Study Scheme & Syllabus of

Undergraduate Degree Program

# Bachelor of Technology

in

# Electronics and Computer Science

(3rd & 4th semester)

Batch 2024 Onwards

By

Board of Studies

Electronics & Communication Engineering

# I K Gujral Punjab Technical University, Jalandhar

## DEPARTMENT VISION

To produce tomorrow's **entrepreneur** and **technology leaders** for the domains of **electronics engineering** and **information technology** those can give **technical solutions** to the **real-world problems** and capable of handling **global challenges** at large.

## DEPARTMENT MISSION

M1. To produce technocrats, researchers and entrepreneurs with inherent human values who can tackle challenges of professional career.

M2. To deliver comprehensive knowledge of electronics engineering and computer science domain through outcome-based curriculum.

M3. To provide state-of-the-art teaching learning infrastructure and conducive environment for practical realization of theoretical concepts.

| **Program Educational Objectives:** At the end of the Program, the student will be able to: - | |
|---|---|
| PEO1 | To solve real-world problems and excel in their professional career based on their technical knowledge in the domains of electronics engineering and information technology. |
| PEO2 | To acquire leading positions in the industry, assume research as career and dare to consider startup challenges with their innovative thinking and practical skills. |

| **Program Outcomes:** At the end of the Program, the student will be able to: - | |
|---|---|
| PO1 | **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems. |
| PO2 | **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences. |
| PO3 | **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations. |
| PO4 | **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions. |
| PO5 | **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations. |

| | |
|---|---|
| PO6 | **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations. |
| PO7 | **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development. |
| PO8 | **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. |
| PO9 | **Individual and teamwork:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings. |
| PO10 | **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. |
| PO11 | **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projectsand in multidisciplinary environments |
| PO12 | **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. |

| **Program Specific Outcomes:** At the end of the Program, the student will be able to: - | |
|---|---|
| PSO1 | **VLSI Design**: To progress in the domains of VLSI design and development of integrated circuits. |
| PSO2 | **Information Technology**: To apply technical knowledge of electronics and computer engineering to create solutions for real-world problems. |
| PSO3 | **Innovation and Design Ability**: Innovative thinking and ability to design and/or improve products and/or systems for the society and industry for better utilization, human safety and reduced cost. |

| **Semester** | **Third (3rd)** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Course Code** | **Course Name/Title** | **Course Type** | **Lord Allocation** | | | | **Marks Distribution** | | **Total Marks** | **Credit / NC** |
| | | | **Lecture** | **Tutorial** | **Practical** | **Total** | **Internal** | **External** | | |
| BTEC321-18 | 4-Week Institutional Summer Training* | Training | 0 | 0 | 4 | 4 | 60 | 40 | 100 | 0 |
| BTCS301-18 | Data Structure & Algorithms | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTCS303-18 | Data Structure & Algorithms Lab | Practical | 0 | 0 | 4 | 4 | 30 | 20 | 50 | 2 |
| BTEC302-18 | Digital System Design | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTEC312-18 | Digital System Design Laboratory | Practical | 0 | 0 | 2 | 2 | 30 | 20 | 50 | 1 |
| BTEC301-18 | Electronic Devices | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTEC311-18 | Electronic Devices Laboratory | Practical | 0 | 0 | 2 | 2 | 30 | 20 | 50 | 1 |
| BTCS302-18 | Object Oriented Programming | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTCS304-18 | Object Oriented Programming Lab | Practical | 0 | 0 | 4 | 4 | 30 | 20 | 50 | 2 |
| HSMC102-18 | Philosophy | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| HSMC101-18 | Development of Societies | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BMPD331-18 | Mentoring & Professional Development | Practical | 0 | 0 | 2 | 2 | S/US | 0 | S/US | 0 |
| **Total** | | | **18** | **0** | **18** | **36** | **420** | **480** | **900** | **24** |

NC = Non-Credit * 4-Week Institutional Training will be conducted by the institute during Summer Vacation after Second semester.

| Semester | Fourth (4th) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Course Code | Course Name/Title | Course Type | Lord Allocation | | | | Marks Distribution | | Total Marks | Credit / NC |
| | | | Lecture | Tutorial | Practical | Total | Internal | External | | |
| BTEC402-18 | Microprocessor and Microcontroller | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTEC412-18 | Microprocessor and Microcontroller Laboratory | Practical | 0 | 0 | 2 | 2 | 30 | 20 | 50 | 1 |
| BTCS403-18 | Design & Analysis of Algorithms | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTCS405-18 | Design & Analysis of Algorithms Lab | Practical | 0 | 0 | 4 | 4 | 30 | 20 | 50 | 2 |
| BTEC401-18 | Analog Circuits | Theory | 3 | 1 | 0 | 4 | 40 | 60 | 100 | 4 |
| BTEC411-18 | Analog Circuits Laboratory | Practical | 0 | 0 | 2 | 2 | 30 | 20 | 50 | 1 |
| BTCS402-18 | Operating System | Theory | 3 | 0 | 0 | 3 | 40 | 60 | 100 | 3 |
| BTCS404-18 | Operating System Lab | Practical | 0 | 0 | 4 | 4 | 30 | 20 | 50 | 2 |
| BTCS-401-18 | Discrete Mathematics | Theory | 3 | 1 | 0 | 4 | 40 | 60 | 100 | 4 |
| BTECS411-24 | Simulation Software Lab | Practical | 0 | 0 | 2 | 2 | S/US | 0 | S/US | 1 |
| | | **Total** | **15** | **2** | **14** | **31** | **320** | **380** | **700** | **24** |

# Third Semester

| L | T | P | BTEC321-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 0 | 4-Week Institutional Training | 0 |

Students of Electronics and Computer Science will undergo 4-Week Institutional Training during summer vacations after end of second semester examinations. This training should give exposure to the practical aspects of discipline.

The training outline may include the following topics depending upon the available time and background of students:

1. Object-Oriented Programming
2. LaTeX for report writing
3. Web Technologies – HTML, CSS, JavaScript, MySQL
4. PHP for web programming
5. Introduction to electronic components and datasheets
6. Using multimeter, function generators and DSOs

| L | T | P | BTCS301-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 0 | Data Structure & Algorithms | 3 |

**Course Outcomes:** The students will be able to:

CO-1. For a given algorithm student will be able to analyze the algorithms to determine the time and computation complexity and justify the correctness.

CO-2. Students will be able to handle operations like searching, insertion, deletion, traversing on various Data Structures and determine time and computational complexity

CO-3. Students will be able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in terms of Space and Time complexity.

CO-4. Students will be able to choose appropriate Data Structure as applied to specific problem definition.

CO-5. Demonstrate the reusability of Data Structures for implementing complex iterative problems.

**Unit1: Introduction**

Basic Terminologies: Elementary Data Organizations, Data Structure Operations: Insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space tradeoff.
**Searching:** Linear Search and Binary Search Techniques and their complexity analysis.

**Unit 2: Stack sand Queues**

ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queues: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.

**Unit 3**: **Linked Lists**

Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: All operations their algorithms and the complexity analysis.
**Trees:** Basic Tree Terminologies, Different types of Trees :Binary Tree, Threaded Binary Tree, Binary SearchTree,AVLTree;Treeoperationsoneachofthetreesandtheiralgorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.

**Unit 4:Sorting and Hashing**

Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.

**Unit5: Graph**

Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

**Suggested Books:**

1. Classic Data Structures ,Samanta and Debasis,2$^{nd}$edition,PHIpublishers.
2. Fundamentals of Data Structures, Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.
3. Data Structures with C (Schaum'sOutlineSeries)”,SeymourLipschutz,1st edition, McGraw Hill Education.
4. Algorithms, Data Structures, and Problem Solving with C++,Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company.
5. How to Solve it by Computer,2$^{nd}$ Impression by R.G.Dromey, Pearson Education.

| L | T | P | BTCS303-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 4 | Data Structure & Algorithms Lab | 2 |

**Course Outcomes:** The students will be able to:

CO-1. Improve practical skills in designing and implementing basic linear data structure algorithms.
CO-2. Improve practical skills in designing and implementing Non-linear data structure algorithms.
CO-3. Use Linear and Non-Linear data structures to solve relevant problems.
CO-4. Choose appropriate Data Structure as applied to specific problem definition.
CO-5. Implement Various searching algorithms and become familiar with their design methods.

**List of Experiments:**

Task 1. Write a program to insert a new element at end as well as at a given position in an array.
Task 2. Write a program to delete an element from a given whose value is given or whose position is given.
Task 3. Write a program to find the location of a given element using Linear Search.
Task 4. Write a program to find the location of a given element using Binary Search.
Task 5. Write a program to implement push and pop operations on a stack using linear array.
Task 6. Write a program to convert an infix expression to a postfix expression using stacks.
Task 7. Write a program to evaluate a postfix expression using stacks.
Task 8. Write a recursive function for Tower of Hanoi problem.
Task 9. Write a program to implement insertion and deletion operations in a queue using linear array.
Task 10. Write a menu driven program to perform the following insertion operation at beginning

a. Insertion at end
b. Insertion after a given node
c. Traversing a linked list

Task 11. Write a menu driven program to perform the following deletion operations in a single linked list:

a. Deletion at beginning
b. Deletion at end
c. Deletion after a given node

Task 12. Write a program to implement push and pop operations on a stack using linked list.
Task 13. Write a program to implement push and pop operations on a queue using linked list.
Task 14. Program to sort an array of integers in ascending order using bubble sort.
Task 15. Program to sort an array of integers in ascending order using selection sort.
Task 16. Program to sort an array of integers in ascending order using insertion sort.
Task 17. Program to sort an array of integers in ascending order using quick sort.
Task 18. Program to traverse a Binary search tree in Pre-order, In-order and Post-order.
Task 19. Program to traverse graphs using BFS.
Task 20. Program to traverse graphs using DFS.

| L | T | P | BTCS302-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Digital System Design | 3 |

**Course Outcomes:** At the end of this course student will demonstrate the ability to:

CO-1. Apply concepts of Boolean algebra for handling logical expressions.
CO-2. Understand working and realization of combinational circuits.
CO-3. Understand working flip-flops and use them in designing sequential circuits.
CO-4. Understand fundamental concepts of logic families and architectural of programmable devices.
CO-5. Use HDL programming tool for simulation of combinational & sequential circuits.

