SUMATHI REDDY
INSTITUTE OF TECHNOLOGY FOR WOMEN
Learning at its best
AN AUTONOMOUS INSTITUTION
Estd:2009
Approved by AICTE | Affiliated to JNTUH | Accredited by NBA | Accredited by NAAC with 'A' Grade

COURSE STRUCTURE & DETAILED SYLLABUS

B.TECH R25 REGULATIONS



CSE (CYBER SECURITY)

+91 96181 22472
office@sritw.org
www.sritw.org

Ananthasagar, Hasanparthy, Hanumakonda, Telangana-506 371

SUMATHI REDDY
INSTITUTE OF TECHNOLOGY FOR WOMEN
Learning at its best
AN AUTONOMOUS INSTITUTION
Estd:2009
Approved by AICTE | Affiliated to JNTUH | Accredited by NBA | Accredited by NAAC with 'A' Grade

VISION OF THE INSTITUTE

To empower women in the field of Science & Technology and transform them as Innovators, Leaders and Entrepreneurs.

MISSION OF THE INSTITUTE

- To create and nurture a unique learning environment for faculty and students by providing state of the art infrastructure.
- Foster a culture of innovation and entrepreneurial spirit among students and faculty.
- To promote industry interaction, avenue of research and employment opportunities.

COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To nurture future-ready professionals and innovative leaders through Computer Science and Engineering for women empowerment.

MISSION OF THE DEPARTMENT

M1: To build a solid foundation in core concepts while sharpening students' ability to think critically and solve problems effectively.

M2: Fostering value-driven learning that empowers women to grow into socially aware and responsible leaders.

M3: To ignite the creative potential within students, channelling it toward innovation, research, and entrepreneurial thinking.

Program Outcomes (POs)

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

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

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

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

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

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

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

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

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

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

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

B. TECH – COMPUTER SCIENCE AND ENGINEERING (Cyber Security)

Program Educational Objectives (PEOs)	Program Specific Outcomes (PSOs)
PEO 1: Empower graduates with strong foundations in Cyber Security and CS fundamentals to excel as innovative security professionals protecting digital assets and building secure systems for industry, government and entrepreneurship.	**PSO 1:** Design and implement secure software systems and network architectures by applying cryptographic protocols, intrusion detection systems and security best practices to protect organisational assets from cyber-attacks.
PEO 2: Enable graduates to lead and adapt in the ever-evolving cyber threat landscape through continuous learning and advanced security research.	**PSO 2:** Conduct vulnerability assessments, penetration testing and digital forensic investigations using industry-standard tools.
PEO 3: Develop ethical and entrepreneurial security practitioners who work effectively in multidisciplinary teams, practise responsible security within legal frameworks and contribute to building a safer digital society.	--

SUMATHI REDDY INSTITUTE OF TECHNOLOGY FOR WOMEN
(AUTONOMOUS)
ANANTHASAGAR, HASANPARTHY, WARANGAL, TELANGANA-506371
B. Tech. in COMPUTER SCIENCE AND ENGINEERING(CYBER SECURITY)
Course Structure & Syllabus (B-25 Regulations)
Applicable from AY 2025-2026 Batch

I Year I Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	B25MA01BS	Matrices and Calculus	3	1	0	4
2.	B25PH01BS	Advanced Engineering Physics	3	0	0	3
3.	B25CS01ES	Programming for Problem Solving	3	0	0	3
4.	B25EE01ES	Basic Electrical Engineering	3	0	0	3
5.	B25ME02ES	Engineering Drawing and Computer Aided Drafting	2	0	2	3
6.	B25PH02BS	Advanced Engineering Physics Lab	0	0	2	1
7.	B25CS02ES	Programming for Problem Solving Lab	0	0	2	1
8.	B25EE02ES	Basic Electrical Engineering Lab	0	0	2	1
9.	B25CS05ES	Python Programming Lab	0	0	2	1
10.		Induction Program				
		Total Credits	14	1	8	20

I Year II Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	B25MA02BS	Ordinary Differential Equations and Vector Calculus	3	0	0	3
2.	B25CH01BS	Engineering Chemistry	3	0	0	3
3.	B25CS03ES	Data Structures	3	0	0	3
4.	B25EC01ES	Electronic Devices and Circuits	3	0	0	3
5.	B25EN01HS	English for Skill Enhancement	3	0	0	3
6.	B25CH02BS	Engineering Chemistry Lab	0	0	2	1
7.	B25CS04ES	Data Structures Lab	0	0	2	1
8.	B25EN02HS	English Language and Communication Skills Lab	0	0	2	1
9.	B25ME01ES	Engineering Workshop	0	0	2	1
10.	B25CS06ES	IT Workshop	0	0	2	1
		Total Credits	15	0	10	20

II Year I Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1	B25MA04BS	Mathematical and Statistical Foundations	3	0	0	3
2	B25CY01PC	Computer Organization	3	0	0	3
3	B25CY02PC	Java Programming	3	0	0	3
4	B25CS12PC	Software Engineering	3	0	0	3
5	B25CS13PC	Database Management System	3	0	0	3
6	B25MA06BS	Computational Mathematics Lab	0	0	2	1
7	B25CY03PC	Java Programming Lab	0	0	2	1
8	B25CS15PC	Software Engineering Lab	0	0	2	1
9	B25CS16PC	Database Management Systems Lab	0	0	2	1
10	B25CY08SD	Node JS/ React JS/ Django, UI Design - Flutter	0	0	2	1
		Total Credits	15	0	10	20

II Year II Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1	B25CS09PC	Discrete Mathematics	3	0	0	3
2	B25CS18PC	Operating Systems	3	0	0	3
3	B25CY05PC	Formal Languages and Automata Theory	3	0	0	3
4	B25CS20PC	Computer Networks	3	0	0	3
5	B25CY09PC	Algorithm Design and Analysis	3	0	0	3
6	B25MS01HS	Innovation and Entrepreneurship	2	0	0	2
7	B25CS22PC	Operating Systems Lab	0	0	2	1
8	B25CS23PC	Computer Networks lab	0	0	2	1
9	B25CY10PC	Algorithm Design and Analysis Lab	0	0	2	1
10	B25CY04SD	Data Visualization-R Programming / Power BI / Tableau / Google Chart	0	0	2	1
11	B25VA03HS	Indian Knowledge System	1	0	0	1
		Total Credits	17	0	8	22

B25MA01BS: MATRICES AND CALCULUS

I Year I Semester

L	T	P	C
3	1	0	4

Pre-requisites: Mathematical Knowledge at pre-university level

Objectives:

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

Course outcomes:

- After learning the contents of this paper, the student must be able to
- Write the matrix representation of a set of linear equations and to analyse the solution of the system of equations
- Find the Eigen values and Eigen vectors
- Reduce the quadratic form to canonical form using orthogonal transformations.
- Solve the applications of the mean value theorems.
- Find the extreme values of functions of two variables with/without constraints.
- Evaluate the multiple integrals and apply the concept to find areas, volumes.

