SARVAJANIK UNIVERSITY
INCLUSIVE | INTEGRATED | INNOVATIVE

SARVAJANIK UNIVERSITY
Sarvajanik College of Engineering and Technology
Master of Technology

Year: M. Tech. I (Semester – I)

Subject Name: Advanced Data Structures and Algorithms **Subject Code:** MTCO24101
Type of course: Professional Elective-I
Prerequisite (if any): Data structure, Algorithms

List of Courses where this course will be prerequisite: --

Rationale: Data structures play a central role in modern computer science. Data structures are essential building blocks in obtaining efficient algorithms. This course covers a systematic study of the methods of structuring and manipulating data in computing and major advanced data structures which can be used for efficient searching.

Teaching and Examination Scheme:

Teaching Scheme				Theory Marks			Practical Marks		Total
L	T	P	C	TEE	CA1	CA2	TEP	CA3	
3	0	2	4	60	25	15	30	20	150

CA1: Continuous Assessment (assignments/projects/open book tests/closed book tests CA2: Sincerity in attending classes/class tests/ timely submissions of assignments/self-learning attitude/solving advanced problems TEE: Term End Examination TEP: Term End Practical Exam (Performance and viva on practical skills learned in course) CA3: Regular submission of Lab work/Quality of work submitted/Active participation in lab sessions/viva on practical skills learned in course

Content:

Sr. No.	Content	Total Hrs
1	Introduction to Basic Data Structures: Importance and need of good data structures and algorithms, Strategies for choosing the appropriate data structures, basics of algorithms, Performance analysis- time complexity and space complexity, Asymptotic Notation-Big Oh, Omega and Theta notations	4
2	Data structures : Linear and non linear data structures, Linear List ADT, Linked representation, singly linked lists -insertion, deletion, search operations, doubly linked lists-insertion, deletion operations. Skip Lists : Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists	8
3	Balanced Trees: Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees	8

4	Text Processing and Compression : Sting Operations, Standard Tries, Compressed Tries, Suffix Tries, The Huffman Coding Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem	6
5	Computational Geometry: One Dimensional Range Searching, Two Dimensional Range Searching, Constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quadtrees, k-D Trees	6
6	Flow-Networks: Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm	5
7	Matrix Operations : Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition Linear Programming : Geometry of the feasibility region and Simplex algorithm	8

Reference Books:

Sr No.	Title of book /article	Author(s)	Publisher and details like ISBN	Year of publication	Publication Edition
1	Algorithm Design	M T Goodrich, Roberto Tamassia,	John Wiley		
2	Data structures and algorithms	Aho, Hopcroft and Ullman			
3	Introduction to Algorithms	Thomas Cormen, et. al	PHI	2009	2nd Edition

Course Outcomes:

Sr. No.	CO statement	Marks % weightage
CO-1	Develop and analyze algorithms for dynamic data structures	26
CO-2	Develop algorithms for efficient search trees and text processing	31
CO-3	Identify suitable data structures and develop algorithms for computational geometry.	14

CO-4	Formulate a real-life problem into a mathematical model and solve it using linear programming.	18
CO-5	Apply the maximum flow algorithms to solve maximum flow problems	11

List of Open learning website:

1. NPTEL course on Data Structures And Algorithms - Video course, Prof. Naveen Garg, Department of Computer Science and Engineering, IIT Delhi

List of Open Source Software: Linux, CentOS

FOR LAB SESSIONS:

List of Experiments:

Sr No.	Practical
1	Write a menu driven program to implement various operations on the singly linked list (as discussed in the theory class).
2	Write a program for binary search tree to demonstrate following functionalities : 1) Create/insert tree, 2) Tree traversals inorder, preorder and post order, 3) Delete operation 4) Copy Tree Above program should also be done using every node as a record of a student and the arrangement of the node should be based on the roll number.
3	Write a program which creates Skip Lists. Implement Insert, Search and Update Operations in Skip-Lists.
4	Write a program which creates AVL Tree. Implement Insert and Delete Operations in AVL Tree. Note that each time the tree must be balanced.
5	Implement 2-3 Tree with basic operations.
6	Implement B Tree with basic operations.
7	Implement Red-Black Tree with basic operations.
8	Implement One Dimensional Range Searching in any language.

9	Text Processing : Implement Huffman-Coding Method. Show the result with a suitable example.
10	Implement Ford-Fulkerson Algorithm.
11	Implement Edmonds-Karp Algorithm.
12	Implement and analyze time complexity of different matrix operations: • Matrix Addition/Subtraction • Matrix Multiplication • Matrix Inversion • Transpose of Matrix
13	Solve the following problem of linear programming using pulp library. MINIMIZE: 3X + 5Y SUBJECT TO: C1: 2X + 3Y >= 12 C2: -X + Y <= 3 C3: X >= 4 C4: Y <= 3

Major Equipment Needed: