes:

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

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

TEXT BOOKS:

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

REFERENCES 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.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2662SD01) DATA VISUALIZATION - R PROGRAMMING/ POWER BI/TABLEAU/GOOGLE CHART

B.Tech. II Year I Sem.

L	T	P	C
0	0	2	1

Course Objectives:

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.

Course Outcomes:

1. Understand how to import data into Tableau.
2. Understand concepts of Dimensions and Measures.
3. Develop Programs and understand how to map Visual Layouts and Graphical Properties
4. Create Dashboard, custom charts, and, publish to tableau online for any real time dataset

List of Programs

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.

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.

WEB RESOURCE:

1. https://developers.google.com/chart/interactive/docs.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600MC01) INDIAN KNOWLEDGE SYSTEM

B.Tech. II Year I Sem. **L T P C**

0 0 2 1

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.

Course Objectives: This course aims:

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

Course Outcomes: Students will be able to:

1. Understand nature, scope and related fields of Indian knowledge system.
2. Demonstrate the scientific literature available in ancient Indian traditions
3. Understanding the application of Bharatiya Jnana Parampara
4. Understand Indian approach towards Wellbeing
5. Appreciate vast contribution of ancient Indian researchers, engineers, scientists and architects to the modern world

Unit 1: **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 2: 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 3: 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 4: Introduction to Ancient Indian Wellness Systems

Concept of Wellness – Yoga System - Ayurveda System - Ancient Indian Aesthetics

Unit 5: 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

- **Pedagogy for Teachers: Apart from Class Room Instruction, the following Methods are Suggested.**

1. Project based activities and learning.
2. Presentation and case studies.
3. Film screening and book reviews.
4. Visit to historical places, archives centre, research centre or library nearby.

***Note**: Activities mentioned above are only suggestive. Teacher-educators should encourage students to be innovative.*

Suggested Readings:

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

Video Resources:

1. Introductory lectures by Prof. Gauri Mahulikar
2. Introductory lectures by Prof. Kapil Kapoor

Websites:

- https://iksin dia.org/index.php
- Official Website of IKS- Indian Knowledge System
- https://www.youtube.com/watch?v=uKcf-hSlcUE
- Address by Prof Kapil Kapoor | Indian Institute of Advanced Study (FDP 2021)
- https://www.youtube.com/watch?v=MDJTXNiH2_A
- Mukul Kanitkar on Bharatiya Knowledge System
- https://www.youtube.com/watch?v=uARMhv97pjk
- https://www.youtube.com/watch?v=oTwgf56GbsA
- Scientific History of India | Mukul Kanitkar Lecture in DTU
- https://youtu.be/gNJNmPJqXJc?si=WFBbuUT65mLZzpOW
- Ancient India's Scientific Achievements & Contribution in Mathematics, Astronomy, Science & Medicine

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

2600BS06: Mathematical and Statistical Foundations

B.Tech. II Year II Sem.

L	T	P	C
2	1	0	3

Pre-requisites: Mathematics courses of first year of study.

Objectives: To learn

1. The Number Theory basic concepts useful for cryptography etc.
2. The theory of Probability, and probability distributions of single random variables.
3. The sampling theory and testing of hypothesis and making inferences.
4. The curve fitting, correlation and regression for the given data.

Course outcomes: After learning the contents of this paper, the student must be able to

1. Apply the number theory concepts to cryptography domain.
2. Apply the concepts of probability and distributions to some case studies.
3. Correlate the material of one unit to the material in other units.
4. Resolve the potential misconceptions and hazards in each topic of study.
5. Fit the curve, correlation and regression for the given data.

UNIT-I: Basics of Number Theory **10 L**

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.