UNIT-I: Matrices

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

UNIT-II: Eigen values and Eigen vectors

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

UNIT-III: Single Variable Calculus

Limit and Continuous of functions and its properties. Mean value theorems: Rolle's theorem – Lagrange's Mean value theorem with their Geometrical Interpretation and applications– Cauchy's Mean value Theorem–Taylor's Series (All the theorems without proof).
Curve Tracing: Curve tracing in cartesian coordinates.

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

UNIT-V: Multivariable Calculus (Integration)
Evaluation of Double Integrals (Cartesian and polar coordinates) – change of order of integration (only Cartesian form)–Change of variables for double integrals (Cartesian to polar). Evaluation of Triple Integrals –Change of variables for triple integrals (Cartesian to Spherical and Cylindrical polar coordinates). Applications: Areas by double integrals and volumes by triple integrals.

TEXT BOOKS:

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

REFERENCE BOOKS:

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

Web References:

1. https://mathworld.wolfram.com/CanonicalForm.html
2. https://mathworld.wolfram.com/topics/LinearSystemsofEquations.html
3. https://mathworld.wolfram.com/topics/MatrixEigenvalues.html
4. https://mathworld.wolfram.com/LagrangeMultiplier.html

E -TEXT BOOKS:

1. **https://www.e-booksdirectory.com/listing.php?category=539**
2. **https://www.e-booksdirectory.com/listing.php?category=4**

MOOCS COURSE:

1. **https://onlinecourses.nptel.ac.in/noc25_ma97/preview**
2. **https://onlinecourses.nptel.ac.in/noc25_ma98/preview**

B25PH01BS: ADVANCED ENGINEERING PHYSICS

I Year I Semester

L	T	P	C
3	0	0	3

Pre-requisites: 10+2 Physics

Course Objectives:

- To study crystal structures, defects, and material characterization techniques like XRD and SEM.
- To understand fundamental concepts of quantum mechanics and their applications in solids and nanomaterials.
- To introduce quantum computing principles, quantum gates, and basic quantum algorithms.
- To learn the properties and applications of magnetic and dielectric materials.
- To explore the working and applications of lasers and fibre optics in modern technology.
-

Course Outcomes:

- CO1: Analyze crystal structures, identify defects, and apply XRD and SEM techniques for material characterization.
- CO2: Apply quantum mechanical principles to explain particle behaviour and energy band formation in solids.
- CO3: Understand quantum computing concepts, use quantum gates, and explain basic quantum algorithms.
- CO4: Classify magnetic and dielectric materials and explain their properties, synthesis, and applications.
- CO5: 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, scanning electron microscopy (SEM): block diagram, working principle.

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, effective mass of electron, formation of energy bands, origin of bandgap, classification of solids, concept of discrete energy levels and quantum confinement in nanomaterials.

UNIT - III: Quantum Computing

Introduction, linear algebra for quantum computation, Dirac's Bra and Ket notation and their properties, Hilbert space, Bloch's sphere, concept of quantum computer, classical bits, Qubits,

multiple Qubit system, quantum computing system for information processing, evolution of quantum systems, quantum measurements, entanglement, quantum gates, challenges and advantages of quantum computing over classical computation, quantum algorithms: Deutsch-Jozsa, Shor, Grover.

UNIT - IV: 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

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://www.fi.muni.cz/usr/gruska/qbook1.pdf**
- **https://profmcruz.wordpress.com/wp-content/uploads/2017/08/quantum-computation-and- quantum-information-nielsen-chuang.pdf**

B25CS01ES: PROGRAMMING FOR PROBLEM SOLVING

I Year I Semester

L	T	P	C
3	0	0	3

Course Objectives:

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

Course Outcomes: The student will learn

- To write algorithms and to draw flowcharts for solving problems.
- To convert the algorithms/flowcharts to C programs.
- To code and test a given logic in the C programming language.
- To decompose a problem into functions and to develop modular reusable code.
- To use arrays, pointers, strings and structures to write C programs.
- 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, Formatting Numbers in Program Output.

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, Multiple Calls to a Function with Input/ Output Parameters, Scope of Names, 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, Parallel Arrays and Enumerated Types, Multidimensional Arrays.

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

UNIT - IV:

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

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

UNIT - V:

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

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

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill.
3. 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.

B25EE01ES: BASIC ELECTRICAL ENGINEERING

I Year I Semester

L	T	P	C
3	0	0	3

Prerequisites: Mathematics

Course Objectives:

- To understand DC and Single &Three phase AC circuits
- To study and understand the different types of DC ,AC machines and Transformers.
- To import the knowledge of various electrical installations and the concept of power, power factor and its improvement.

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

- Understand and analyze basic Electrical circuits
- Study the working principles of Electrical Machines and Transformers
- Introduce components of Low Voltage Electrical Installations.

Course Objectives	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
To understand DC and Single & Three phase AC circuits.	3	2	1		2	0	0	1	2	0	1	2
To study and understand the different types of DC, AC machines and Transformers.	3	2	1	1	3	0	0	0	2	0	1	1
To import the knowledge of various electrical installations and the concept of power, power factor and its improvement.	3	2	0		3	0	0	0	1	2	1	1

Course Outcomes	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
Understand and analyse basic Electrical circuits	3	2	1	0	1	0	0	0	2	0	2	2
Study the working principles of Electrical Machines and Transformers	3	2	1	0	3	1	0	1	1	2	1	2
Introduce components of Low Voltage Electrical Installations.	3	2	1	1	3	2	0	0	1	0	2	2

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

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

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

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

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

TEXTBOOKS:

1. D.P.Kothariand I.J.Nagrath, “Basic Electrical Engineering”, Tata Mc Graw Hill, 4th Edition, 2019.
2. MSNaiduandSKamakshaiah,“BasicElectricalEngineering”,TataMcGrawHill,2nd Edition, 2008.

