C. Syllabus Detailing and Learning objectives

Module	Chapter	Detailed Content	Syllabus Detailing	Learning Objectives
Module 1	**Chapter 1 Set Theory (Hours -04)**	**Sets, Venn diagrams, Operations on Sets, Laws of set theory, Power sets and Products, Partition of sets, The Principle of Inclusion and Exclusion.**	**Purpose**: To make students understand the meaning of set; describe the laws of set theory; demonstrate the operations on set and explain the principle of inclusion/exclusion. **Scope –** **1. Academic Aspects-** set theory is used in the definitions of nearly all mathematical objects. **2. Technology Aspect-** logical and mathematical operators are used in programming languages. **3. Application Aspect-** Used for research areas like fuzzy sets. **Students Evaluation** – 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on set theory.	1. To **define** the sets and draw Venn diagrams for them **(R)** 2. To **solve** problem based on operations on sets**(A)** 3.To **apply** laws of set theory to solve expressions **(U**) 4. To **calculate** power set and products of given set.**(A)** 5. To **state** and **apply** the principle of Inclusion exclusion for solving real life problems. **(A)**
Module 2	**Chapter 2 Logic (Hours -08)**	**Propositions and logical operations, Truth tables, Equivalence, Implications, Laws of Logic, Normal**	**Purpose-** This chapter focuses on learning about propositions; represent English sentences in logical form; construct truth tables; and prepare formal proofs. In addition, this course will enhance critical reasoning skills and strengthen the ability to express that reasoning clearly.	1. To **describe** the basics of propositional logic. **(R)** 2. To **express** truth values for given statements and produce truth tables.(**U)**

		forms, Predicate and Quantifiers, Mathematical Induction	**Scope –** **1. Academic Aspects-** a formal language for representing knowledge and for making **logical** inferences. **2. Technology Aspect-** It is used in Artificial Intelligence to infer rules; logical operators are used in programming languages. **3. Application Aspect**- It is used to write formal rules in theoretical research work. **Students Evaluation –** 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on Propositional Logic.	3. To **derive** conclusions using various laws of logic. **(U)** 4. To compare Boolean functions by **expressing** them into normal forms. **(U)** 5. To **interpret** statements using predicate and quantifiers. **(A)** 6. To understand mathematical induction and **apply** it to give formal proofs. **(A)**
Module 3	**Chapter 3 Relations and Functions (Hours -12)**	**Relations, Paths and Digraphs, Properties and types of binary relations, Operations on relations, Closures, Warshall's algorithm, Equivalence and Partial ordered relations, Poset, Hasse diagram and Lattice Functions: Types of functions- Injective,**	**Purpose –** To provide students with the knowledge of relations and functions; describe properties and their types; apply Warshall's algorithm to compute transitive closure; determine whether a relation on a set is a POSET or a lattice; compute composition, identity and inverse of a given function. **Scope –** **1. Academic Aspects-** idea of relations and functions is introduced and differentiated. **2. Technology Aspect-** used to define mathematical models in theoretical computer science. **3. Application Aspect-** Operations like union, intersection etc. are used in database systems.	1. To **differentiate** between Relations and Functions and **describe** their types.**(R)** 2. To **apply** operations on relation using examples. **(U)** **3. To apply** Warshall's algorithm to find transitive closure **(A**). 4. To **determine** a relation on a set is a POSET or a Lattice and to draw corresponding Hasse Diagram **(AN)** 5. To **compute** composition, identity and inverse of a given function. **(A)**

		Surjective and Bijective, Composition of functions, Identity and Inverse function, Pigeon-hole principle	**Student Evaluation -** 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on relations and functions.	6. To **describe** pigeonhole principle and use it to **solve** problems.**(A)**
Module 4	**Chapter 4 Counting (Hours -06)**	**Permutations, Combinations, Elements of Probability, Discrete Probability and Conditional Probability, Generating Functions and Recurrence Relations, Recursive Functions, Introduction to Functional Programming**	**Purpose –** To make the student understand difference between permutation and combination; relate counting of outcomes to probability; define discrete probability and conditional probability; use recurrence relations, simplify codes using functional programming. **Scope –** **1. Academic Aspects-** Apply the theory of permutations and combinations to solve counting problems **2. Technology Aspect-** Counting techniques are used for game development. **3. Application Aspect-** The Fundamental Counting Principle is the guiding rule for finding the number of ways to accomplish a tasks. **Student Evaluation -** 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on Counting.	1. To **differentiate** between permutation and combination. **(U)** 2. To **describe** the elements of probability. **(R)** 3. To **define** discrete probability and conditional probability. **(R)** 4. To **Solve** generating functions and recurrence relations and apply them for finding the complexity of algorithms. **(A)** 5. To **use** functional programming approach to simplify the code. **(A)**
Module 5	**Chapter 5 Graphs (Hours -06)**	**Definitions, Paths and circuits: Eulerian and Hamiltonian, Types**	**Purpose –** This chapter explains graphs and computes sub-graphs; represent real-life situations with mathematical graphs; define paths and circuits; determine isomorphism in graphs.	1. To **define** graphs and **compute** sub graphs**. (R)** 2. To **define** paths and circuits and

		of graphs, Sub-graphs, Isomorphism of graphs		**differentiate** between Eulerian and Hamiltonian circuits. **(U)** 3. To **determine** isomorphism in graphs. **(AN)**
			Scope – **1. Academic Aspects-** study of graphs that concerns with the relationship among edges and vertices. **2. Technology Aspect-** used in designing routing algorithms, communication protocols. **3. Application Aspect-** to identify which programs can run concurrently having similar type of resource requirements using graph theory techniques; identify information flow in social networking etc.	
			Student Evaluation – 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on graphs.	
Module 6	**Chapter 6 Algebraic Structures and Coding Theory (Hours -08)**	**Algebraic structures with one binary operation: semigroup, monoid and group, Abelian group, Isomorphism, Homomorphism and Automorphism, Cyclic groups, Normal subgroups, Codes and group codes**	**Purpose –** Study of algebraic structures; define groups, semigroup, monoid, Abelian group; understand Isomorphism, Homomorphism and Automorphism; apply codes and group codes for various applications.	1. To **define** algebraic structures with one binary operation. **(R)** 2. To define and **differentiate** between semigroup, monoid, and group. **(U)**
			Scope – **1. Academic Aspects-** understand algebraic structures and basics of coding theory. **2. Technology Aspect-** used as a part of encryption tools. **3. Application Aspect-** can be used for data compression, cryptography, error coding and network coding.	3. To **define** an Abelian group. **(R)** 4. To state the definition of group Homomorphism, Isomorphism, Automorphism and **use** it to prove or disprove that a given map is a

				Homomorphism, Isomorphism, Automorphism. **(A)**
			Student Evaluation – 1. Questions for solving as a part of tutorial. 2. Solving assignments. 3. Solving GATE questions based on algebraic structures and coding theory.	5. To **define** Cyclic groups and Normal Subgroups and give examples**.(R)** 6 To **apply** codes and group codes for various applications i.e. Data compression, cryptography, error coding and network coding**. (A)**