

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY-GURAJADA VIZIANAGARAM
R26-Course Structure for B. Tech (Electronics and Communication Engineering)

B.Tech. – I Year I Semester (5 Theory 4 laboratory, 01 MNC)

S. No.	Course Category	Course Code	Title	L	T	P	Credits
1	HSM		English for Technical Communication	3	0	0	3
2	BSC		Linear Algebra and Calculus	3	0	0	3
3	BSC		Applied Chemistry	3	0	0	3
4	ESC		AI Tools and Applications	0	1	2	2
5	ESC		Fundamentals of Basic Electronics Circuits	3	0	0	3
6	HSM		English for Technical Communication laboratory	0	0	3	1.5
7	BSC		Applied chemistry Laboratory	0	0	2	1.0
8	ESC		Fundamentals of Basic Electronics Circuits Laboratory	0	0	3	1.5
9	ESC		IT and Electronic Workshop Laboratory	0	0	2	1.0
10	MNC-1		Universal Human Values (No credits)	2	0	0	0
Total				14	1	12	19.0

B.Tech (I year - II Semester) – II semester (5 Theory & 4 Laboratories & 01 MNC)

S.No.	Course Category	Course Code	Title	L	T	P	Credits
1	BSC		Differential Equations and Vector Calculus	3	0	0	3
2	BSC		Applied Physics	3	0	0	3
3	ESC		Computational Thinking and Problem Solving using 'C'	3	0	0	3
4	PCC		Network Analysis	3	0	0	3
5	ESC		Fundamentals of Quantum information Technology	2	0	0	2
6	ESC		Data Structures for networking and communication	0	2	2	3
7	PCC		Network Analysis Laboratory	0	0	3	1.5
8	ESC		Computational Thinking and Problem Solving using 'C' Laboratory	0	0	3	1.5
9	BSC		Applied Physics Laboratory	0	0	2	1.0
10	MNC-2		NSS/NCC/Scouts & Guides/Community Service/Sports (No Credits)	0	0	2	0
Total				13	2	13	21.0

hcr

L	T	Credits
3	0	3

Fundamentals Basic Electronics Circuits

Course Objectives

1. To introduce the construction, operation, and characteristics of basic solid-state devices.
2. To develop understanding of linear wave shaping circuits and their response to standard input signals.
3. To develop understanding of non-linear wave shaping circuits using diodes and their applications.
4. To introduce the fundamentals of digital electronics, number systems, and logic circuits.
5. To familiarize students with the principles of lasers and their engineering applications.
6. To introduce the principles of fiber optic communication and its advantages over conventional media.

Course Outcomes (COs): On successful completion of this course, students will be able to:

1. Explain the construction, operation, and V-I characteristics of solid-state devices such as diodes, BJTs, and FETs.
2. Analyze the response of RC, RL, and RLC circuits used for linear wave shaping (differentiation and integration) to various input waveforms.
3. Design and analyze non-linear wave shaping circuits such as clippers, clampers, and comparators using diodes.
4. Apply Boolean algebra and number systems to design and simplify basic combinational logic circuits.
5. Explain the principle of laser operation and identify applications of different laser types in engineering.
6. Explain the principle of light propagation in optical fibers and describe the basic architecture of a fiber optic communication system.

Unit I: Solid State Devices



Review of semiconductor physics: intrinsic and extrinsic semiconductors, PN junction formation, energy band diagram. PN junction diode: V-I characteristics, diode resistance, diode capacitance. Zener diode and its use as a voltage regulator. Bipolar Junction Transistor (BJT): construction, operation, CB, CE, CC configurations, input/output characteristics. Field Effect Transistor (FET): JFET construction and characteristics, introduction to MOSFET (conceptual overview).

Unit II: Linear Wave Shaping

High-pass RC circuit as a differentiator: response to sinusoidal, step, pulse, and square wave inputs. Low-pass RC circuit as an integrator: response to sinusoidal, step, pulse, and square wave inputs. Concept of time constant and its effect on output waveform. RL and RLC circuits — brief treatment. Attenuators (resistive) — introductory concept.

Unit III: Non-Linear Wave Shaping

Clipping circuits: series and shunt clippers (positive, negative, and combination/slicer circuits) using diodes; biased clippers. Clamping circuits: positive and negative clampers, biased clampers; clamping circuit theorem. Comparators: basic diode comparator circuits and applications. Applications of wave shaping circuits in signal processing and waveform generation.