**Unit 1: Boolean Algebra & Combinational Circuits**

Logic gates; Boolean algebra; De Morgan's theorem, SOP & POS forms, canonical forms, Karnaugh maps up to 6 variables, binary codes, code conversion, MSI devices like comparators; multiplexers; encoder; decoder; driver & multiplexed display; half and full adders; subtractors; serial and parallel adders; BCD adder; barrel shifter and ALU.

**Unit 2: Sequential Circuits**

Building blocks of sequential circuits like S-R, J-K, T& D flip-flops; master-slave J-K FF; edge triggered FF; ripple counters; synchronous counters; shift registers; finite state machines; design of synchronous FSM, algorithmic state machines charts; designing synchronous circuits like pulse train generator; pseudo random binary sequence generator; clock generation.

**Unit 3: Programmable Devices & ADC and DAC**

Specifications: noise margin, propagation delay, fan-in, fan-out, Tristate; TTL, ECL, CMOS families and their interfacing; architectures of PLA, PAL, GAL, CPLD&FPGA. DAC: weighted resistor, R-2R ladder, resistor string; ADC: single slope, dual slope, successive approximation, flash.

**Unit 4: Introduction to VHDL**

VHDL constructs; Data types and objects; different modelling styles in VHDL; Dataflow, Behavioral and Structural Modelling; Synthesis and Simulation; HDL programming for basic combinational and sequential circuits.

**Recommended Books**

1. R.P.Jain, Modern digital Electronics, Tata McGraw Hill
2. Douglas Perry,VHDL,Tata McGraw Hill
3. W.H.Gothmann, Digital Electronics-An introduction to theory and practice, PHI
4. D.V.Hall, Digital Circuits and Systems,Tata McGraw Hill
5. Charles Roth, Digital System Design using VHDL,TataMcGrawHill

| L | T | P | BTEC312-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 4 | Digital System Design Laboratory | 1 |

**Course Outcomes:** At the end of this course student will demonstrate the ability to:

CO-1. Realize combinational circuits using logic gates.
CO-2. Realize sequential circuits using logic gates.
CO-3. Write & simulate VHDL programs for combinational & sequential circuits.
CO-4. Think and design working projects using digital 74XX ICs.

## Part-A: List of Experiments

Task 1. To verify the Truth-tables of all logic gates.
Task 2. To realize and verify the Half & full adder circuits using logic gates.
Task 3. To realize Half & full subtractor circuits using logic gates.
Task 4. To realize 4-bit binary-gray & gray-binary converters.
Task 5. To realize comparator circuit for two binary numbers of 2-bit each.
Task 6. To realize Full adder & full subtractor circuits using 8x3 encoder.
Task 7. To design Full adder & full subtractor circuits using 8x3 demultiplexer.
Task 8. To design and verify the Truth tables of all flip-flops.
Task 9. To design Mod-6/Mod-9 synchronous up-down counter.
Task 10. To write VHDL program for combinational & sequential circuits from S. No. 2 to 7
Task 11. To write VHDL program for universal shift-register operations

## Part-B: Lab Projects

Every individual student is required to design one Lab Project under the supervision of course teacher. Topic of the project may be any from the theory contents and not limited to following list:

1. Pulse Width Modulator based LED dimmer using 555 timer IC.
2. Up-down 4-bit counter with seven-segment display.
3. Construction of combinational circuits using universal gates.
4. Bi-directional visitors counter
5. Traffic light control system
6. Any project based on Arduino platform

| L | T | P | BTEC301-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Electronic Devices | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Understand physics of semiconductors and behavior of charge carriers within semiconductors

CO-2. Understand the working of semiconductor diodes supported with mathematical explanation.

CO-3. Understand the working of BJT and MOSFET with their equivalent small signal models.

CO-4. Understand the chemical processes used in fabrication of integrated circuits.

**Unit 1: Semiconductor Physics**

Review of quantum mechanics; electrons in periodic lattices; e-k diagrams; energy bands in intrinsic and extrinsic silicon; diffusion current; drift current; mobility and resistivity; sheet resistance; design of resistors.

**Unit 2: Diodes**

Generation and recombination of carriers; Poisson and continuity equation p-n junction characteristics; V-I characteristics; small signal switching models; avalanche breakdown; Zener diode; Schottky diode; light emitting diode; tunnel diode; varactor diode, solar cell, Rectifier & Regulator circuits.

**Unit 3: Transistors**

Bipolar junction transistor; V-I characteristics; Ebers-Moll model; Transistor Configurations - CE, CB, CC; MOS capacitor; MOSFET - Construction and Working; I-V characteristics; Depletion-type and Enhancement-type MOS.