REFERENCEBOOKS:

1. P.Ramana,M.Suryakalavathi,G.T.Chandrasheker,“BasicElectricalEngineering”,S.Chand, 2nd Edition, 2019.
2. D.C.Kulshreshtha,“Basic Electrical Engineering”,McGrawHill,2009
3. M.S.Sukhija,T.K.Nagsarkar,“BasicElectricalandElectronicsEngineering”,Oxford,1st Edition, 2012.
4. AbhijitChakrabarthi,SudiptaDebnath,ChandanKumarChanda,“BasicElectrical
5. Engineering”,2nd Edition,McGrawHill,2021.
6. L.S.Bobrow,“FundamentalsofElectricalEngineering”,OxfordUniversityPress,2011.
7. E.Hughes,“ElectricalandElectronicsTechnology”,Pearson,2010.
8. V.D.Toro, “Electrical Engineering Fundamentals”, Prentice Hall India, 1989

B25ME02ES: ENGINEERING DRAWING AND COMPUTER AIDED DRAFTING

I Year I Semester

L	T	P	C
2	0	2	3

Course Objectives:

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

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

1. Understand and apply the principles of orthographic and isometric projections.
2. Create sectional views and dimensioned drawings using BIS standards.
3. Use CAD software to generate 2D engineering drawings.
4. Visualize and construct solid models from 2D views.
5. Interpret and produce engineering drawings of mechanical components and assemblies.
6. Demonstrate drafting skills for practical and industrial applications.

UNIT – I: Introduction to Engineering Graphics (Conventional)

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

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

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

UNIT – III: Projections of Regular Solids (Conventional and Computer Aided)

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

UNIT – IV: Development of Surfaces (Conventional): Prism, Cylinder, Pyramid and Cone.

UNIT – V: Isometric Projections (Conventional and Computer Aided)

Principles of Isometric Projection, Isometric Scale, Isometric Views, Conventions, Isometric Views of Lines, Plane Figures, Simple and Compound Solids, Isometric Projection of objects having non, isometric lines. Isometric Projection of Spherical Parts. Conversion of Isometric Views to Orthographic Views and Vice- versa, Conventions. Conversion of orthographic projection into isometric view.

Note:

1. The End Semester Examination will be in conventional mode.
2. CIE – I will be in conventional mode.
3. CIE – II will be using Computer.

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25PH02BS: ADVANCED ENGINEERING PHYSICS LAB

I Year I Semester

L	T	P	C
0	0	2	1

Course Objectives:

- To understand how light interacting with matter can behave like a stream of photons(particles).
- To synthesize and study the physical properties of materials like semiconductors, ferromagnetic, and ferroelectric substances.
- To perform semiconductor characterization using Hall effect and band gap experiments.
- To explore the working principles of lasers and optical fibers through hands-on experiments.
- To develop skills in data analysis, interpretation, and scientific reporting.

Course Outcomes:

- CO1: Understanding the phenomenon of light causing the emission of electrons from material and the factors influencing this process.
- CO2: Determine key electrical, magnetic, and optical properties of semiconductors and other functional materials.
- CO3: Characterize semiconductors using Hall effect and energy gap measurement techniques.
- CO4: Demonstrate working knowledge of laser systems and optical fiber parameters through experimental study.
- CO5: Apply scientific methods for accurate data collection, analysis, and technical report writing.

List of Experiments:

1. Photo electric effect.
2. Determination of energy gap of a semiconductor.
3. Determination of Hall coefficient and carrier concertation of a given semiconductor.
4. Determination of magnetic moment of a bar magnet and horizontal earth magnetic field.
5. Study of B-H curve of a ferro magnetic material.
6. Two Probe Method.
7. Determination of dielectric constant of a given material.
8. Determination of Curie's temperature of a given ferroelectric material.
9. A) Determination of wavelength of a laser using diffraction grating.
   B) Study of V-I & L-I characteristics of a given laser diode.
10. A) Determination of numerical aperture of a given optical fibre.
    B) Determination of bending losses of a given optical fibre.

Note: Any 8 experiments are to be performed.

B25CS02ES: PROGRAMMING FOR PROBLEM SOLVING LAB

I Year I Semester

L	T	P	C
0	0	2	1

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

Course Objectives: The students will learn the following:

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

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

- formulate the algorithms for simple problems
- translate given algorithms to a working and correct program
- correct syntax errors as reported by the compilers
- identify and correct logical errors encountered during execution
- represent and manipulate data with arrays, strings and structures
- use pointers of different types
- create, read and write to and from simple text and binary files
- modularize the code with functions so that they can be reused

PRACTICE SESSIONS:

Simple numeric problems:

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

Expression Evaluation:

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

b) Write a program that finds if a given number is a prime number.
c) Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.
d) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

Arrays, Pointers and Functions:

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

Files:

a) Write a C program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.
b) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).

Strings:

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

Sorting and Searching:

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

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
2. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill
3. 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

B25EE02ES: BASIC ELECTRICAL ENGINEERING LAB

I Year I Semester

L	T	P	C
0	0	2	1

Prerequisites: Basic Electrical Engineering

Course Objectives:

- To measure the electrical parameters for different types of DC and AC circuits using conventional and theorems approach.
- To study the transient response of various R, L and circuits using different excitations.
- To determine the performance of different types of DC, AC machines and Transformers.

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

- Verify the basic Electrical circuits through different experiments.
- Evaluate the performance calculations of Electrical Machines and Transformers through various testing methods.
- Analyze the transient responses of R, Land C circuits for different input conditions.