Unit IV: Digital Electronics

Number systems: binary, octal, hexadecimal, and their inter-conversions; binary arithmetic. Boolean algebra: postulates, theorems, and simplification of logic expressions; Karnaugh map (K-map) simplification (up to 4 variables). Logic gates: AND, OR, NOT, NAND, NOR, XOR, XNOR — symbols, truth tables, and realization. Introduction to combinational circuits: half adder, full adder (basic design). Overview of flip-flops (SR, D — conceptual introduction).

Unit V: Laser and Fibre Optics

Laser: Characteristics of laser light, spontaneous and stimulated emission, population inversion, principle of laser action. Basic components of a laser system: active medium, pumping mechanism, optical resonator. Types of lasers: Ruby laser, He-Ne laser, semiconductor laser (construction and working — conceptual level). Applications of lasers in engineering, communication, and industry.



Fibre Optics: Principle of light propagation in optical fiber — total internal reflection. Structure of optical fiber: core, cladding, buffer. Acceptance angle and numerical aperture. Types of optical fibers: step-index and graded-index, single-mode and multi-mode fibers. Basic block diagram of a fiber optic communication system. Advantages of optical fiber communication over conventional copper-based communication.

Text Books

1. Robert L. Boylestad, Louis Nashelsky, *Electronic Devices and Circuit Theory*, 11th Edition, Pearson Education.
2. Jacob Millman, Christos Halkias, Chetan Parikh, *Millman's Electronic Devices and Circuits*, 2nd Edition, McGraw Hill. *(For wave shaping circuits — Units II & III)*
3. R. P. Jain, *Modern Digital Electronics*, 4th Edition, McGraw Hill. *(For Unit IV)*
4. Ajoy Ghatak, *Optics*, 6th Edition, McGraw Hill. *(For laser and fiber optics — Unit V)*

Reference Books

1. David A. Bell, *Electronic Devices and Circuits*, 5th Edition, Oxford University Press.
2. Donald A. Neamen, *Semiconductor Physics and Devices*, 4th Edition, McGraw Hill.
3. M. Morris Mano, Michael D. Ciletti, *Digital Design*, 6th Edition, Pearson Education.
4. John M. Senior, *Optical Fiber Communications: Principles and Practice*, 3rd Edition, Pearson Education.
5. K. Thyagarajan, Ajoy Ghatak, *Lasers: Fundamentals and Applications*, 2nd Edition, Springer.

L	T	P	Credits
0	0	3	1.5

Fundamentals Basic Electronics Circuits Laboratory

Course Objectives:

1. To verify the V-I characteristics of solid-state devices (diodes, BJTs, FETs) through experimentation.
2. To study the response of linear wave shaping circuits (differentiators, integrators) to various input waveforms.
3. To design and test non-linear wave shaping circuits such as clippers and clampers.
4. To verify the truth tables of basic and universal logic gates and realize simple combinational circuits.
5. To study the characteristics of laser sources and optical fiber-based light propagation.
6. To develop skills in using CRO, function generators, and regulated power supplies for circuit testing.

Course Outcomes (COs): On successful completion of this laboratory course, students will be able to:

1. Plot and interpret V-I characteristics of PN junction diode, Zener diode, BJT, and FET.
2. Design and test RC differentiator and integrator circuits, and analyze their output for different input waveforms and time constants.
3. Design and test clipping and clamping circuits and verify their output waveforms against theoretical predictions.
4. Verify truth tables of logic gates and realize basic combinational circuits (half adder/full adder).
5. Demonstrate the characteristics of a laser source and determine numerical aperture/propagation loss in an optical fiber.
6. Use CRO, function generators, and multimeters to measure and troubleshoot analog and digital circuits.



List of Experiments

1. Plotting V-I characteristics of a PN junction diode (forward and reverse bias).
2. Plotting V-I characteristics of a Zener diode and study of Zener diode as a voltage regulator.
3. Plotting input and output characteristics of a BJT in Common Emitter (CE) configuration.
4. Plotting drain and transfer characteristics of a JFET. | Design of an RC differentiator circuit and study of output response for square wave input at different time constants.
5. Design of an RC integrator circuit and study of output response for square wave input at different time constants.
6. Design and testing of series/shunt diode clipping circuits (positive, negative, and combination clippers).
7. Design and testing of positive and negative clamping circuits and verification of clamping theorem.
8. Verification of truth tables of basic logic gates (AND, OR, NOT) using ICs. |
9. Verification of truth tables of universal gates (NAND, NOR) and realization of basic gates using universal gates.
10. Verification of truth tables of XOR and XNOR gates and study of their applications.
11. Design and realization of a Half Adder and Full Adder circuit using logic gates and verification of truth table.
12. Study of laser characteristics — determination of wavelength of a laser source using diffraction grating. |
13. Determination of numerical aperture of a given optical fiber. |
14. Study of propagation/bending loss in an optical fiber and study of a basic fiber optic analog/digital link (transmitter-receiver kit). |