**Unit 4: Fabrication Processes**

Oxidation; diffusion; ion-implantation; Annealing; photolithography; etching; chemical vapour deposition (CVD); sputtering; twin-tub CMOS process.

**Recommended Books**

1. G. Streetman, S. K. Banerjee, Solid State Electronic Devices, Pearson.
2. D. Neamen, D. Biswas, Semiconductor Physics and Devices, McGraw-Hill Education
3. S. M. Sze and K. N. Kwok, Physics of Semiconductor Devices, John Wiley & Sons
4. C. T. Sah, Fundamentals of solid-state electronics, World

| L | T | P | BTEC311-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 2 | Electronic Devices Laboratory | 1 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Realize use of diodes in circuits with proper understanding to their working.
CO-2. Understand characteristics &working mechanisms of BJT in different configurations.
CO-3. Understand characteristics & working of MOSFET in circuits.
CO-4. Think and design working circuits based on diodes, BJTs and MOSFETs.

## Part-A: List of Experiments

Task 1. To Study of datasheets of semiconductor devices.
Task 2. To study the V-I characteristics of PN junction diode.
Task 3. To study Zener diode as voltage regulator.
Task 4. To study the output waveform of a Half-wave rectifier.
Task 5. To study the functioning of a Diode as a switch
Task 6. To study the output waveform of a Full-wave center -tapped and bridge rectifier
Task 7. To study Input & output V-I characteristics of npn/pnp BJT in CE configuration
Task 8. To study Input & output V-I characteristics of npn/pnp BJT in CB configuration
Task 9. To study the functioning of a BJT as a switch.
Task 10. To study V-I Characteristics of a MOSFET.

## Part-B: Lab Projects

Every individual student is required to design and build one Lab Project under the supervision of course teacher. Topic of the project may be any from the theory contents and not limited to following list:

1. Blinking linear/circular lights
2. Ambient light sensor-based controller
3. Regulated dual power supply of ±5V or ±12V or mixed
4. BJT audio amplifier
5. BJT circuit for sampling of analog signal
6. Any other in consultation with concerned teacher.

| L | T | P | BTCS302-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Object Oriented Programming | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Identify classes, objects, members of a class and the relationships among them needed to solve a specific problem

CO-2. Demonstrate the concept of constructors and destructors. And create new definitions for some of the operators

CO-3. Create function templates, overload function templates

CO-4. Understand and demonstrate the concept of data encapsulation, inheritance, polymorphism with virtual functions.

CO-5. Demonstrate the concept of file operations, streams in C++ and various I/O manipulators.

**Unit 1: Introduction**

Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & function components, recursive functions, user - defined types, function overloading, inline functions, Classes & Objects – I: classes, Scope resolution operator, passing objects as arguments, returning objects, and object assignment.

**Unit 2: Classes & Objects**

Constructors, Destructors, friend functions, Parameterized constructors, Static data members,

Functions,Arraysofobjects,Pointerstoobjects,thispointer,andreferenceparameter,Dynamic allocation of objects, Copy constructor, Operator overloading using friend functions, overloading.

**Unit 3: Inheritance**

Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes, Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes.

**Unit 4: Virtual functions, Polymorphism**

Virtual function, calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, Abstract classes, Using virtual functions, Early and late binding.

**Unit 5: Exception Handling**

Basics of exception handling, exception handling mechanism, throwing mechanism, catching mechanism,I/OSystemBasics,FileI/0:Exceptionhandlingfundamentals,Exceptionhandling options. C++ stream classes, Formatted I/O, fstreamand the File classes, Openingandclosing a file, Reading and writing text files.

**Recommended Books**

1. E.Balagurusamy,ObjectOrientedProgrammingwithC++,TataMcGrawHill.
2. StanleyB. Lippmann,JoseeLajoie:C++Primer,4thEdition,AddisonWesley, 2012.
3. Herbert Schildt: TheCompleteReferenceC++,4thEdition,TataMcGrawHill,2011.
4. Scientific Publishing Co.Inc

| L | T | P | BTCS304-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 4 | Object Oriented Programming Lab | 2 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Develop classes incorporating object-oriented techniques.
CO-2. Design and implement object-oriented concepts of inheritance and polymorphism.
CO-3. Illustrate and implement STL class of containers and need for exceptions to handle errors for object-oriented programs.
CO-4. Design and implement any real world-based problem involving GUI interface using object-oriented concepts.

## List of Experiment

Task 1. Write a program that uses a class where the member functions are defined inside a class.
Task 2. Write a program that uses a class where the member functions are defined outside a class.
Task 3. Write a program to demonstrate the use of static data members.
Task 4. Write a program to demonstrate the use of const data members.
Task 5. Write a program to demonstrate the use of zero argument and parameterized constructors.
Task 6. Write a program to demonstrate the use of dynamic constructor.
Task 7. Write a program to demonstrate the use of explicit constructors.
Task 8. Write a program to demonstrate the use of the initializer list.
Task 9. Write a program to demonstrate the overloading of increment and decrement operators.
Task 10. Write a program to demonstrate the overloading of memory management operators.
Task 11. Write a program to demonstrate the typecasting of basic type to class type.
Task 12. Write a program to demonstrate the typecasting of class type to basic type.
Task 13. Write a program to demonstrate the typecasting of class type to class type.
Task 14. Write a program to demonstrate the multiple inheritances.
Task 15. Write a program to demonstrate the runtime polymorphism.
Task 16. Write a program to demonstrate the exception handling.
Task 17. Write a program to demonstrate the use of a class template.
Task 18. Write a program to demonstrate the reading and writing of mixed types of data.

| L | T | P | HSMC101-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Development of Societies | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:
CO-1. Develop classes incorporating object-oriented techniques.

