DISCRETE STRUCTURES

II B. TECH- I SEMESTER

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIE	SEE	Total
A4CS03	PCC	3	1	-	4	30	70	100

COURSE OBJECTIVES

The course should enable the students to:

1. To help students understand discrete and continuous mathematical structures
2. To impart basics of relations and functions
3. To facilitate students in applying principles of Recurrence Relations to calculate generating functions and solve the Recurrence relations
4. To acquire knowledge in graph theory

COURSE OUTCOMES:

At the end of the course, student will be able to

1. Apply the knowledge of discrete and continuous mathematical structures.
2. Solve various problems on relations and functions.
3. Apply the principles of Recurrence Relations to generate functions and solve various problems on it.
4. Solve problems using the knowledge of graph theory.

UNIT-I	MATHEMATICAL LOGIC	Classes: 11

Statements and notations, Connectives, Well formed formulas, Truth Tables, Tautology, Equivalence implication, Normal forms, Logical Inference, Rules of inference, Direct Method, Direct Method using CP(Conditional Proof), Consistency, Proof of contradiction, Automatic Theorem Proving. Quantifiers, Universal quantifiers. Predicates: Predicative logic, Free & Bound variables.

UNIT-II	RELATIONS	Classes: 16

Introduction to set theory**,** Relations, Properties of Binary Relations, Equivalence Relation, Transitive closure, Compatibility and Partial ordering relations, Lattices, Hasse diagram. Functions: inverse Function , Composition of functions, Recursive Functions

UNIT-III	ELEMENTARY COMBINATORICS	Classes: 12

Basis of counting, Combinations & Permutations, Enumeration of Combinations and Permutations, Enumeration of Combinations and Permutations With repetitions, Enumerating Permutations with Constrained repetitions, Binomial Coefficients, Binomial and Multinomial theorems, The principles of Inclusion **–** Exclusion, Pigeon- hole principles and its applications.

UNIT-IV	RECURRENCE RELATION	Classes: 11

Generating Functions, Function of Sequences, Calculating Coefficient of generating function, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristics roots, Solution of Inhomogeneous Recurrence Relation.

UNIT-V	GRAPHS	Classes: 10

Basic Concepts, Isomorphism and Sub graphs, Trees and their properties, Spanning Trees- DFS,BFS, Minimal Spanning Trees- Prims, Kruskal's Algorithm, Planar Graphs, Euler's Formula, Multi graph and Euler circuits, Hamiltonian Graphs, Chromatic number.

TEXT BOOKS:

1. T1. Discrete Mathematics for computer scientists & Mathematicians, *J.L. Mott, A. Kandel, T.P.* Baker PHI
2. Discrete Mathematical Structures With Applications to Computer Science, JP Tremblay, R Manohar

Reference Books:

1. R1. Logic and Discrete Mathematics, *Grass Man & Trembley*, Pearson Education.

DATA STRUCTURES

II Year I Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIE	SEE	Total
A4CS04	PCC	3	1	-	4	30	70	100

Course Objectives:

1. Impart the basic concepts of data structures and algorithms.
2. Understand concepts linked lists and their applications.
3. Understand basic concepts about stacks, queues and their applications.
4. Understand basic concepts of trees, graphs and their applications.
5. Enable them to write algorithms for sorting and searching and hashing.
6. Use advanced data structures like B-Trees, AVL-trees etc., for efficient problem solving.

Course Outcomes

At the end of the course, student will be able to:

1. Evaluate algorithms in terms of time and memory complexity.
2. Formulate new solutions for problems or improve existing code using data structures and algorithms.
3. Implement basic data structures such as arrays, linked lists, stacks and queues.
4. Solve problem involving graphs, trees and heaps
5. Apply Algorithms for solving problems like sorting, searching, and hashing.
6. Implement advanced data structures such as B-Trees, Red-Black, and AVL-Trees

UNIT-I	INTRODUCTION TO DATA STRUCTURES	Classes: 12

Basic concepts- Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction, Performance analysis- time complexity and space complexity, Asymptotic Notation-Big O,Omega and Theta notations, **Introduction to Linear and Non Linear data structures**-Singly Linked Lists-Operations-Insertion, Deletion, Concatenating singly linked lists, Circularlylinked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations- Insertion,Deletion. Representation of single, two dimensional arrays, sparse matrices-array and linkedrepresentations.

UNIT-II	STACKS AND QUEUES	Classes: 10

Stacks-Stack ADT, definition, operations, array and linked implementations in C, applications-infix topostfix conversion, Postfix expression evaluation, recursion implementation,

Queues-Queue ADT,definition and operations ,array and linked Implementations in C, Circular queues-Insertion anddeletion operations, Dequeue (Double ended queue)ADT, array and linked implementations in C.

UNIT-III	TREES AND GRAPHS	Classes: 14

Trees – Terminology, Representation of Trees, Binary tree ADT, Properties of Binary Trees,Binary Tree Representations-array and linked representations, Binary Tree traversals, Threadedbinary trees,

Max Priority Queue-ADT-implementation-Max Heap-Definition, Insertion into aMax Heap, Deletion from a Max Heap. Graphs , Introduction, Definition, Terminology, GraphADT, Graph Representations-Adjacency matrix, Adjacency lists, Graph traversals- DFS andBFS.

UNIT-IV	SEARCHING AND SORTING	Classes: 12

Searching- Linear Search, Binary Search, Static Hashing-Introduction, hash tables, hashfunctions, Overflow Handling.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Merge Sort, Heap Sort, Comparison of

Sorting methods.		
UNIT-V	**BINARY SEARCH TREES**	**Classes: 12**
Search Trees-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion,AVL Trees- Definition and Examples, Insertion into an AVL Tree ,B-Trees, Definition, B-Treeof order m, operations-Insertion and Searching, Introduction to Red-Black and SplayTrees(Elementary treatment-only Definitions and Examples), Comparison of Search Trees. **Pattern matching algorithm**- The Knuth-Morris-Pratt algorithm, Tries (examples only).		
Text Books:		
1. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press. 2. Fundamentals of Data structures in C, 2nd Edi on, E.Horowitz, S.Sahni and SusanAnderson-Freed, Universi es Press.		
Reference Books:		
1. Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company 2. "How to Solve it by Computer", 2nd Impression by R. G. Dromey, Pearson Education		
Web References:		
1. https://hackr.io/tutorials/learn-data-structures-algorithms 2. https://www.geeksforgeeks.org/fundamentals-of-algorithms/ 3. https://www.udemy.com/introduction-to-algorithms-and-data-structures-in-c/ 4. https://leetcode.com		
E-Text Books:		
1. http://www.freetechbooks.com/algorithm-analysis-and-design-t1030.html 2. http://www.freetechbooks.com/algorithmic-problem-solving-t373.html 3. http://www.freetechbooks.com/algorithms-and-data-structures-the-basic-toolbox-t871.html		
MOOC Course		
1. https://www.coursera.org/specializations/data-structures-algorithms 2. https://onlinecourses.nptel.ac.in/noc16_cs06/preview		