Equipment / Components Required

- Regulated DC power supplies, function generators, CRO/DSO, digital multimeters.
- PN junction diodes, Zener diodes, BJTs (CE configuration kits), JFETs, breadboards, resistors, capacitors.
- Digital IC trainer kit; logic gate ICs (74LS series — AND, OR, NOT, NAND, NOR, XOR, XNOR).
- He-Ne laser source or laser pointer, diffraction grating, optical bench.



- Optical fiber cables (single-mode/multi-mode), fiber optic analog/digital transmitter-receiver trainer kit.

Reference Books

1. Robert L. Boylestad, Louis Nashelsky, *Electronic Devices and Circuit Theory*, 11th Edition, Pearson Education. *(Reference for Units I–III experiments)*
2. R. P. Jain, *Modern Digital Electronics*, 4th Edition, McGraw Hill. *(Reference for Unit IV experiments)*
3. John M. Senior, *Optical Fiber Communications: Principles and Practice*, 3rd Edition, Pearson Education. *(Reference for Unit V experiments)*
4. David A. Bell, *Electronic Devices and Circuits Lab Manual*, 5th Edition, Oxford University Press.



L	T	P	Credits
0	0	2	1

IT and Electronics Workshop

Course Objectives:

1. To familiarize students with computer hardware components, assembly, and basic troubleshooting.
2. To provide hands-on exposure to operating system installation and essential productivity software.
3. To introduce basic networking concepts and internet/cloud usage for engineering work.
4. To familiarize students with electronic components, their identification, testing, and specifications.
5. To develop practical skills in soldering, PCB handling, and simple circuit assembly.
6. To build confidence in using multimeters and basic electronic test equipment.

Course Outcomes (COs)

On successful completion of this course, students will be able to:

1. Identify computer hardware components and assemble/disassemble a desktop PC. |
2. Install operating systems and configure basic system/network settings. |
3. Use office productivity tools and internet-based resources effectively for technical documentation. |
4. Identify, test, and interpret specifications of common electronic components (resistors, capacitors, diodes, transistors, ICs). |
5. Perform soldering/de-soldering and assemble simple electronic circuits on breadboard/PCB. |
6. Use a multimeter and basic test equipment to measure voltage, current, resistance, and continuity, and troubleshoot simple circuit faults. |

Part A: IT Workshop

List of Experiments

1. Identification of PC components (motherboard, CPU, RAM, SMPS, storage devices, ports) and study of specifications.
2. Assembling and disassembling a desktop computer (hardware setup).



3. Installation of an operating system (Windows/Linux) and installation of device drivers.
4. Study of basic BIOS/UEFI settings and system configuration.
5. Configuration of a local area network (LAN) — IP addressing, sharing files/printers over a network.
6. Preparation of technical documents using word processing software — formatting a technical report with headers, references, and equations.
7. Preparation of a technical presentation using presentation software, incorporating charts and diagrams.
8. Basic spreadsheet operations — data entry, formulas, charts, for tabulating experimental/observation data.

Part B: Electronics Workshop

List of Experiments

1. Identification of electronic components — resistors (colour code), capacitors, inductors — and their testing using a multimeter.
2. Identification and testing of semiconductor devices — diodes, LEDs, BJTs — using a multimeter.
3. Study of soldering tools and practice of soldering/de-soldering techniques on a general-purpose PCB.
4. Study of a digital multimeter and CRO — measurement of voltage, current, resistance, frequency, and continuity testing.
5. Assembly of a simple electronic circuit (e.g., LED blinking / battery charger indicator) on breadboard.
6. Soldering and assembly of the simple electronic circuit (from Exp. 13) on a PCB, and functional testing.
7. Study of PCB design basics and layout using PCB design software (e.g., Eagle/KiCad) — introductory exercise.

Equipment / Tools / Software Required

1. Desktop PC components/kits for assembly, screwdriver sets, anti-static wrist bands.
2. Operating system installation media, LAN cables, switch/router (for networking demo).
3. Office productivity suite (word processor, spreadsheet, presentation software).
4. Electronic components: resistors, capacitors, diodes, LEDs, transistors, breadboards, general-purpose PCBs.



5. Soldering iron, soldering wire, flux, de-soldering pump.
6. Digital multimeter, CRO/DSO, regulated DC power supply.
7. PCB design software (KiCad/Eagle — open source preferred).