**Unit 1: Social Development**

1. Concepts behind the origin of Family, Clan and Society
2. Different Social Systems
3. Relation between Human beings and Society
4. Comparative studies on different models of Social Structures and their evolution

**Unit 2: Political Development**

1. Ideas of Political Systems as learnt from History
2. Different models of Governing system and their comparative study

**Unit 3: Economic Development**

1. Birth of Capitalism, Socialism, Marxism
2. Concept of development in pre-British, British and post British period Barter, Jajmani
3. Idea of development in current context.
4. E. F. Schumacher's idea of development, Buddhist economics.
5. Gandhian idea of development.
6. Swaraj and Decentralization.

| L | T | P | HSMC102-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Philosophy | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:
CO-1. Develop classes incorporating object-oriented techniques.

**Unit 1:** The difference between knowledge (Vidya) and Ignorance (Avidya):

1. Upanishads.
2. Six systems orthodox and Heterodox Schools of Indian Philosophy.
3. Greek Philosophy

**Unit 2:** Origin of the Universe:

1. NasidiyaSukta: "Who really knows?”
2. Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.
3. Taittiriya Upanishad: Siksha Valli.
4. Plato’s Symposium: Lack as the source of desire and knowledge.
5. Socratic method of knowledge as discovery.
6. Language: Word as root of knowledge (Bhartrahari’s Vakyapadiyam)
7. Fourteen Knowledge basis as a sources of Vidya: Four Vedas; Six auxiliary sciences (Vedangas); Purana, Nyaya, Mimamsa and Dharma Sastras.

**Unit 3:** Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

**Unit 4:** Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

**Unit 5:** Knowledge as invention: Modern definition of creativity; scientific activity in the claim that science

invents new things at least through technology.

**Unit 6:** Knowledge about the self, transcendental self; knowledge about society, polity and nature.

**Unit 7:** Knowledge about moral and ethics codes.

**Unit 8:** Tools of acquiring knowledge: Tantrayuktis, a system of inquiry (Caraka, Sushruta, Kautilya, Vyasa)

**Recommended Books:**

1. Copleston, Frederick, History of Philosophy, Vol. 1. Great Britain: Continuum.
2. Hiriyanna, M. Outlines of Indian Philosophy, Motilal Banarsidass Publishers.
3. Sathaye, Avinash, Translation of NasadiyaSukta
4. Ralph T. H. Griffith. The Hymns of the Ŗgveda. Motilal Banarsidass: Delhi: 1973.
5. Raju, P. T. Structural Depths of Indian Thought, Albany: State University of New York Press.
6. Plato, Symposium, Hamilton Press.
7. Kautilya Artha Sastra. Penguin Books, New Delhi.
8. Arnold, Edwin. The Song Celestial.
9. Foucault, Knowledge/Power.
10. Wildon, Anthony, System of Structure.
11. Dasgupta, S. N. History of Indian Philosophy, Motilal Banasidas, Delhi.
12. Passmore, John, Hundred Years of Philosophy, Penguin

| L | T | P | BMPD331-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 2 | Mentoring and Professional Development | 0 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:
**Guidelines:** The objective of mentoring will be development of:

1. Overall Personality
2. Aptitude (Technical and General)
3. General Awareness (Current Affairs and GK)
4. Communication Skills
5. Presentation Skills

The course shall be split into two sections i.e. (1) outdoor activities and (2) class activities. For achieving the above, suggestive list of activities to be conducted are:

**Part – A (Class Activities)**

1. Expert and video lectures
2. Aptitude Test
3. Group Discussion
4. Quiz (General/Technical)
5. Presentations by the student
6. 6.Team building exercises

**Part – B (Outdoor Activities)**

1. Sports/NSS/NCC
2. Society Activities of various student chapters, i.e., ISTE, SCIE, SAE, CSI, Cultural Club, etc.

Evaluation shall be based on rubrics for Part – A & B Mentors/Faculty Incharges shall maintain proper record of students for each activity conducted and the same shall be submitted to the department.

# Fourth Semester

| L | T | P | BTEC402-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Microprocessor and Microcontroller | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:
CO-1. Understand architecture & functionalities of different building block of8085 microprocessor.
CO-2. Understandworkingofdifferentbuildingblocksof8051microcontroller.
CO-3. Comprehendandapplyprogrammingaspectsof8051microcontroller.
CO-4. Interface & interact with different peripherals and devices.