Course Objectives	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
To measure the electrical parameters for different types of DC and AC circuits using conventional and theorems approach	3	2	1		2	0	0	1	2	0	1	2
To study the transient response of various R, Land C circuits using Different excitations	3	2	1	1	3	0	0	0	2	0	1	1
To determine the performance of different types of DC, AC machines and Transformers	3	2	0		3	0	0	0	1	2	1	1

Course Outcomes	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
Verify the basic Electrical circuits through different experiments	3	2	1	0	1	0	0	0	2	0	2	2
Evaluate the performance calculations of Electrical Machines and Transformers through various Testing methods	3	2	1	0	3	1	0	1	1	2	1	2
Analyze the transient responses of R,L and C circuits for different input conditions	3	2	1	1	3	2	0	0	1	0	2	2

List of experiments/demonstrations:

PART-A(compulsory)

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

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

- Verification of Superposition theorem.
- ThreePhaseTransformer:VerificationofRelationshipbetweenVoltagesandCurrents (Star-Delta, Delta-Delta, Delta-star, Star-Star)
- LoadTestonSinglePhaseTransformer(CalculateEfficiencyandRegulation)
- MeasurementofActiveandReactivePowerinabalancedThree-phasecircuit
- No-Load Characteristics of a Three-phase Alternator

TEXTBOOKS:

1. D.P.KothariandI.J.Nagrath,"BasicElectricalEngineering",TataMcGrawHill,4th Edition, 2019.
2. MSNaiduandSKamakshaiah,"BasicElectricalEngineering",TataMcGrawHill,2nd Edition, 2008.

REFERENCEBOOKS:

1. P.Ramana,M.Suryakalavathi,G.T.Chandrasheker,"Basic Electrical Engineering",S.Chand, 2nd Edition, 2019.
2. D.C.Kulshreshtha,"Basic Electrical Engineering",McGrawHill,2009
3. M.S.Sukhija,T.K.Nagsarkar,"Basic Electrical and Electronics Engineering",Oxford,1st Edition, 2012.
4. AbhijitChakrabarthi,SudiptaDebnath,ChandanKumarChanda,"BasicElectrical Engineering",2ndEdition,McGrawHill,2021.
5. L.S.Bobrow,"FundamentalsofElectricalEngineering",OxfordUniversityPress,2011.
6. E.Hughes,"ElectricalandElectronicsTechnology",Pearson,2010.
7. V.D.Toro, "Electrical EngineeringFundamentals",PrenticeHallIndia,1989.

B25CS07ES: PYTHON PROGRAMMING LAB

I Year I Semester

L	T	P	C
0	0	2	1

Course Objectives:

- 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 I/O, Python standard library
- Implement Digital Systems using Python

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.
1. Start a Python interpreter and use it as a Calculator.
2. Write a program to calculate compound interest when principal, rate and number of periods are given.
3. Read the name, address, email and phone number of a person through the keyboard and print the details.
4. Print the below triangle using for loop.
   5
   4 4
   3 3 3
   2 2 2 2
   1 1 1 1 1
5. Write a program to check whether the given input is digit or lowercase character or uppercase character or a special character(use 'if-else-if' ladder)
6. Python program to print all prime numbers in a given interval (use break)
7. Write a program to convert a list and tuple into arrays.
8. Write a program to find common values between two arrays.
9. Write a function called palindrome that takes a string argument and returnsTrue if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.
10. Write a function called is_sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.
11. Write a function called has_duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.

12. Write a function called remove_duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.
13. The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add "I", "a", and the empty string.
14. Write a python code to read dictionary values from the user. Construct a function to invert its content. i.e., keys should be values and values should be keys.
15. Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
16. Remove the given word in all the places in a string?
17. Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding upper case letter and the rest of the letters in the word by corresponding letters in lower case without using a built-in function?
18. Writes a recursive function that generates all binary strings of n-bit length
19. Write a python program that defines a matrix and prints
20. Write a python program to perform multiplication of two square matrices
21. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.
22. Use the structure of exception handling all general-purpose exceptions.
23. Write a function called draw_rectangle that takes a Canvas and a Rectangle as arguments and draws a representation of the Rectangle on the Canvas.
24. Add an attribute named color to your Rectangle objects and modify draw_rectangle so that it uses the color attribute as the fill color.
25. Write a function called draw_point that takes a Canvas and a Point as arguments and draws a representation of the Point on the Canvas.
26. Define a new class called Circle with appropriate attributes and instantiate a few Circle objects. Write a function called draw_circle that draws circles on the canvas.
27. Write a python code to read a phone number and email-id from the user and validate it for correctness.
28. Write a Python code to merge two given file contents into a third file.
29. Write a Python code to open a given file and construct a function to check for given words present in it and display on found.
30. Write a Python code to Read text from a text file, find the word with most number of occurrences
31. Write a function that reads a file *file1*and displays the number of words, number of vowels, blank spaces, lower case letters and uppercase letters.
32. Import numpy, Plotpy and Scipy and explore their functionalities.
33. Install NumPypackage with pip and explore it.
34. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
35. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as Submit and Reset.

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming A Modular Approach with Graphics, Database, Mobile, and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson

3. Introduction to Python Programming, Gowrishakar S, Veena A, CRC Press
4. Programming with Python, A User's Book, Michael Dawson, Cengage Learning, India Edition
5. Python for Data Science, Dr. Mohd Abdul Hameed, Wiley publications
6. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech press
7. Introduction to Python, Gowrishankar S, Veena A., CRC Press

B25MA02BS: ORDINARY DIFFERENTIALEQUATIONS AND VECTOR CALCULUS

I Year II Semester

L	T	P	C
3	0	0	3

Pre-requisites: Mathematical Knowledge at pre-university level

Course Objectives: To learn

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

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

- Identify whether the given differential equation of first order is exact or not
- Solve higher differential equation and apply the concept of differential equation to real world problems.
- Use the Laplace Transforms techniques for solving Ordinary Differential Equations.
- Evaluate the Line, Surface and Volume integrals and converting them from one to another

UNIT-I: First Order Ordinary Differential Equations

Exact differential equations –Equations reducible to exact differential equations–linear and Bernoulli's equations–Orthogonal Trajectories (only in Cartesian Coordinates). Applications: Newton's law of cooling–Law of natural growth and decay.

UNIT-II: Ordinary Differential Equations of Higher Order

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

UNIT-III: Laplace Transforms

Laplace Transforms: Laplace Transform of standard functions –First shifting theorem –Laplace transforms of functions multiplied by 't' and divided by 't' –Laplace transforms of derivatives and integralsoffunction–EvaluationofintegralsbyLaplacetransforms–Laplacetransformofperiodic functions –Inverse Laplace transform by different methods, convolution theorem (without proof). Applications: solving Initial value problems by Laplace Transform method.

UNIT-IV: Vector Differentiation

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

UNIT-V:Vector Integration 10L

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, 36th Edition, 2010.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 5th Edition, 2016.