Text Books:

1. P. Abdul Khader Jilani, *IT Workshop*, Universities Press.
 2. Anita Goel, *Computer Fundamentals*, Pearson Education.
 3. Paul Horowitz, Winfield Hill, *The Art of Electronics*, 3rd Edition, Cambridge University Press. *(Basic component and soldering reference) *
 4. K. A. Navas, *Electronics Lab Manual*, Vol. 1, PHI Learning.
 5. Manufacturer's datasheets for standard components (resistors, diodes, transistors) as reference during identification exercises.
- 

L	T	P	Credits
3	0	0	3

Network Analysis

Course Objectives

- To introduce basic circuit elements, sources, and fundamental laws governing electrical networks.
- To develop proficiency in applying network theorems for simplifying and analyzing complex circuits.
- To analyze the transient and steady-state behavior of RL, RC, and RLC circuits.
- To familiarize students with network topology (graph theory) as applied to electrical circuits.
- To introduce two-port network parameters and their application in circuit modeling.
- To build a strong analytical foundation for competitive examinations such as GATE, ESE, and other PSU recruitment exams.

Course Outcomes (COs): On successful completion of this course, students will be able to:

CO1 Apply Kirchhoff's laws and network theorems to solve DC and AC circuits.

CO2 Analyze the transient response of first- and second-order circuits using classical and Laplace transform methods.

CO3 Evaluate the sinusoidal steady-state response of circuits, including resonance phenomena in series and parallel RLC circuits.

CO4 Construct network graphs and apply topological methods (incidence matrix, tie-set, cut-set) to solve networks.

CO5 Determine two-port network parameters (Z , Y , $ABCD$, h) and analyze interconnected two-port networks.

CO6 Solve numerical problems of the type asked in GATE, ESE, and PSU examinations within a time-bound format.



Unit I: Basic Circuit Concepts and Network Theorems

Circuit elements: active and passive, linear and nonlinear, bilateral and unilateral elements; independent and dependent sources; source transformation. Kirchhoff's voltage and current laws. Mesh analysis and nodal analysis (including supermesh and supernode). Star-delta transformation.

Network theorems: Superposition theorem, Thevenin's theorem, Norton's theorem, Maximum Power Transfer theorem, Reciprocity theorem, Millman's theorem, Substitution theorem, Tellegen's theorem — applied to DC and AC circuits.

Unit II: Transient Analysis of Electrical Circuits

Introduction to transient and steady-state response. Behavior of R, L, C elements under switching conditions. Transient analysis of RL, RC, and RLC series/parallel circuits for DC excitation using differential equations and Laplace transform. Initial and final conditions, initial value and final value theorems. Concept of time constant. Transient response with sinusoidal excitation (brief treatment).

Unit III: Sinusoidal Steady-State Analysis and Resonance

AC fundamentals: phasor representation, impedance, admittance. Steady-state analysis of RLC circuits using phasor techniques. Power in AC circuits: active, reactive, apparent power, power factor. Series and parallel resonance: resonant frequency, bandwidth, quality factor (Q-factor), selectivity. Locus diagrams. Magnetically coupled circuits: self and mutual inductance, coefficient of coupling, dot convention, analysis of coupled circuits.

Unit IV: Network Topology (Graph Theory)

Graph of a network: definitions — tree, co-tree, twigs, links, tie-set, cut-set. Incidence matrix, tie-set matrix, cut-set matrix. Formulation of network equations using loop and node methods based on graph theory. Duality in networks.

Unit V: Two-Port Networks and Laplace Transform Applications



Two-port network parameters: Z (impedance), Y (admittance), $ABCD$ (transmission), h (hybrid) parameters; inter-relationships and conversions. Interconnection of two-port networks (series, parallel, cascade). Image impedance and characteristic impedance (introductory treatment).

Application of Laplace transforms to circuit analysis: transformed circuits (s -domain), network functions, poles and zeros, concept of transfer function. Introduction to filter fundamentals using network functions.

Text Books

1. M. E. Van Valkenburg, Network Analysis, 3rd Edition, Prentice Hall of India.
2. D. Roy Choudhury, Networks and Systems, 2nd Edition, New Age International Publishers.
3. William H. Hayt, Jack E. Kemmerly, Steven M. Durbin, Engineering Circuit Analysis, 8th Edition, McGraw Hill.