**Unit 1: Microprocessor 8085**
History of microprocessors; microprocessor 8085 Architecture, Pin configuration; Memory Interfacing; microprocessor programming model; 8085 instructions; Addressing modes; programming techniques, counters and time delays; stack and subroutines; interrupts.

**Unit 2: Microcontroller 8051 - Building Blocks**
Microprocessor vs microcontroller; RISC vs CISC architectures; microcontroller 8051: architecture, pin configuration, flag-bits and PSW register, input-output ports, register banks and stack; semiconductor memories: ROM, SRAM, DRAM, virtual memory, cache memory; memory organization.

**Unit 3: Microcontroller 8051 - Programming**
Assembly language programming; data types and directives; jump loop and call instructions; I/O port programming; addressing modes and accessing memory using various addressing modes; arithmetic instructions and programs; logic instructions and programs; single bit instructions and programming, 8051 interrupts; timer/counter programming in the 8051.

**Unit 4: Microcontroller 8051 - Interfacing**
Parallel and serial ADC& DAC interfacing; LCD interfacing, Keyboard interfacing; sensor interfacing; interfacing with external memory; matrix keypad; stepper motor interfacing; DC motor interfacing and PWM.

**Recommended Books:**

1. RS Gaonkar, Microprocessor Architecture, Programming and Application with 8085,Penram International Publishing Pvt. Ltd.
2. KennethAyala,The8051Microcontroller,CengageLearning
3. Douglas Hall, Microprocessors Interfacing, TataMcGrawHill
4. SubrataGhoshal,8051Microcontroller:Internals,Instructions,ProgrammingandInterfacing, Pearson Education
5. KUmaRao,Andhe Pallavi,The8051Microcontrollers:Architecture,Programming.

| L | T | P | BTEC412-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Microprocessor and Microcontroller Laboratory | 1 |

**Course Outcomes** At the end of this course students will demonstrate the ability to:

CO-1. Write programs for common arithmetic operations with 8-bit/16-bit numbers using 8085.

CO-2. Write programs for transfer, sort block of data with 8085/8086 processor(s).

CO-3. Write programs for controlling stepper and DC motors using Microprocessor(s).

CO-4. Write programs to generate waveforms and interface ADC and DAC using 8051 Microcontroller.

## Part-A: Write programs in Assembly language & embedded C to

Task 1. Addtwo8-bitnumbersstoredinregistersorinternal/Externalmemorylocations.

Task 2. Multiplytwo8-bit numbers.

Task 3. Multiplytwo16-bitnumbers.

Task 4. Transfer block of data from internal memory locations to external memory locations

Task 5. Sort block of data in ascending or descending order.

Task 6. Generate5KHzpulsewaveformof50%dutycycle.

Task 7. Interface ADC and DAC.

Task 8. Interface Matrix Keyboard.

Task 9. Interface LCD Displays.

Task 10. Interface Stepper Motor.

Task 11. Control DC motor using PWM.

## Part-B: Lab Projects

Every individual student is required to design one Lab Project under the supervision of course teacher. Topic of the project may be any from the theory contents and not limited to following list:

1. RFID attendance system
2. Home automation
3. Robotic vehicle
4. Sensor traffic lights
5. Floor cleaning robot
6. Robot for defense applications
7. GPS vehicle tracking
8. Accident identification and SMS

| L | T | P | BTCS403-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Design and Analysis of Algorithms | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. For a given algorithm analyze the worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms.

CO-2. Explain when an algorithmic design situation calls for which design paradigm (greedy/ divide and conquer/backtrack etc.

CO-3. Explain models for a given engineering problem, using tree or graph, and write the corresponding algorithm to solve the problems.

CO-4. Demonstrate the ways to analyze approximation/randomized algorithms (expected running time, probability of error); &

CO-5. Examine the necessity for NP class-based problems and explain the use of heuristic techniques.

**Unit 1: Introduction**

Characteristics of algorithms. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

**Module2:Fundamental Algorithmic Strategies**

Brute-Force, Greedy, Dynamic Programming, Branch- and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving: Bin Packing, Knap Sack, TSP.

**Module3:Graph andTreeAlgorithms**

Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

**Module4:Tractable and Intractable Problems**

Computability of Algorithms, Computability classes–P,NP,NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques.

**Module5: Advanced Topics**

Approximation algorithms, Randomized algorithms, Heuristics and their characteristics.

**Recommended Books:**

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Data Structures and Algorithms in C++, Weiss, 4th edition, Pearson.
3. Fundamentals of Computer Algorithms – E. Horowitz, Sartaj Saini, Galgota Publications.
4. AlgorithmDesign,1stEdition,JonKleinbergandÉvaTardos, Pearson.
5. AlgorithmDesign:Foundations,Analysis,andInternetExamples,SecondEdition, Michael T Goodrich and Roberto Tamassia, Wiley.
6. Algorithms -- ACreative Approach, 3RDEdition, UdiManber, Addison-Wesley, Reading, MA.

| L | T | P | BTCS405-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 4 | Design and Analysis of Algorithms Lab | 2 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Improve practical skills in designing and implementing complex problems with different techniques.