UNIT-II: Random Variables and Probability Distributions **8 L**

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: Continuous Distributions and Sampling **10 L**

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

UNIT-IV: Tests of Hypotheses (Large and Small Samples) **10 L**

Statistical Hypotheses: General Concepts – Testing a Statistical Hypothesis. Single sample: Tests concerning a single mean. Two samples: Tests on two mean (Unknown for equal variance). One sample: Test on a single proportion. Two samples: Tests on two proportions. Two- sample tests concerning variances: F-distribution

UNIT-V: Applied Statistics **10 L**

Curve fitting by the method of least squares – Fitting of straight lines – Second degree parabolas and more general curves – Correlation and Regression – Rank correlation.

TEXT BOOKS:

1. Kenneth H. Rosen, Elementary Number Theory & its Applications, sixth edition, Addison Wesley, ISBN 978 0-321-50031-1.
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics

for Engineers & Scientists, 9th Ed. Pearson Publishers.

3. S C Gupta and V K Kapoor, Fundamentals of Mathematical Statistics, Khanna publications.

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. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC13) COMPUTER ORGANIZATION

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Course Outcomes:

1. Demonstrate the ability to represent and manipulate data using binary number systems and apply Boolean algebra to analyze and design digital logic circuits.
2. Simplify Boolean functions using Karnaugh Maps and design minimal logic circuits using universal logic gates.
3. Classify data representation formats and implement arithmetic, logic, and shift microoperations using register transfer language.
4. Illustrate the basic organization of a computer and differentiate between hardwired and microprogrammed control unit designs.
5. Analyze CPU structure, addressing modes, and execute arithmetic operations including fixed-point, floating-point, and BCD formats.

UNIT - I

Binary Systems: Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers, Binary Codes, Binary Storage and Registers, Binary Logic.

Boolean Algebra And Logic Gates: Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, Digital Logic Gates, Integrated Circuits.

UNIT - II

Gate-Level Minimization: The Map method, Four-variable map, Five-variable map, Product of Sum's simplifications, Don't care conditions, NAND and NOR implementation, other two level implementations, Exclusive-OR Function.

UNIT - III

Basic Structure of Computers: Computer Types, Functional unit, Data Representation, Fixed Point Representation, Floating Point Representation, Error Detection codes.

Register Transfer Language and Micro operations: Register Transfer language, Register

Transfer, Bus and memory transfers, Arithmetic Micro operations, Logic micro-operations, Shift micro-operations, Arithmetic logic shift unit.

UNIT - IV

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, Micro program Sequencer, Hard wired control Vs Micro programmed control

UNIT - V

Central Processing Unit Organization: General Register Organization, STACK organization. Instruction formats, Addressing modes. DATA Transfer and manipulation, Program control. Reduced Instruction set computer.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Floating – point Arithmetic, BCD Adder.

TEXT BOOKS:

1. Digital Design: With an Introduction to the Verilog HDL – Fifth Edition, M. Morris Mano, Pearson Education.
2. Fundamentals of Logic Design – Roth, 7th Edition, Thomson.
3. Computer Systems Architecture – M. Moris Mano, 3rd Edition, Pearson/PHI
4. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5th Edition, McGraw Hill.

REFERENCE BOOKS:

1. Digital Principles and Design – Donald D. Givone, Tata Mc Graw Hill.
2. Fundamentals of Digital Logic and Micro Computer Design, 5th Edition, M. Rafiquzzaman, Wiley-Interscience.
3. Computer Organization and Architecture – William Stallings 7^{th} Edition, Pearson/ PHI.
4. Structured Computer Organization – Andrew S. Tanenbaum, 6^{th} Edition PHI/Pearson.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC12) FORMAL LANGUAGES AND AUTOMATA THEORY

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites: Mathematical Foundations

Course Objectives

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.

Course Outcomes

1. Understand the concept of abstract machines and their power to recognize the languages.
2. Employ finite state machines for modeling and solving computing problems.
3. Design context free grammars for formal languages.
4. Distinguish between decidability and undecidability.

UNIT - I

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 €-transitions to NFA without €-transitions. Conversion of NFA to DFA, Moore and Melay machines

UNIT - II

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

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

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

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.

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.

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.
4. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
5. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC07) COMPUTER NETWORKS

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites

1. A course on "Programming for problem solving".
2. A course on "Data Structures".

Course Objectives

1. Equip the students with a general overview of the concepts and fundamentals of computer networks.
2. Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.
3. Elucidate the students about working and implementation of protocols at various layers in protocols stack.
4. Appreciating the protocol working by observing and analysing outputs of the packet sniffer,

Course Outcomes

1. Gain the knowledge of the basic computer network technology.
2. Gain the knowledge of the functions of each layer in the ISO-OSI and TCP/IP reference model.
3. Obtain the skills of subnetting and routing mechanisms.
4. Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.
5. Understanding working of the protocols through traces captured by a packet sniffer

UNIT - I

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

UNIT-II

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

UNIT - III

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

TCP Connection Management, Principles of Congestion Control, TCP Congestion Control, Fairness.

UNIT - IV

Network Layer: Data and Control plane, Forwarding and Routing 308, Network Service Models, Virtual Circuit and Datagram Networks, Router working, The Internet Protocol (IP): Forwarding and Addressing in the Internet, Datagram Format, IPv4 Addressing, Internet Control Message Protocol (ICMP), IPv6, IP Security, Routing Algorithms- The Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet-Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter-AS Routing: BGP, Broadcast and Multicast Routing, Broadcast Routing Algorithms, Multicasting.

UNIT - V

The Link Layer: The Services Provided by the Link Layer, Error-Detection and -Correction Techniques-Parity Checks, Checksum Methods, Cyclic Redundancy Check (CRC), Hamming code, Multiple Access Links and Protocols, Channel Partitioning Protocols, Random Access Protocols, Taking-Turns Protocols, DOCSIS: The Link-Layer Protocol for Cable Internet Access, Switched Local Area Networks, Link-Layer Addressing and ARP, Ethernet, Link-Layer Switches, Virtual Local Area Networks (VLANs), Link Virtualization-Multiprotocol Label Switching (MPLS), Data Center Networking, A Day in the Life of a Web Page Request. Wireless network characteristics, Wireless LAN.

TEXT BOOKS:

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

REFERENCE BOOK:

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

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC04) OPERATING SYSTEMS

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites:

1. A course on "Computer Programming and Data Structures".
2. A course on "Computer Organization and Architecture".

Course Objectives:

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

Course Outcomes:

1. Will be able to control access to a computer and the files that may be shared
2. Demonstrate the knowledge of the components of computers and their respective roles in computing.
3. Ability to recognize and resolve user problems with standard operating environments.
4. Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

UNIT - I

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

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

UNIT - II

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

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

UNIT - III

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors **Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT - IV

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

UNIT - V

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

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.

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.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC65) OPERATING SYSTEMS LAB

B.Tech. II Year II Sem.

L	T	P	C
0	0	2	1

Prerequisites:

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

Co-requisite: A course on "Operating Systems".

Course Objectives:

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

Course Outcomes:

1. Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
2. Able to implement C programs using Unix system calls.

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

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.

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.

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC64) COMPUTATIONAL MATHEMATICS LAB

(Using Python/MATLAB software)

B.Tech. II Year II Sem. **L T P C**

0 0 2 1

Pre-requisites: Matrices, Iterative methods and ordinary differential equations

Course Objectives: 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

Course outcomes: After learning the contents of this paper, the student must be able to

1. Develop the code to find the Eigen values and Eigen Vectors using Python/MATLAB.
2. Develop the code find solution of Algebraic and Transcendental Equations and Linear system of equations using Python/MATLAB
3. Write the code to solve problems of **First-Order ODEs Higher order linear differential equations with** constant coefficients

* **Visualize all solutions Graphically through programmes**

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

Programs:

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

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

Bisection method, Newton Raphson Method

Programs:

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

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

Jacobi's iteration method and Gauss-Seidal iteration method

Programs:

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

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

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

Programs:

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

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

P Programs:

- Solving homogeneous ODEs
- Solving non-homogeneous ODEs

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.

REFERENCE BOOKS:

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

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2605PC65) COMPUTER NETWORKS LAB

B.Tech. II Year II Sem. **L T P C**

0 0 2 1

Course Objectives:

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

Course Outcomes:

- Implement data link layer farming methods
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.
- Implement Encoding and Decoding techniques used in presentation layer
- To be able to work with different network tools

List of Experiments

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijsktra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. **Wireshark**
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
1. How to run Nmap scan
2. Operating System Detection using Nmap
3. Do the following using NS2 Simulator
    I. NS2 Simulator-Introduction
    II. Simulate to Find the Number of Packets Dropped
    III. Simulate to Find the Number of Packets Dropped by TCP/UDP
    IV. Simulate to Find the Number of Packets Dropped due to Congestion
    V. Simulate to Compare Data Rate & Throughput.

VI. Simulate to Plot Congestion for Different Source/Destination

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

TEXT BOOK:

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

REFERENCES:

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

MALLA REDDY ENGINEERING COLLEGE FOR WOMEN

(2600SD02) Linux Programming, UI Design-Flutter

B.Tech. II Year II Sem **L T P C**

0 0 2 1

Course Objectives:

- To write shell scripts to solve problems.
- To implement some standard Linux utilities such as ls, cp etc using system calls.
- To develop network-based applications.
- To create Flutter apps with responsive layouts, state management, and custom widgets.
- To enhance Flutter apps by integrating REST APIs and adding animations for better user experience.

Course Outcomes:

- **Demonstrate** proficiency in using basic Linux utilities for file handling, process management, and system operations.
- **Develop** shell scripts to automate tasks, manage system configurations, and process text data efficiently.
- **Utilize** Linux system calls for file I/O operations, including creation, reading, writing, and modification of files.
- **Implement** process control techniques, including process creation, management, and signal handling in a Linux environment.
- Create Flutter apps with custom widgets, layouts, and forms.
- Integrate APIs, add animations, and perform UI testing in Flutter apps.

WEEK 1:

Practice File handling utilities, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities.

WEEK 2:

a) Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or directory and reports accordingly. Whenever the argument is a file it reports no of lines present init.

b) Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

WEEK 3:

a) Write a shell script to list all of the directory files in a directory

b) Write a shell script that deletes all lines containing the specified word in one or more files supplied as arguments to it.

c) Write a shell script to find factorial of a given number.

WEEK 4:

Write an awk script to count number of lines in a file that does not contain vowels Write an awk

script to find the no of characters ,words and lines in a file

WEEK 5:

Implement in c language the following Unix commands using system calls a)cat c) ls c) Scanning Directories (Ex: opendir(),readdir(),etc.)

6) Create a Flutter app showcasing common widgets (Text, Image, Container, Card, ListView).

7) Design a responsive UI using Row, Column, Stack, media queries, and breakpoints.

8) Implement screen navigation using Navigator and named routes.

9) Use stateful & stateless widgets; manage state using Provider or setState.

10) Design a form in Flutter with input fields, validation, and error handling (e.g., student registration form).

11) Fetch and display data from a REST API (e.g., weather or student info) in Flutter.

12) Add basic animations (fade or slide) to a Flutter app for enhancing UI interactions

TEXT BOOKS:

1. Unix System Programming using C++, T.Chan, PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH, 2006.
3. Beginning Linux Programming, 4th Edition, N.Matthew, R.Stones,Wrox, Wiley India Edition,rp-2008
4. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.

REFERENCES:

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Advanced Programming in the UNIX environment, 2nd Edition, W.R.Stevens, Pearson Education.
3. Unix Network Programming, W.R.Stevens,PHI.
4. UNIX for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson Education.