Reference Books

- A. Sudhakar, Shyammohan S. Palli, Circuits and Networks: Analysis and Synthesis, 5th Edition, McGraw Hill.
1. Charles K. Alexander, Matthew N. O. Sadiku, Fundamentals of Electric Circuits, 6th Edition, McGraw Hill.
2. Franklin F. Kuo, Network Analysis and Synthesis, 2nd Edition, Wiley India.
3. J. David Irwin, R. Mark Nelms, Basic Engineering Circuit Analysis, 11th Edition, Wiley.
4. Previous years' GATE EC/EE solved question papers (Network Theory section) — for practice and exam orientation.



L	T	P	Credits
2	0	0	2

Fundamentals of Quantum Information Technology

Course Objectives

1. To introduce the basic postulates of quantum mechanics relevant to information processing.
2. To familiarize students with the concept of qubits and their representation using Dirac notation and the Bloch sphere.
3. To develop understanding of quantum gates, circuits, and elementary quantum algorithms.
4. To introduce the principles of quantum entanglement, superposition, and measurement, and their role in information technology.
5. To provide an overview of quantum communication and quantum cryptography techniques.
6. To create awareness of current quantum computing hardware platforms and emerging applications.

Course Outcomes (COs): On successful completion of this course, students will be able to:

1. Explain the basic postulates of quantum mechanics and represent quantum states using Dirac notation.
2. Describe qubits, superposition, and the Bloch sphere representation, and distinguish them from classical bits.
3. Analyze single- and multi-qubit quantum gates and construct simple quantum circuits.
4. Explain quantum entanglement and its role in phenomena such as teleportation and superdense coding.
5. Describe the working principles of elementary quantum algorithms (Deutsch-Jozsa, Grover's, Shor's — conceptual level).
6. Describe the fundamentals of quantum key distribution and compare quantum **information platforms/technologies at an introductory level.**

Unit I: Introduction to Quantum Mechanics for Information Processing



Limitations of classical information theory. Basic postulates of quantum mechanics: state space, Dirac (bra-ket) notation, superposition principle, quantum measurement, unitary evolution. Linear algebra prerequisites: complex vector spaces, Hilbert space, inner product, unitary and Hermitian operators (review). Concept of observables and eigenvalues in a measurement context.

Unit II: Qubits and Quantum States

Classical bit vs. qubit. Single-qubit states and Bloch sphere representation. Multi-qubit systems, tensor products, and composite state spaces. Quantum superposition and its physical significance. No-cloning theorem. Introduction to quantum entanglement: Bell states and their properties.

Unit III: Quantum Gates and Circuits

Single-qubit gates: Pauli-X, Y, Z, Hadamard, Phase, and rotation gates. Multi-qubit gates: CNOT, SWAP, Toffoli. Quantum circuit representation and circuit composition rules. Reversibility of quantum gates. Quantum measurement in circuits. Simple circuit design exercises (state preparation, Bell-state generation).

Unit IV: Quantum Algorithms and Entanglement Applications

Quantum parallelism, concept of quantum speedup. Deutsch and Deutsch-Jozsa algorithm (conceptual overview). Grover's search algorithm — principle and significance. Shor's factoring algorithm — conceptual overview and implications for cryptography. Applications of entanglement: quantum teleportation and superdense coding (conceptual treatment with circuit description).

Unit V: Quantum Communication and Emerging Platforms

Principles of quantum key distribution (QKD): BB84 protocol (conceptual overview). Quantum cryptography and its advantage over classical cryptography. Overview of quantum computing hardware platforms: superconducting qubits, trapped ions, photonic qubits (introductory comparison). Overview of current quantum computing tools/simulators (e.g., IBM Qiskit, cloud-based quantum simulators) and their role in learning/experimentation. Emerging applications and industry trends in quantum information technology.

Text Books:



1. Michael A. Nielsen, Isaac L. Chuang, *Quantum Computation and Quantum Information*, 10th Anniversary Edition, Cambridge University Press.
2. Eleanor Rieffel, Wolfgang Polak, *Quantum Computing: A Gentle Introduction*, MIT Press.
3. N. David Mermin, *Quantum Computer Science: An Introduction*, Cambridge University Press.

Reference Books

1. David McMahon, *Quantum Computing Explained*, Wiley-IEEE Computer Society Press.
2. Chris Bernhardt, *Quantum Computing for Everyone*, MIT Press.
3. Noson S. Yanofsky, Mirco A. Mannucci, *Quantum Computing for Computer Scientists*, Cambridge University Press.
4. Vlatko Vedral, *Introduction to Quantum Information Science*, Oxford University Press.
5. IBM Qiskit Textbook (open-access online resource) — for hands-on circuit-building exercises and simulation practice.



L	T	P	Credits
0	0	3	1.5

Network Analysis Laboratory

Objectives:

1. To verify fundamental network theorems experimentally on DC and AC circuits.
2. To study the transient and steady-state response of RL, RC, and RLC circuits.
3. To analyze resonance phenomena and coupled circuit behavior practically.
4. To determine two-port network parameters through hardware experimentation.
5. To familiarize students with circuit simulation software (e.g., MULTISIM/PSpice/MATLAB) as a verification and design tool.

Outcomes:

On completion, students will be able to verify theoretical network concepts through hardware and simulation, interpret waveforms/measurements, and correlate experimental results with analytical calculations — skills directly useful for GATE/PSU-style application questions.

List of Experiments:

1. Study of components of a circuit and Verification of KCL and KVL.
2. Verification of mesh and nodal analysis for AC circuits
3. Verification of Superposition, Thevenin's & Norton theorems for AC circuits
4. Verification of maximum power transfer theorem for AC circuits
5. Verification of Tellegen's theorem for two networks of the same topology.
6. Study of DC transients in RL, RC and RLC circuits
7. To study frequency response of various 1st order RL & RC networks
8. To study the transient and steady state response of a 2nd order circuit by varying its various parameters and studying their effects on responses
9. Find the Q Factor and Bandwidth of a Series and Parallel Resonance circuit.

h

10. Determination of open circuit (Z) and short circuit (Y) parameters 11. Determination of hybrid (H) and transmission (ABCD) parameters 12. To measure two port parameters of a twin-T network and study its frequency response.

Hardware Requirements: Regulated Power supplies, Analog/Digital Function Generators, Digital Multimeters, Decade Resistance Boxes/Rheostats, Decade Capacitance Boxes, Ammeters (Analog or Digital), Voltmeters (Analog or Digital), Active & Passive Electronic Components

Software Required: MULTISIM / PSpice / MATLAB-Simulink (for Experiments 12 and 15, and for optional verification of hardware experiments).

Text Book:

1. M. E. Van Valkenburg, *Network Analysis*, 3rd Edition, Prentice Hall of India.
2. Sanjay Sharma, *Electric Circuit Analysis Laboratory Manual* — for standard lab procedure formats and observation table templates.
3. A.Sudhakar, Shyammohan S. Palli, *Circuits and Networks: Analysis and Synthesis*, 5th Edition, McGraw Hill. *(Reference for two-port parameter and topology experiments)



L	T	P	Credits
0	2	2	3

Data Structures for Networking and Communication

Course Objectives:

The course aims to:

1. Understand the fundamental concepts of data structures and their applications.
2. Analyze and implement linear and non-linear data structures.
3. Develop knowledge of computer networks and data communication principles.
4. Understand network protocols, routing, and network security.
5. Apply data structures and networking concepts to solve real-world problems.

Course Outcomes (COs)

After successful completion of the course, students will be able to:

- CO1: Explain fundamental data structures and analyze their time and space complexity.
- CO2: Implement linear and non-linear data structures for solving computational problems.
- CO3: Describe the basic concepts, models, and transmission techniques of computer networks.
- CO4: Analyze networking protocols, routing algorithms, and transport-layer services.
- CO5: Apply data structures and network communication concepts to develop simple real-world applications.

Unit-wise Syllabus

UNIT-I: Introduction to Data Structures and Linear Data Structures

Introduction to data structures: Definition, types, and applications, Algorithm analysis: Time and space complexity, Big-O notation,



Arrays: Operations and applications, Linked lists: Singly, doubly, and circular linked lists., Stacks: Operations, applications, and expression evaluation, Queues: Circular queues, priority queues, and applications.

UNIT-II: Non-Linear Data Structures and Searching Trees:

Terminology and binary trees, Binary Search Trees (BST): Operations and applications, Tree traversals: Inorder, preorder, and postorder, Heaps and priority queues, Graphs: Representation using adjacency matrix and adjacency list, Graph traversal: Breadth First Search (BFS) and Depth First Search (DFS), Searching techniques: Linear search and binary search.

UNIT-III: Fundamentals of Network Communication: Introduction to data communication and computer networks: Components and characteristics of data communication, Network types: LAN, MAN, WAN, and PAN, Network topologies and transmission media, OSI and TCP/IP reference models.

UNIT-IV: Data Link and Network Layers: Data link layer: Framing, error detection, and error correction, Flow control and error control protocols, Medium Access Control (MAC): Ethernet and IEEE 802.11, Networking devices: Repeaters, bridges, switches, routers, and gateways, Network layer functions and addressing, IPv4, IPv6, subnetting, and CIDR, Routing algorithms: Distance vector and link state routing.

