



Daffodil International University
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Final Examination, Spring 2025
Course Code: CSE123, Course Title: Data Structures
Level:1 Term:2 Batch: ALL

Time: 2.0 Hours**Marks: 40**

Answer ALL Questions

[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]

1.	a) Explain the role of Balancing in Binary Search Trees (BST) and why AVL trees are preferred over unbalanced BSTs. Provide an example. b) Problem Scenario: Efficient Student Record Management Using BST <i>A university wants to efficiently manage student records using a Binary Search Tree (BST). Each node in the BST represents a student and contains the following information:</i> • Student ID (int) – serves as the unique key • Name (string) • CGPA (float) The system must support the following operations: 1. Insertion: New student records should be inserted while maintaining BST properties. 2. Searching: Given a Student ID, find and display the corresponding student's Name and CGPA. 3. Deletion: If a student record is deleted, ensure that the BST remains valid (i.e., replacing with inorder successor or predecessor). Tasks: i) (2 Marks) Define the BST node based on the above requirements. ii) (3 Marks) Illustrate step-by-step how the following Student IDs are inserted into an initially empty BST: (105, 120, 90, 75, 110, 95, 130, 125, 140) • Draw the BST after each insertion.	5	CO1
		10	CO3

	iii) (3 Marks) Suppose the Student ID 120 needs to be deleted. Show the updated BST after performing deletion while maintaining BST properties. iv) (2 Marks) Analyze the worst-case and average-case time complexity of insertion, searching, and deletion in a BST. Discuss how an unbalanced BST can affect performance and propose an alternative approach to improve efficiency. <i>(Full correct implementation: 10 marks, Partial correctness: 6-8 marks, Minor issues: 3-5 marks)</i>		
2.	a) (i) Explain the differences between In-order, Pre-order, and Post-order traversal of a Binary Tree with an example? (ii) How a MinHeap is different from MaxHeap. Give an example. b) Write a C program to implement an AVL Tree with the following functionalities: Insertion of a node (Ensure the balance factor is maintained). Left and Right Rotation functions to maintain balance. Display the tree in In-order traversal to verify balancing. Requirements: (1) Explain the importance of the balance factor. (2) Demonstrate with an example how an imbalance is corrected using rotations. <i>(Full correct implementation: 10 marks, Partial correctness: 6-8 marks, Minor issues: 3-5 marks)</i>	5	CO1
		10	CO3
3.	Problem Solving (Analysis & Evaluation Level): A traffic navigation system needs to efficiently manage road networks where each intersection is a node, and roads are weighted edges (distance/time). The system should be able to: Find the shortest path between two locations. Identify the best route when certain roads are closed due to maintenance. Task: Propose a Graph-based model for this system. Explain how this Graph can be used for shortest path calculation. Draw a diagram to represent the structure of this system. <i>(Solution approach: 5 marks, Diagram: 3 marks, Explanation: 2 marks)</i>	10	CO3

Good Luck