REFERENCEBOOKS:

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

Web References:

1. https://www.geeksforgeeks.org/maths/ordinary-differential-equations/
2. https://www.efunda.com/math/laplace_transform/index.cfm?search_string=laplace%20transforms

E -TEXT BOOK

1. https://www.e-booksdirectory.com/details.php?ebook=10205

MOOCS COURSE

1. https://archive.nptel.ac.in/content/storage2/courses/122104018/node69.html
2. https://onlinecourses.swayam2.ac.in/cec20_ma22/preview

B25CH01BS: ENGINEERING CHEMISTRY

I Year II Semester

L	T	P	C
3	0	0	3

Course Objectives:

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

Course Outcomes:

- Students will be able to understand the fundamental properties of water and its applications in both domestic and industrial purposes.
- Students will gain basic knowledge of electrochemical processes and their relevance to corrosion and its control methods.
- Students will comprehend the significance and practical applications of batteries and various energy sources, enhancing their potential as future engineers and entrepreneurs.
- Students will learn the basic concepts and properties of polymers and other engineering materials.
- 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:

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

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

UNIT-II: Electrochemistry and Corrosion:

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

Corrosion: Introduction- Definition, causes and effects of corrosion – Theories of corrosion, chemical and electrochemical theories of corrosion, Types of corrosion: galvanic, water-line and pitting corrosion. Factors affecting rate of corrosion - Nature of the metal, Nature of the

corroding environment. Corrosion control methods - Cathodic protection Methods - Sacrificial anode and impressed current methods.

UNIT–III:Energy sources:
Batteries: Introduction – Classification of batteries - Primary, secondary and reserve batteries with examples. Construction, working and applications of Zn-air and Lithium ion battery. Fuel Cells – Differences between a battery and a fuel cell, Construction and applications of Direct Methanol Fuel Cell (DMFC).

Fuels: Introduction and characteristics of a good fuel, Calorific value–Units-HCV, LCV-Dulongs formula - Numerical problems.

Fossil fuels: Introduction, Classification, Petroleum-Refining of Crude oil, Cracking-Types of cracking - 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:
Definition-Classification of polymers: Based on origin and tacticity with examples–Types of polymerization - Addition (free radical addition mechanism) and condensation polymerization. Plastics, Elastomers and Fibers: Definition and applications (PVC,Buna-S, Nylon-6,6). Differences between thermoplastics and thermo setting plastics, Fiber reinforced plastics (FRP).

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

UNIT-V-Advanced Functional Materials:
Smart materials: Introduction, Classification with examples - Shape Memory Alloys – Nitinol, Piezoelectric materials – quartz and their engineering applications. Biosensor- Definition, Amperometric Glucose monitor sensor.

Interpretative spectroscopic applications 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) - Tumor detection in medical applications.

TEXTBOOKS:

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

REFERENCEBOOKS:

1. Engineering Chemistry: by ThirumalaChary Laxminarayana & Shashikala, Pearson Publications (2020)
2. Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company (P) Ltd.Delhi2011.
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 inGreen Hydrogen byEditors: Paramvir Singh, AvinashKumar 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|October2023**
8. **E-books:**
7. **https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2u**

B25CS03ES: DATA STRUCTURES

I Year II Semester

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.

UNIT – I

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

UNIT - II

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

UNIT – III

Multi way Search Trees: Introduction, B Trees, B Trees ADT, 2-3 Trees, 2-3- Tree, B* Tree, B+ Trees

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

Searching: Introduction, Interpolation Search, Jump search

UNIT - IV

Graphs: Introduction, Directed Graphs, Bi connected Components, Representation of Graphs, Graph Traversal Algorithms, Graph ADT, Applications of Graphs

Sorting: Radix Sort, Heap sort, Shell Sort, Tree Sort,

UNIT – V

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

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

B25EC01ES: ELECTRONIC DEVICES AND CIRCUITS

L	T	P	C
3	0	0	3

I Year II Semester

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

Course Outcomes: By the end of this course, students will be able to:

- CO1: Analyze the electrical characteristics and models of semiconductor diodes and apply them in rectifier and clipping circuits.
- CO2: Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.
- CO3: Design appropriate biasing networks for BJTs and determine the operating point for amplifier applications.
- CO4: Analyze transistor amplifier circuits using h-parameter models and assess performance for various configurations.
- CO5: Analyze the structure, working, and characteristics of JFETs, MOSFETs, and advanced devices like FinFETs and CNTFETs, and compare modern device technologies.

Course Articulation Matrix

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

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

UNIT - II:
Bipolar Junction Transistor (BJT): Structure and working principle of BJT, Current components and transistor action, Configurations: Common Base (CB), Common Emitter (CE), Common Collector (CC), Input and output characteristics, Determination of h-parameters from transistor characteristics.

UNIT - III:

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

UNIT - IV:

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

UNIT - V:

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

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25EN01HS - ENGLISH FOR SKILLS ENHANCEMENT

L	T	P	C
3	0	0	3

I Year II Semester

LEARNING OBJECTIVES: This course will enable the students to:

- Improve their vocabulary.
- Use appropriate sentence structures in their oral and written communication.
- Develop their reading and study skills.
- Equip students to write paragraphs, essays, précis and draft letters.
- Acquire skills for technical report writing.

COURSE OUTCOMES: Students will be able to:

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

SYLLABUS: The course content / study material is divided into Five Units.

UNIT –I

Theme:

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

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

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

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

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

UNIT –II

Theme:

Digital Transformation Lesson on 'Emerging Technologies' from the prescribed textbook titled English for the Young in the Digital World published by Orient Blackswan Pvt. Ltd.

Vocabulary: Homophones, Homonyms and Homographs

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

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

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

UNIT –III

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

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

Writing: Writing Practices- Note Making-Précis Writing.

UNIT –V

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

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

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

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

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

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

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

TEXT BOOK:

1. 1. Board of Editors. 2025. English for 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.Karal, Rajeevan. 2023.
2. English Grammar Just for You. Oxford University Press. New Delhi2024.
3. Empowering with Language: Communicative English for Undergraduates. Cengage Learning India Pvt. Ltd. New Delhi Sanjay Kumar & Pushp Lata. 2022.
4. Communication Skills – A Workbook. Oxford University Press. New Delhi
5. Wood,F.T. (2007). Remedial English Grammar. Macmillan.

B25CH02BS: ENGINEERING CHEMISTRY LAB

I Year II Semester

L	T	P	C
0	0	2	1

Course Description:
The course includes experiments based on fundamental principles of chemistry essential for engineering students, aiming to develop practical skills and reinforce theoretical concepts.