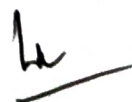
UNIT-V: Transport Layer, Application Layer, and Network Security: Transport layer services and protocols, Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), Flow control and congestion control, Application layer protocols: DNS, HTTP, FTP, and email.

Textbooks

1. Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed, Fundamentals of Data Structures in C, Universities Press.
2. Behrouz A. Forouzan, Data Communications and Networking, 5th Edition, McGraw Hill Education.
3. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach, 8th Edition, Pearson.

Reference Books

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson.
2. Robert Lafore, Data Structures and Algorithms in Java, 2nd Edition, Sams Publishing.



3. Andrew S. Tanenbaum and David J. Wetherall, Computer Networks, 5th Edition, Pearson.
4. Larry L. Peterson and Bruce S. Davie, Computer Networks: A Systems Approach, 5th Edition, Morgan Kaufmann.
5. William Stallings, Data and Computer Communications, 10th Edition, Pearson.



L	T	P	Credits
3	0	0	3

Computational Thinking and Problem-Solving Using 'C'

Course Objectives

The course aims to:

1. Understand computational thinking and fundamental problem-solving techniques.
2. Develop algorithms and flowcharts for solving engineering problems.
3. Learn the fundamentals of C programming, including operators and control structures.
4. Apply arrays, strings, functions, pointers, and structures in C programming.
5. Develop efficient and modular C programs for solving real-world engineering problems.

Course Outcomes (COs)

After successful completion of the course, students will be able to:

- CO1: Apply computational thinking techniques, algorithms, and flowcharts to solve basic engineering problems.
- CO2: Develop C programs using data types, operators, expressions, and control structures.
- CO3: Implement arrays, strings, and searching and sorting techniques using C.
- CO4: Apply functions, recursion, pointers, and dynamic memory allocation in C programs.
- CO5: Develop modular C programs using structures and file handling to solve engineering problems.

Unit-wise Syllabus

UNIT-I: COMPUTATIONAL THINKING AND PROBLEM-SOLVING FUNDAMENTALS:

Introduction to computational thinking: Decomposition, pattern recognition, abstraction, and algorithmic thinking, Problem definition and analysis. , Problem-solving strategies and logical reasoning.

W

Algorithms: Characteristics, representation, and analysis., Flowcharts and pseudocode, Top-down design and stepwise refinement, Introduction to computers: CPU, memory, input/output, and storage.

UNIT-II: INTRODUCTION TO C PROGRAMMING AND CONTROL STRUCTURES: History and features of C programming, Structure of a C program and program development process, Character set, tokens, keywords, identifiers, and constants, Data types, variables, and type conversion, Input and output functions, Operators and expressions: Arithmetic, relational, logical, assignment, and bitwise operators, Operator precedence and associativity, Decision-making statements: if, if-else, nested if, and switch, Looping statements: for, while, and do-while. Jump statements: break, continue, and goto.

UNIT-III: ARRAYS, STRINGS, SEARCHING, AND SORTING: Arrays: One-dimensional and multidimensional arrays, Array operations and applications, Matrix operations: Addition and multiplication, Character arrays and strings, String input/output and standard string-handling functions.

- Searching techniques: Linear search and binary search.
- Sorting techniques: Bubble sort, selection sort, and insertion sort.
- Basic engineering applications using arrays and strings.

UNIT-IV: FUNCTIONS, RECURSION, AND POINTERS: **Functions:** Declaration, definition, and function calls, Parameter passing and return values, Library functions and user-defined functions, Scope and storage classes, Recursion and recursive problem-solving techniques, **Pointers:** Declaration, initialization, and pointer arithmetic, Pointers and arrays, Passing pointers to functions. Dynamic memory allocation: malloc(), calloc(), realloc(), and free().

UNIT-V: STRUCTURES, UNIONS, AND FILE HANDLING:

Structures: Declaration, initialization, and accessing structure members, Arrays of structures and nested structures, Pointers to structures, Unions and enumerations, Introduction to preprocessor directives and macros, File handling: Opening, reading, writing, and closing files, Text and binary files, Error handling in file operations, Problem-solving applications using structures and file handling.

Textbooks

1. Jeri R. Hanly and Elliot B. Koffman, Problem Solving and Program Design in C, Pearson.

2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw Hill.
3. E. Balagurusamy, Programming in ANSI C, McGraw Hill Education.