CO-2. Understand comparative performance of strategies and hence choose appropriate, to apply to specific problem definition.

CO-3. Implement Various tree and graph-based algorithms and become familiar with their design methods.

CO-4. Design and Implement heuristics for real world problems.

**List of Experiment:**

Task 1. Code and analyze solutions to the following problem with given strategies:

Task 2. Knap Sack using greedy approach

Task 3. Knap Sack using dynamic approach

Task 4. Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.

Task 5. Code and analyze to find an optimal solution to TSP using dynamic programming.

Task 6. Implementing an application of DFS such as:

a. to find the topological sort of a directed acyclic graph
b. to find a path from source to goal in a maze.

Task 7. Implement an application of BFS such as:

a. to find connected components of an undirected graph
b. to check whether a given graph is bipartite.

Task 8. Code and analyze to find shortest paths in a graph with positive edge weights using Dijkstra’s algorithm.

Task 9. Code and analyze to find shortest paths in a graph with arbitrary edge weights using Bellman-Ford algorithm.

Task 10.Code and analyze to find shortest paths in a graph with arbitrary edge weights using Flyods’ algorithm.

Task 11.Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Prims’ algorithm

Task 12.Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Kruskals’ algorithm.

Task 13.Coding any real world problem or TSP algorithm using any heuristic technique.

**Recommended Books:**

1. Data Structures and Algorithms in C++,Weiss, 4thedition, Pearson
2. Data Structures and Algorithms using Python and C++,DavidM.Reed and John Zelle, 2009 edition (available as e book), Franklin Beedle& Associates

| L | T | P | BTEC401-18 | Credits |
|---|---|---|---|---|
| 3 | 1 | 0 | Analog Circuits | 4 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Understand the biasing of transistors and analyze BJT/FET amplifiers
CO-2. Analyze various rectifiers and amplifier circuits
CO-3. Analyze sinusoidal and non-sinusoidal oscillators
CO-4. Understand various types of Power Amplifiers

**Unit 1: Diode and Transistor Amplifier Circuits**

Diode Circuits, Amplifiers types: Voltage amplifier, current amplifier, trans-conductance amplifier and trans-resistance amplifier; biasing schemes for BJT and FET amplifiers; bias stability; transistor configurations: CE/CS, CB/CG, CC/CD and their features; small-signal analysis; low-frequency transistor models; amplifier analysis: current gain, voltage gain, input resistance and output resistance; amplifier design procedure; low frequency analysis of multistage amplifiers. High frequency transistor models.

**Unit 2: Feedback Amplifiers**

Feedback topologies: Voltage series, current series, voltage shunt and current shunt feedback; effect of feedback on gain, bandwidth, input & output impedances; concept of stability, gain margin and phase margin.

**Unit 3: Oscillators**

Introduction, Types of Oscillators, Barkhausen criterion, RC-phase shift, Wien bridge, Hartley, Colpitts, Clapp oscillators and Non-sinusoidal oscillators.

**Unit 4: Power Amplifiers**

Class A, B, AB and C power amplifiers, their efficiency and distortions; frequency response: single stage, multistage amplifiers and cascade amplifier.

**Recommended Books:**

1. J Millman & AGrabel, Microelectronics, McGraw Hill
2. J Millman & CHalkias, Integrated Electronics,TataMcGraw Hill
3. ARamakant,Gayakwad,Op-Amps AndLinear IntegratedCircuits,PHI
4. P Horowitz& W Hill, The Art of Electronics, Cambridge University Press
5. AS Sedra & KC Smith, Microelectronic Circuits, Saunder's College Publishing

| L | T | P | BTEC411-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 2 | Analog Circuits Laboratory | 1 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Study and verify the characteristics of diodes/BJTs in circuits with proper understanding

CO-2. Understand frequency response & working of various types of Oscillators.

CO-3. Understand characteristics & working of Power amplifiers.

CO-4. Think and design working circuits based on diodes, BJTs and MOSFETs.

### Part-A: List of Experiments

Task 1. To Study the Output waveforms of diode clipper and Diode Clamper circuits.

Task 2. To study BJT amplifier in CE configuration.

Task 3. To study V-I Characteristics of FET/MOSFET.

Task 4. To study Emitter follower circuit.

Task 5. To calculate the frequency and observe the output waveform of RC phase shift oscillator.

Task 6. To measure the frequency and observe the output waveform of Wein bridge oscillator.

Task 7. To measure the frequency and observe the output waveform of Hartley oscillator.

Task 8. To measure the frequency and observe the output waveform of Colpitt’s oscillator.

Task 9. To study Output waveform of Class-A Power Amplifier.

Task 10. To study Output waveform of Class-B Power Amplifier.

### Part-B: Lab Projects

Every individual student is required to design one Lab Project under the supervision of course teacher.Topicoftheprojectmaybeanyfromthetheorycontentsandnotlimitedtofollowing list:

1. BJT audio amplifier/power amplifier
2. Any project based on IoT/Arduino platform

| L | T | P | BTCS402-18 | Credits |
|---|---|---|---|---|
| 3 | 0 | 0 | Operating Systems | 3 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Explain basic operating system concepts such as overall architecture, system calls, user mode and kernel mode.