Course Objectives

- Students will understand and perform experiments based on core chemical principles relevant to engineering applications.
- Students will earn to estimate the hardness of water to assess its suitability for drinking purposes.
- Students will acquire the ability to perform acid-base titrations using instrumental methods such as conductometry, potentiometry, and pH metry.
- Students will gain hands-on experience in synthesizing polymers like Bakelite and Nylon – 6,6 in the laboratory.
- Students will earn to determine the unknown concentration of potassium permanganate (KMnO4) using a calibration curve.

Course Outcomes:

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

List of Experiments:

I. Volumetric Analysis: Estimation of Hardness of water by EDTA Complexometry method.
II. Conductometry:
    1. Estimation of the concentration of strong acid by Conductometry.
    2. Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.
III. Potentiometry:
    1. Estimation of concentration of Fe^{+2}ion by Potentiometry using KMnO4.
    2. Estimation of concentration of strong acid with strong baseby Potentiometry using quinhydrone
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. 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.

REFERENCEBOOKS:

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 5^{th} 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).

B25CS04ES: DATA STRUCTURES LAB

L	T	P	C
0	0	2	1

I Year II Semester

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

Course Objectives:

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

Course Outcomes:

- 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.
- Ability to Implement searching and sorting algorithms

List of Experiments

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

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

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

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

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

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

Write a program that implement stack (its operations) using

i) Arrays ii) ADT

Write a program that implement Queue (its operations) using

i) Arrays ii) ADT

Write a program that implements the following sorting methods to sort a given list of integers in ascending order

i) Radix Sort, ii) Heap sort, iii) Shell Sort, iv) Tree Sort

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

Write a program to implement

i) Binary Search tree ii) B Trees iii) B+ Trees iv) AVL trees v) Red - Black trees

Write a program to implement the graph traversal methods.

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

TEXT BOOKS:

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

REFERENCE BOOK:

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

B25EN02HS - ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB

I Year II Semester

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.

Listening Skills:
Objectives

- To enable students develop their active listening skills
- To equip students with necessary training in listening, so that they can comprehend the speech of people from different linguistic backgrounds
- Speaking Skills:
- To improve their pronunciation and neutralize accent
- To enable students express themselves fluently and appropriately
- To practise speaking in social and professional contexts

Learning Outcomes: Students will be able to:

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

Syllabus: English Language and Communication Skills Lab (ELCS) shall have two parts:
Computer Assisted Language Learning (CALL) Lab which focusses on listening skills
Interactive Communication Skills (ICS) Lab which focusses on speaking skills

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

Exercise – I
CALL Lab: Instruction: Speech Sounds-Listening Skill - Importance – Purpose - Types-Barriers- Active Listening

Practice: Listening to Distinguish Speech Sounds (Minimal Pairs) - Testing Exercises
ICS Lab: Diagnostic Test – Activity titled ‘Express Your View’
Instruction: Spoken and Written language - Formal and Informal English - Greetings - Introducing Oneself and Others

Practice: Any Ice-Breaking Activity

Exercise – II
CALL Lab: Instruction: Listening vs. Hearing - Barriers to Listening
Practice: Listening for General Information - Multiple Choice Questions - Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)

ICS Lab:
Instruction: Features of Good Conversation – Strategies for Effective Communication Practice: Role Play Activity - Situational Dialogues –Expressions used in Various Situations –Making Requests and Seeking Permissions – Taking Leave - Telephone Etiquette

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

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

Exercise – IV
CALL Lab: Instruction: Techniques for Effective Listening
Practice: Listening for Specific Details - Listening - Gap Fill Exercises - Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)
ICS Lab: Instruction: How to Tell a Good Story - Story Star- Sequencing-Creativity Practice: Activity on Telling and Retelling Stories - Collage

Exercise – V
CALL Lab: Instruction: Identifying the literal and implied meaning
Practice: Listening for Evaluation - Write the Summary – Listening Comprehension Exercises (It is essential to identify a suitable passage with exercises for practice.)
ICS Lab: Instruction: Understanding Non-Verbal Communication
Practice: Silent Speech - Dumb Charades Activity □
Post-Assessment Test on 'Express Your View'
Minimum Requirement of infrastructural facilities for ELCS Lab:
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:
Computers with Suitable Configuration
ii) High Fidelity Headphones
Interactive Communication Skills (ICS) Lab:
The Interactive Communication Skills Lab: A Spacious room with movable chairs and audiovisual aids with a Public Address System, a T. V. or LCD, a digital stereo – audio & video system and camcorder etc.

Note: English Language Teachers are requested to prepare Materials / Handouts for each Activity for the Use of those Materials in CALL & ICS Labs.

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 Jeff Stranks 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 ICS Lab Activities. Orient Blackswan Pvt. Ltd.
3. Mishra, Veerendra et al. (2020). English Language Skills: A Practical Approach. Cambridge University Press
4. (2022). English Language Communication Skills – Lab Manual cum Workbook. Cengage Learning India Pvt. Ltd.
5. Ur, Penny and Wright, Andrew. 2022. Five Minute Activities – A Resource Book for Language Teachers. Cambridge University Press.

B25ME01ES: ENGINEERING WORKSHOP

I Year II Semester

L	T	P	C
0	0	2	1

Prerequisites: Practical skill

Course Objectives:

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

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

1. Understand the basic manufacturing processes and operations.
2. Use hand tools and equipment safely and efficiently.
3. Perform basic operations in carpentry, fitting, welding, sheet metal work, and machining
4. Read and interpret workshop drawings
5. Develop teamwork, time management, and quality awareness in a workshop environment.