Reference Books

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, 2nd Edition, Prentice Hall of India.
 2. R. G. Dromey, How to Solve It by Computer, Pearson Education.
 3. Stephen G. Kochan, Programming in C, 4th Edition, Pearson.
 4. Yashavant Kanetkar, Let Us C, BPB Publications.
 5. S. Sridhar, S. S. Sridhar, and V. M. Hariharan, Problem Solving and Programming in C: A Computational Thinking Approach, Pearson, 2025.
- 

L	T	P	Credits
0	0	3	1.5

Computational Thinking and Problem-Solving Using 'C' Lab

Course Objectives

The course aims to:

1. Develop computational thinking and logical problem-solving skills through practical programming.
2. Implement algorithms and flowcharts using the C programming language.
3. Gain hands-on experience with control structures, arrays, strings, and functions.
4. Apply pointers, recursion, and dynamic memory allocation to solve programming problems.
5. Develop modular C programs using structures and file handling for engineering applications.

Course Outcomes (COs)

After successful completion of the course, students will be able to:

- CO1: Develop algorithms, flowcharts, and C programs to solve basic computational problems.
- CO2: Implement C programs using operators, decision-making, and looping statements.
- CO3: Apply arrays, strings, searching, and sorting techniques in C programs.
- CO4: Develop C programs using functions, recursion, pointers, and dynamic memory allocation.
- CO5: Implement structures and file-handling techniques to solve engineering-related problems.

LIST OF EXPERIMENTS

The following experiments are proposed for the Computational Thinking and Problem-Solving Using 'C' Laboratory.

UNIT-I: COMPUTATIONAL THINKING AND PROBLEM-SOLVING

1. Develop algorithms and flowcharts for solving basic mathematical and engineering problems, such as finding the largest of three numbers and calculating simple and compound interest.
2. Write C programs to implement basic problem-solving techniques, including unit conversion, temperature conversion, and electricity bill calculation.

UNIT-II: C PROGRAMMING AND CONTROL STRUCTURES

3. Write C programs to demonstrate the use of data types, operators, expressions, and formatted input/output.



4. Develop C programs using decision-making statements to solve problems such as finding the roots of a quadratic equation and determining grade classifications.
5. Write C programs using looping statements to generate multiplication tables, calculate factorials, and check whether a number is prime or a palindrome.

UNIT-III: ARRAYS, STRINGS, SEARCHING, AND SORTING

6. Write C programs to perform one-dimensional and two-dimensional array operations, including matrix addition and multiplication.
7. Develop C programs to perform string operations such as finding string length, copying, concatenation, comparison, and reversing a string.
8. Implement linear search and binary search techniques using C.
9. Implement bubble sort, selection sort, and insertion sort techniques using C.

UNIT-IV: FUNCTIONS, RECURSION, AND POINTERS

10. Write C programs using user-defined functions to solve mathematical and engineering problems.
11. Develop recursive C programs to calculate factorials, Fibonacci series, and the greatest common divisor (GCD) of two numbers.
12. Write C programs to demonstrate pointers, pointer arithmetic, and the relationship between pointers and arrays.
13. Develop a C program to demonstrate dynamic memory allocation using malloc(), calloc(), realloc(), and free().

UNIT-V: STRUCTURES, UNIONS, AND FILE HANDLING

14. Write C programs to store and process student or employee details using structures and arrays of structures.
15. Develop a C program to demonstrate unions, enumerations, and preprocessor directives.
16. Write C programs to create, read, write, append, and update data in text files.
17. Develop a C program to store and retrieve structured records using file handling.
18. Develop a mini-project using C programming concepts to solve a basic engineering or real-world problem.



Suggested Laboratory Requirements

- Computer systems with a C compiler.
- C programming environment such as GCC, Code::Blocks, or Visual Studio Code.
- Flowchart and algorithm development tools.

Note: Any 12–15 experiments may be selected by the concerned Board of Studies based on the allotted laboratory hours and institutional requirements.

Textbooks

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw Hill.
2. E. Balagurusamy, Programming in ANSI C, McGraw Hill Education.
3. Jeri R. Hanly and Elliot B. Koffman, Problem Solving and Program Design in C, Pearson.

Reference Books

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, 2nd Edition, Prentice Hall of India.
 2. Stephen G. Kochan, Programming in C, 4th Edition, Pearson.
 3. Yashavant Kanetkar, Let Us C, BPB Publications.
 4. R. G. Dromey, How to Solve It by Computer, Pearson Education.
 5. K. N. King, C Programming: A Modern Approach, 2nd Edition, W. W. Norton & Company.
- 