CO-2. Distinguish concepts related to processes, threads, process scheduling, race conditions and critical sections.

CO-3. Analyze and apply CPU scheduling algorithms, deadlock detection and prevention algorithms.

CO-4. Examine and categorize various memory management techniques like caching, paging, segmentation, virtual memory, and thrashing.

CO-5. Design and implement a file management system.

CO-6. Appraise high-level operating systems concepts such as file systems, disk-scheduling algorithms and various file systems.

**Module 1: Introduction**

Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

**Module2: Processes**

Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

**Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

**Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

**Module 3: Inter-process Communication**

Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson’s Solution, The Producer\Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader’s & Writer Problem, Dinning Philosopher Problem etc.

### Module 4: Deadlocks

Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

### Module5: Memory Management

Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation –Fixed and variable partition-Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation-Hardware support for paging, Protection and sharing, Disadvantages of paging.

**Virtual Memory**: Basics of Virtual Memory–Hardware and control structures–Locality of reference, Page fault, Working Set, Dirtypage/Dirtybit - Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used(NRU) and Least Recently used (LRU).

### Module6: I/O Hardware

I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms

**File Management**: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free Space Management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

**Disk Management:** Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.

### Recommended Books:

1. Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and DesignPrinciples,5thEdition, WilliamStallings, Prentice Hall of India.
3. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
4. OperatingSystems:AModernPerspective,2ndEdition by GaryJ.Nutt, Addison-Wesley
5. DesignoftheUnixOperatingSystems,8thEditionbyMauriceBach,Prentice-Hallof India
6. Understanding the Linux Kernel,3rd Edition, Daniel P.Bovet, Marco Cesati, O'Reilly and Associates.

| L | T | P | BTCS404-18 | Credits |
|---|---|---|---|---|
| 0 | 0 | 4 | Operating Systems Lab | 2 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. Understand and implement basic services and functionalities of the operating system.

CO-2. Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority.

CO-3. Implement commands for files and directories.

CO-4. Understand and implement the concepts of shell programming.

CO-5. Simulate file allocation and organization techniques.

CO-6. Understand the concepts of deadlock in operating systems and implement them in multiprogramming systems.

### List of Experiments

Task 1. Installation Process of various operating systems.

Task 2. Implementation of CPU scheduling algorithms to find turnaround time and waiting time.
a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority.

Task 3. Virtualization, Installation of Virtual Machine Software and installation of Operating System on Virtual Machine.

Task 4. Commands for files & directories: cd, ls, cp, md, rm, mkdir, rmdir. Creating and viewing files using cat. File comparisons. Disk related commands: checking disk free spaces. Processes in linux, connecting processes with pipes, background processing, managing multiple processes. Background process: changing process priority, scheduling of processes at command, batch commands, kill, ps, who, sleep. Printing commands, grep, fgrep, find, sort, cal, banner, touch, file. File related commands ws, sat, cut, grep.

Task 5. Shell Programming: Basic of shell programming, various types of shell, Shell Programming in bash, conditional & looping statement, case statements, parameter passing and arguments, shell variables, shell keywords, creating shell programs for automate system tasks, report printing.

Task 6. Implementation of Bankers algorithm for the purpose of deadlock avoidance.

**Reference Books:**

1. Operating Systems: Design and Implementation, Albert S. Woodhull and Andrew S. Tanenbaum, Pearson Education.

| L | T | P | BTCS401-18 | Credits |
|---|---|---|---|---|
| 3 | 1 | 0 | Discrete Mathematics | 4 |

**Course Outcomes:** At the end of this course students will demonstrate the ability to:

CO-1. To be able to express logical sentences in terms of predicates, quantifiers, and logical connectives.

CO-2. To derive the solution for a given problem using deductive logic and prove the solution based on logical inference.

CO-3. For a given mathematical problem, classify its algebraic structure.

CO-4. To evaluate Boolean functions and simplify expressions using the properties of Boolean algebra.

### Unit 1:Sets, Relation and Function

Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem.

Principles of Mathematical Induction: The Well-Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic.

### Unit 2: Counting Theory

Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination.

### Unit 3: Propositional Logic

Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The use of Quantifiers. Proof Techniques: Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency.

### Unit 4: Algebraic Structures and Morphism

Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings, Integral

Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form

**Unit 5: Graphs and Trees**

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

**Recommended Books:**

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
2. Susanna S. Epp, Discrete Mathematics with Applications, 4th edition, Wadsworth Publishing Co. Inc.
3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw – Hill.
4. J. P. Tremblay and R. Manohar, Discrete Mathematical Structure and Its Application to Computer Science", TMG Edition, Tata McGraw-Hill
5. Norman L. Biggs, Discrete Mathematics, 2nd Edition, Oxford University Press. Schaum's Outlines Series, Seymour Lipschutz, Marc Lipson,
6. Discrete Mathematics, Tata McGraw – Hill