1. TRADES FOR EXERCISES: At least two exercises from each trade:
i. Carpentry: T- Lap Joint, Dovetail Joint, Mortise and Tenon Joint
ii. Fitting: V- Fit, Dovetail Fit and Semi- circular fit
iii. Tin Smithy: Square Tin, Rectangular Tray and Conical Funnel
iv. Foundry: Preparation of Green Sand Mould using Single Piece and Split Pattern
v. Welding Practice: Arc Welding and Gas Welding
vi. House wiring: Parallel and Series, Two-way Switch and Tube Light
vii. Black Smithy: Round to Square, Fan Hook and S- Hook

2. TRADES FOR DEMONSTRATION AND EXPOSURE: Plumbing, Machine Shop, Metal Cutting (Water Plasma), Power tools in construction and Wood Working

TEXT BOOKS:

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

REFERENCE BOOK:

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

B25CS06ES: IT WORKSHOP

I Year II Semester

L	T	P	C
0	0	2	1

Course Objectives:

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

Course Outcomes:

- Perform Hardware troubleshooting
- Understand Hardware components and inter dependencies
- Safeguard computer systems from viruses/worms
- Document/ Presentation preparation
- Perform calculations using spreadsheets

PC Hardware

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

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

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

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Internet & World Wide Web

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

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

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

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

Task 1 – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of LaTeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using LaTeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 3: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel

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

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

Task 2: Calculating GPA - Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP

Task 3: Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

PowerPoint

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

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

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

REFERENCE BOOKS:

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

B25MA04BS: MATHEMATICAL AND STATISTICAL FOUNDATIONS

	L	T	P	C
II Year I Semester	**3**	**0**	**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.

B25CY01PC: COMPUTER ORGANIZATION

L	T	P	C
3	0	0	3

II Year I Semester

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 7th Edition, Pearson/ PHI.
4. Structured Computer Organization – Andrew S. Tanenbaum, 6th Edition PHI/Pearson.

B25CY02PC: JAVA PROGRAMMING

II Year I Semester

L	T	P	C
3	0	0	3

Course Outcomes:

1. Identify the model of Object-Oriented Programming: Abstract data types, Encapsulation, Inheritance and Polymorphism.
2. Summarize the fundamental features like Interfaces, Exceptions and Collections.
3. Correlate the advantages of Multi-threading.
4. Design interactive programs using Applets, AWT and Swings.
5. Develop real time applications using the features of Java.

UNIT - I

OBJECT ORIENTED THINKING

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

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

UNIT - II

CLASSES, INHERITANCE, POLYMORPHISM

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

Strings: String, String Buffer, String Tokenizer.

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

UNIT - III

INTERFACES, PACKAGES, EXCEPTIONS

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

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

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

UNIT - IV

MULTITHREADING, COLLECTIONS

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

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

UNIT - V

INTRODUCTION TO APPLET, SWING-INTRODUCTION, LIMITATIONS OF AWT, MCV ARCHITECTURE, SWING COMPONENTS.

JDBC ARCHITECTURE: The 4 types of JDBC drivers. Connectivity Steps: Registering the driver, establishing connections, and creating statements. Executing Queries: Working with ResultSet, PreparedStatement, and CallableStatement. Transaction Management: Commit and Rollback operations. Advanced Topics: Metadata and connection pooling basics.

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25CS12PC: SOFTWARE ENGINEERING

L	T	P	C
3	0	0	3

II Year I Semester

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Ability to translate end-user requirements into system and software requirements, using e.g.
- UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT - I

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

UNIT - II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.
Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III

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

UNIT - IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.
Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT - V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM. Quality Management: Quality concepts, software quality

assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.
4. Fundamentals of Software Engineering-Rajib Mall, PHI.

B25CS13PC: DATABASE MANAGEMENT SYSTEMS

II Year I Semester

L	T	P	C
3	0	0	3

Prerequisites: A course on "Data Structures".

Course Objectives:

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

Course Outcomes:

1. Gain knowledge of fundamentals of DBMS, database design and normal forms
2. Master the basics of SQL for retrieval and management of data.
3. Be acquainted with the basics of transaction processing and concurrency control.
4. Familiarity with database storage structures and access techniques

UNIT - I

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

UNIT - II

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

UNIT - III

SQL: QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and
EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.
Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued dependencies, FOURTH normal form, FIFTH normal form.

UNIT - IV

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions.

UNIT - V

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25MA06BS: COMPUTATIONAL MATHEMATICS LAB
(Using Python/MATLAB software)

II Year I Semester

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 coefficients6P

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.

B25CY03PC: JAVA PROGRAMMING LAB

II Year I Semester

L	T	P	C
0	0	2	1

Course Outcomes:

1. Analyze a problem, identify and define the computing requirements appropriate to its solution
2. using object-oriented programming concepts.
3. Design the applications using Inheritance, Polymorphism and Synchronization concepts.
4. Handle exceptions at Compile time and Run time.
5. Solve the real-world problems using Java Collection framework.
6. Develop GUI applications using Applets, AWT and Swings.

TASK 1

Write java programs that implement the following

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

TASK 2

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

TASK 3

Write java programs that uses the following keywords

a) this b) super c) static d) final

TASK 4

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

TASK 5

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

TASK 6

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

TASK 7

a) Write a Java program that creates three threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds.

b) Write a Java program that correctly implements producer consumer problem using the concept of interthread communication.

TASK 8

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

TASK 9

a) Develop an applet that displays a simple message.
b) Develop an applet that receives an integer in one text field and compute its factorial value and return it in another text field, when the button named "Compute" is clicked.
c) Write a Java program that works as a simple calculator. Use a grid layout to arrange button for the digitsand for the +, -,*, % operations. Add a text field to display the result.

TASK 10

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

TASK 11

a) Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields Num1 and Num 2.
b) The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1or Num2 were not an integer, the program would throw Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception and display the exception in a message dialog box.

TASK 12

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25CS15PC: SOFTWARE ENGINEERING LAB

L	T	P	C
0	0	2	1

II Year I Semester

Prerequisites: A course on "Programming for Problem Solving".
Co-requisite: A Course on "Software Engineering".

Course Objectives:

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

Course Outcomes:

- Ability to translate end-user requirements into system and software requirements
- Ability to generate a high-level design of the system from the software requirements
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

List of Experiments

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

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

Sample Projects:

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

B25CS16PC: DATABASE MANAGEMENT SYSTEMS LAB

L	T	P	C
0	0	2	1

II Year I Semester

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.

B25CY08SD: NODE JS/ REACT JS/ DJANGO, UI DESIGN - FLUTTER

L	T	P	C
0	0	2	1

II Year I Semester

Prerequisites: Object Oriented Programming through Java, HTML Basics.

Course Objectives:

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

Course Outcomes:

1. Design responsive web pages using HTML, CSS, Bootstrap, and JavaScript.
2. Develop backend applications using Java and Node.js.
3. Build single-page applications with React.js.
4. Create Flutter apps with custom widgets, layouts, and forms.
5. Integrate APIs, add animations, and perform UI testing in Flutter apps.

