Gyanmanjari
Innovative University

Course Syllabus
Gyanmanjari College of Computer Science
Semester-1(MSCIT)

Subject: Mathematics - MSCIT11301

Type of course: Multidisciplinary

Prerequisite: Basic Algebra, basic mathematics, binary number system etc

Rationale: The basic concepts of sets, logic functions and graph theory are applied to Boolean algebra and logic networks, while the advanced concepts of functions and algebraic structures are applied to finite state machines and coding theory.

Sets and Relations: Sets and Relations are a basic concept in discrete mathematics that provides a way of grouping objects together. Sets are used in many areas of mathematics, computer science, and other fields to represent collections of items. Sets and Relations can be used to define functions, create models of data structures, and describe algorithms.

Matrices: Matrices are a way of representing data in a compact and efficient form. They are used in many areas of mathematics, science, and engineering to represent systems of equations, linear transformations, and data structures. Matrices can also be used to analyze networks and graphs, and to solve optimization problems.

Graph Theory: Graph theory is the study of graphs, which are mathematical structures that represent relationships between objects. Graphs are used in many areas of computer science, mathematics, and engineering to model complex systems, such as communication networks, transportation systems, and social networks. Graph theory provides tools for analyzing the structure and properties of graphs, and for solving optimization problems on graphs.

Boolean algebra: Boolean algebra is a branch of algebra that deals with binary variables and logic operations. It is used in digital circuit design, computer programming, and other areas of computer science and engineering to analyze and manipulate logic circuits. Boolean algebra provides a rigorous mathematical framework for reasoning about logical expressions, and for designing and optimizing digital circuits.

Teaching and Examination Scheme:

GYANMANJARI INNOVATIVE UNIVERSITY BHAVNAGAR

Teaching Scheme			Credits	Examination Marks					Total Marks
				Theory Marks		Practical Marks		CA	
CI	T	P	C	ESE	MSE	V	P	ALA	
3	0	2	4	60	30	10	20	30	150

Legends: CI-Class Room Instructions; T – Tutorial; P - Practical; C – Credit; ESE - End Semester Examination; MSE- Mid Semester Examination; V – Viva; CA - Continuous Assessment; ALA- Active Learning Activities.

Continuous Assessment:

(For each activity maximum-minimum range is 5 to 10 marks)

Sr. No	Active Learning Activities	Marks
1	**Sets and Relations:** Use Venn diagrams to visually represent sets and their relationships. Have students practice creating Venn diagrams and using them to solve problems involving set operations.	6
2	**Functions and Combinatorics:** Graphing functions: Have students practice graphing different types of functions, such as linear, quadratic, and exponential functions. ii. Permutations: have students about permutations and have them practice finding the number of permutations of different sets	6
3	**Graph theory and tree:** i. Introduce graph coloring and have students practice coloring different types of graphs with different numbers of colors. ii. Introduce binary search trees and have students practice implementing them and using them to solve problems.	6
4	**Boolean algebra:** Introduce Karnaugh maps and have students practice using them to simplify Boolean expressions.	6
5	**Matrices and its Application:** Introduce different applications of matrices in real-life situations, such as image processing, optimization problems, and computer graphics. Have students practice applying their knowledge of matrices to solve problems in these fields.	6
	Total	**30**

Course Content:

Sr. No	Course content	Hrs	% Weightage
1	**Sets and Relations** Set operations: union, intersection, difference, complement Power set, Set identities and laws Relations between sets: subsets, partitions, equivalence classes, Binary relations: definition, types, properties, Composition, and inverse of relations, Equivalence relations and partitions, Applications of relations in modeling real-world scenarios, such as social networks and computer networks	10	18
2	**Functions and Combinatorics** Definitions and properties of functions: domain, range, co-domain, injectivity, surjectivity, bijectivity, Composition of functions, Inverse functions, and some elementary functions, Basic counting principles for multiplication, addition, inclusion-exclusion, Permutations and combinations, Generating function and Recurrence relations with applications.	10	18
3	**Graph theory and tree** Definitions of graphs, vertices, and edges, Types of graphs: directed and undirected, simple and multi graphs, weighted and un-weighted graphs, Degree of vertices, paths, cycles, and connectedness, Graph algorithms: graph traversal algorithms, shortest path algorithm, minimum spanning tree algorithms, Definition of trees, roots, branches, and leaves, Properties of trees: height, depth, size, and degree, Tree traversal algorithms: preorder, inorder, and postorder traversal, Binary trees and their applications in sorting and searching algorithms	12	24
4	**Number theory and Boolean algebra** Properties of integers: divisibility, prime numbers, Greatest common divisor (GCD) and least common multiple (LCM), Decimal, binary, octal, hexadecimal integer numbers in computer logic design, Logical operators, law of commutative, associative, and distributive, De Morgan's theorem, truth table, Karnaugh map, Boolean functions, simplification techniques for logical expressions.	10	24
5	**Matrices and its Application** Types of matrices, Matrix operations: Addition, subtraction, and multiplication of matrices, Scalar multiplication, Transpose of a matrix, Determinant of square matrix, Linear transformations: rotation, reflection, scaling, and shearing, Use of matrix in encoding and decoding.	8	16

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks

(Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage	25%	50%	25%	00%	00%	00%

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcome:

After learning the course the students should be able to:	
CO1	Develop critical thinking skills through problem solving exercise of sets, relations, and functions.
CO2	Understand and apply counting principles, permutations, combinations, and generation of functions.
CO3	Understand and apply graph algorithms, tree structures as a mathematical tool.
CO4	Understand integer number theory and Boolean algebra as a tool to solve complex logic expressions.
CO5	Understand the basic concept of matrices as a mathematical tool to implement data structure and to develop encryption and decryption algorithm in cryptography.

List of Practical

(Minimum-10 practical):

Sr. No	Descriptions	Unit No	Hrs
1	Write a C program to determine the greatest common divisor (GCD) and least common multiple (LCM) of two given integers. Your program should take the input integers and output their GCD and least common multiple (LCM).	4	2
2	Write a C program to generate all divisors of a given integer. Your program should take the input integer and output a list of all its divisors.	4	2
3	Write a C program to determine the sum of all divisors of a given integer. Your program should take the input integer and output the sum of its divisors.	4	2
4	Write a C program to implement a set data structure. Your program should allow the user to add and remove elements from the set, check if an element is in the set, and perform	1	2

	set operations such as union, intersection, and difference.		
5	Write a C program to compute the Cartesian product of two given sets. Your program should take the input sets as arrays of integers and output the Cartesian product as a list of pairs.	1	2
6	Write a C program to check if a given function is injective, surjective, or bijective. Your program should take the input function as a list of pairs and output the properties of the function.	2	2
7	Write a C program to implement basic Boolean operations such as AND, OR, and NOT. Your program should take two Boolean values as input and output the result of the specified operation.	4	2
8	Write a C program to implement Karnaugh maps for simplifying Boolean functions. Your program should take the input truth table as a matrix of integers and output the simplified Boolean function as a list of terms.	4	2
9	Write a C program to implement the De Morgan's laws of Boolean algebra. Your program should take the input Boolean expression as a string and output the simplified expression using De Morgan's laws	4	2
10	Write a C program to add, subtract and multiply two matrices of the same size. Your program should take the input matrices.	5	2
11	Write a C program to transpose a matrix. Your program should take the input matrix and output its transpose.	5	2
12	Write a C program to implement a tree data structure. Your program should allow the user to add and remove nodes, check if a node is in the tree, and perform tree algorithms such as finding the height and diameter of the tree.	3	2
13	Write a C program to implement a binary search tree data structure. Your program should allow the user to add and remove nodes, check if a node is in the tree, and perform tree algorithms such as finding the minimum and maximum value in the tree.	3	2
14	Write a C program to check if a given function is injective, surjective, or bijective. Your program should take the input function as a list of pairs and output the properties of the function.	2	2
		Total	**28**

Instructional Method:

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory

The internal evaluation will be done on the basis of Active Learning Assignment

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.

Reference Books:

[1] R. S. Agarwal, "Mathematics for computer application"

[2] M. Morris Mano, "Digital Logic and Computer Design," Pearson Education

[3] William Stallings, "Cryptography and Network Security," Pearson

[4] J. P. Tremblay and W. K. Grassman. "Logic and Discrete Mathematics," Pearson Education