B. Sc. (Information Technology)		Semester – I	
Course Name: Discrete Mathematics		**Course Code: VGVUSTVSE101**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand / implement
1. Concept of set theory
2. Elementary Number Theory and Methods of Proof.
3. Use of Addition Rule and probability concepts.
4. Concept of Relations, graphs and trees.
5. Sequences, Mathematical Induction, and Recursion.

Unit	Details	Lectures
I	**Introduction**: Variables, The Language of Sets, The Language of Relations and function. **Functions**: Functions Defined on General Sets, One-to-One and Onto, Inverse Functions, Composition of Functions, and Cardinality with Applications to Computability. **Relations**: Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations. **Set Theory**: Definitions and the Element Method of Proof, Properties of Sets, Disproof's, Algebraic Proofs, Boolean Algebras, Russell's Paradox and the Halting Problem.	**10**
II	**The Logic of Compound Statements**: Logical form and Logical Equivalence, Conditional Statements, Valid and Invalid Arguments. **Quantified Statements**: Predicates and Quantified Statements, Statements with Multiple Quantifiers, Arguments with Quantified Statements. **Elementary Number Theory and Methods of Proof**: Introduction to Direct	**10**

	Proofs, Rational Numbers, Divisibility, Division into Cases and the QuotientRemainder Theorem, floor and Ceiling, Indirect Argument Contradiction and Contraposition, Two Classical Theorems, Applications in algorithms. **Probability** : Basics of Probability, Addition rule	
III	**Sequences, Mathematical Induction, and Recursion**: Sequences, Mathematical Induction, Strong Mathematical Induction and the Well-Ordering Principle or the Integers, Correctness of algorithms, defining sequences recursively, solving recurrence relations by iteration, Second order linear homogeneous recurrence relations with constant coefficients. General recursive definitions and structural induction. **Graphs and Trees**: Definitions and Basic Properties, Trails, Paths, and Circuits, Matrix Representations of Graphs, Isomorphism of Graphs, Trees, Rooted Trees, Isomorphism of Graphs, Spanning trees and shortest paths	**10**

Course Outcome
Learners should be able to
CO1 Understand the basic principles of sets and operations in sets.
CO2 Understand the working with relations and investigate their properties.
CO3 Analyse mathematical properties using mathematical induction methods, study functions, spaces, and other mathematical structures using sequences and use of recursion.
CO4 Understand relation, graphs and trees in various applications.
CO5 Understand the use of the SCILAB tool to solve mathematical problems.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Discrete Mathematics with Applications	Sussana S. Epp	Cengage Learning	4th	2010
2.	Discrete Mathematics, Schaum's Outlines Series	Seymour Lipschutz, Marc Lipson	Tata MCGraw Hill		2007
3.	Discrete Mathematics and its Applications	Kenneth H. Rosen	Tata MCGraw Hill		
4.	Discrete mathematical structures	B KolmanRC Busby, SRoss	PHI		
5.	Discrete structures	Liu	Tata MCGraw Hill		

B. Sc. (Information Technology)		Semester – I	
Course Name: Numerical Computation using Scilab Practical		**Course Code: VGVUSTVSEP101**	
Periods per week (1 Period is 120 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	**100**
	Internal	**--**	**--**

Course Objective
To make learner understand / implement
1. To be able execute programs of discrete mathematics concepts in Scilab software easily.
2. To implement programs on Set theory, functions and algorithms.
3. To execute different topics related to probability using the Scilab software.
4. To represent concepts of graph theory, directed graphs, and their subtopics in the form of a program.
5. To implement topics like Algebraic systems, Boolean algebra, Recurrence relations in a practical manner.

List of Practical: Write the programs or the following using SCILAB	
1.	**Set Theory :**
a.	Inclusion Exclusion principle.
b.	Power Sets
2.	**Functions and Algorithms :**
a.	Recursively defined Functions
b.	Cardinality
c.	Polynomial evaluation
d.	Greatest Common Divisor
3.	**Sequences :**
a.	Summation Notation, Product Notation

b.	Mathematical Induction
4.	**Probability Theory :**
a.	Sample space and events
b.	Finite probability spaces
5.	**Counting Elements Of Disjoints Sets :**
a.	Addition Principle
b.	Conditional Probability
c.	Independent events
d.	Repeated trials with two outcomes
6.	**Directed Graphs :**
a.	Adjacency matrix
b.	Path matrix
7.	**Undirected Graphs :**
a.	Adjacency matrix
b.	Path matrix
8.	**Graph Theory :**
a.	Paths and connectivity
b.	Minimum spanning tree
c.	Isomorphism
9.	**Tree :**
a.	Minimum spanning tree
b.	Shortest path algorithm Kruskal or Prims
10.	**Recurrence relations :**
a.	Linear homogeneous recurrence relations with constant coefficients
b.	Solving linear homogeneous recurrence relations with constant coefficients
c.	Solving general homogeneous linear recurrence relations

Course Outcome
To make learner understand / implement
1. Learner will be able to become familiar with the Scilab environment
2. Implement programs on Inclusion Exclusion principle, power sets, recursively defined functions, Mathematical Induction Cardinality in Scilab
3. Execute programs like Sum principle, Product principle, Factorial, Permutations and Combinations, Sample space and events, Conditional Probability, Finite probability spaces
4. Implement concepts in Scilab like paths and connectivity, minimum spanning tree, isomorphism, adjacency matrix, path matrix.
5. Implement recurrence relations by iteration, Second order linear homogenous recurrence relations with constant coefficients