List of Experiments: Students need to implement the following experiments

1 Build a responsive website with registration, login, catalog, and cart pages using HTML, CSS3, Bootstrap, and JavaScript validation.
2 Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages
3 Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
4 Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
5 Implement JWT authentication in Node.js to create secure endpoints.
6 Create a TODO application in react with necessary components and deploy it into github.
7 Integrate a React frontend with Node.js backend to perform CRUD operations on a shared dataset (e.g., MongoDB, MySQL.)
8 Build a Django REST API for managing student data, tested using Postman.
9 Install Flutter and Dart SDK; Write Dart programs on data types, control flow, Functions, Class & Objects and collections to understand syntax and features.
10 Create a Flutter app showcasing common widgets (Text, Image, Container, Card, ListView).
11 Design a responsive UI using Row, Column, Stack, media queries, and breakpoints.
12 Implement screen navigation using Navigator and named routes.
13 Use stateful & stateless widgets; manage state using Provider or setState.
14 Design a form in Flutter with input fields, validation, and error handling (e.g., student

registration form).

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

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

B25CS09PC: DISCRETE MATHEMATICS

II Year II Semester

L	T	P	C
3	0	0	3

Course Objectives:

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

Course Outcomes:

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

UNIT - I

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

UNIT - II

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

UNIT - III

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

UNIT - IV

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

UNIT - V

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

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph. P. Grimald, Pearson education, 5^{th} edition.

2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill Publishing co.

B25CS18PC: OPERATING SYSTEMS

II Year II Semester

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.

B25CY05PC: FORMAL LANGUAGES AND AUTOMATA THEORY

L	T	P	C
3	0	0	3

II Year II Semester

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.

B25CS20PC: COMPUTER NETWORKS

II Year II Semester

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.

B25CY09PC: ALGORITHMS DESIGN AND ANALYSIS

II Year II Semester

L	T	P	C
3	0	0	3

Prerequisites: Programming for problem solving and Data Structures

Course Objectives

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

Course Outcomes

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

UNIT - I

Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation, and Little oh notation. Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.

UNIT - II

Disjoint Sets: Disjoint set operations, union and find algorithms, Priority Queue- Heaps, Heapsort Backtracking: General method, applications, n-queens problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

UNIT - III

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

UNIT - IV

Greedy method: General method, applications- Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.
Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected components, Biconnected components.

UNIT - V

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

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

TEXT BOOK:

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

REFERENCE BOOKS:

1. Design and Analysis of algorithms, Aho, Ullman, and Hopcroft, Pearson education.
2. Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C.Stein, PHI Pvt. Ltd./ Pearson Education.

- Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley and Sons.

B25MS01HS: INNOVATION AND ENTREPRENEURSHIP

L	T	P	C
2	0	0	2

II Year II Semester

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

B25CS22PC: OPERATING SYSTEMS LAB

II Year II Semester

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.

B25CS23PC: COMPUTER NETWORKS LAB

L	T	P	C
0	0	2	1

II Year II Semester

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.

B25CY10PC: ALGORITHM DESIGN AND ANALYSIS LAB

L	T	P	C
0	0	2	1

II Year II Semester

COURSE OBJECTIVES :

The objectives of this course are to:

1. Understand and analyze algorithm performance using time complexity, space complexity, and asymptotic notations.
2. Design and implement algorithms using divide and conquer techniques for searching and sorting problems.
3. Apply advanced data structures and techniques such as disjoint sets, heaps, and backtracking to solve computational problems.
4. Develop solutions for optimization problems using dynamic programming and greedy methods.
5. Implement efficient algorithms for complex problems using branch and bound techniques and graph algorithms.

COURSE OUTCOMES (COs) :

Upon successful completion of the course, the student will be able to:

1. Analyze the efficiency of algorithms using operation count and asymptotic complexity measures.
2. Design and implement divide and conquer algorithms such as Binary Search, Quick Sort, and Merge Sort.
3. Apply data structures, backtracking, and heap-based algorithms to solve combinatorial problems.
4. Develop optimal solutions using dynamic programming and greedy strategies for classical optimization problems.
5. Implement and evaluate graph algorithms and branch and bound techniques for solving real-world optimization problems.

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

LIST OF LAB EXPERIMENTS (15)

1. Write a program to analyze the time and space complexity of a given algorithm using operation count.
2. Write a program to implement Binary Search and analyze its time complexity.
3. Write a program to implement Quick Sort using divide and conquer technique.
4. Write a program to implement Merge Sort and analyze its performance.
5. Write a program to implement Disjoint Set operations using Union and Find algorithms.
6. Write a program to implement Heap data structure and perform Heap Sort.

7. Write a program to solve the N-Queens problem using backtracking technique.
8. Write a program to solve the 0/1 Knapsack problem using dynamic programming.
9. Write a program to find shortest paths between all pairs of vertices using Floyd's algorithm.
10. Write a program to solve the Travelling Salesperson Problem using dynamic programming.
11. Write a program to solve the Job Sequencing with Deadlines problem using greedy method.
12. Write a program to find the Minimum Cost Spanning Tree using Prim's algorithm.
13. Write a program to find the Minimum Cost Spanning Tree using Kruskal's algorithm.
14. Write a program to find the shortest path from a single source using Dijkstra's algorithm.
15. Write a program to solve the 0/1 Knapsack problem using Branch and Bound technique.

REFERENCES

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, *Introduction to Algorithms*, 3rd Edition, MIT Press, 2009.
2. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran, *Computer Algorithms*, 2nd Edition, Universities Press, 2008.
3. Aho, A. V., Hopcroft, J. E., and Ullman, J. D., *The Design and Analysis of Computer Algorithms*, Pearson Education, 2003.
4. Jon Kleinberg and Éva Tardos, *Algorithm Design*, Pearson Education, 2006.
5. Robert Sedgewick and Kevin Wayne, *Algorithms*, 4th Edition, Addison-Wesley, 2011.
6. Anany Levitin, *Introduction to the Design and Analysis of Algorithms*, 3rd Edition, Pearson Education, 2012.

B25CY04SD: DATA VISUALIZATION - R PROGRAMMING/ POWER BI/TABLEAU/GOOGLE CHART

II Year II Semester

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.

B25VA03HS: INDIAN KNOWLEDGE SYSTEM

L	T	P	C
1	0	0	1

II Year II Semester

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