United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Final Exam: Summer 2025
Course Code: CSE 2215, Course Title: Data Structures and Algorithms 1
Total Marks: 40 Duration: 2 hours

Answer all questions. Marks are indicated in the right side of each question.

[Any examinee found adopting unfair means will be expelled from the trimester/program as per UIU disciplinary rules.]

---

1. a. Determine whether the following postfix expression is valid or not with proper justification: [3]

5 3 + 6 * 10 2 / - +

b. Consider the following postfix expression: **100 P 2 * + 10 -**

After evaluating the expression, the result is 100. Now determine the value of P for this result. [3]

c. Is the Tower of Hanoi puzzle solvable without the auxiliary pillar? Explain briefly. [2]

2. Your friend Rafi is building a simple **Student Record App** where student IDs are stored in a **Binary Search Tree (BST)** for quick searching and sorting. While testing, he faces some problems and asks for your help.

a. The student IDs were inserted in the following order:

50, 30, 70, 40, 20, 60, 80,35.

Draw the Binary Search Tree (BST) formed after all insertions. [4]

b. His system currently performs **Inorder traversal** to display student IDs in ascending order. Explain how to modify the traversal logic so that it prints the IDs in **descending order**. [2]

c. Later, Rafi wants to **delete the student record with ID 50**. Draw the BST after deleting this node.[2]

3. In an eSports tournament, the organizer keeps track of the top players based on their scores. To efficiently update and retrieve the highest-scoring player, the system uses a max-heap data structure.

The current player scores are as follows: 10, 20, 30, 40, 25, 50, 35

a. Draw the complete binary tree corresponding to the given scores in the order listed (before applying the heap property). [1]

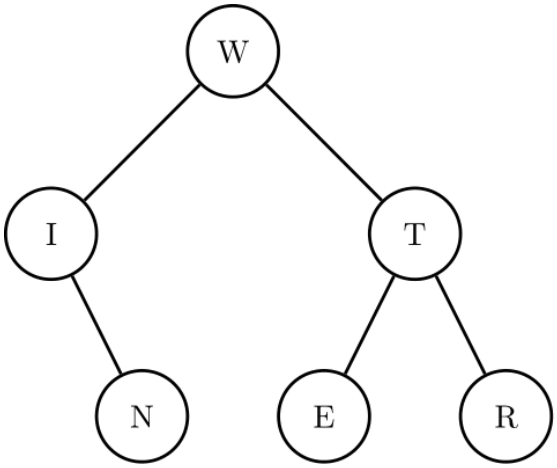
b. Build a max-heap from this tree using the Build-Heap (bottom-up) approach. Show intermediate heapify steps. [2]

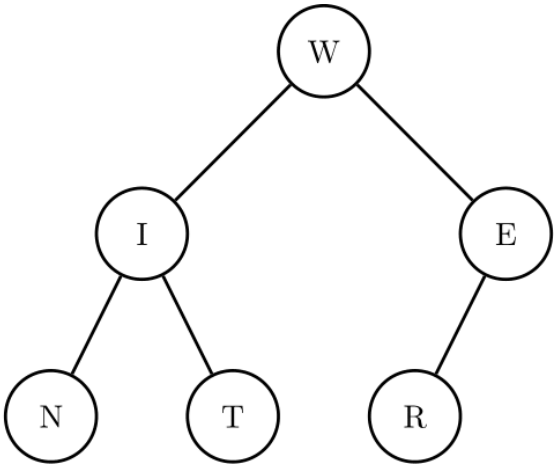
c. A new player joins with a score of 60. Insert this score into the heap and show the updated tree after re-heapification. [2]

d. The current top player is disqualified. Delete the root node (top score) from the heap and show the final tree after re-heapification. [2]

e. Briefly explain why the tournament system prefers the Build-Heap approach over inserting elements one by one into an empty heap. Compare their time complexities and efficiency for large datasets. [1]

4. a. The pre-order traversal result of the two following trees is “WINTER”.





Construct 5 **new binary trees** that have the same pre-order traversal result. [5]

b. A complete binary tree can have a linked representation and an array representation. The following table shows the correspondence between the path from the root and the array index (1-indexed).

Index	**Path from root**
1	root
2	root → left
3	root → right
4	root → left → left
5	root → left → right
7	A
B	root → left → left → left → left
20	C

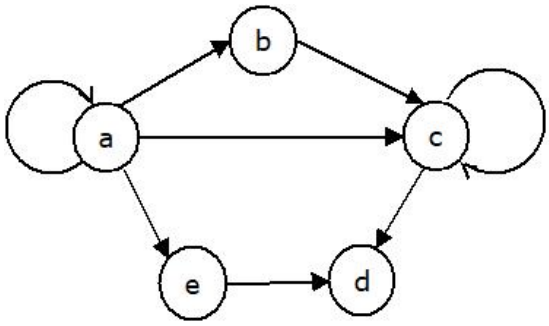
Determine A, B, and C. [3]

Note:

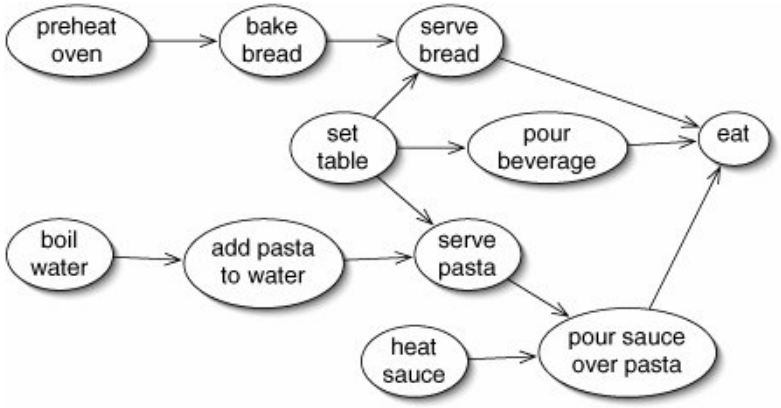
For a 1-indexed complete binary tree:

- The index of the left child of node-i is 2i.
- The index of the right child of node-i is 2i+1.
- The index of the parent of node-i is $\left\lfloor \frac{i}{2} \right\rfloor$.

5. a. Represent the following graph using adjacency matrix and adjacency list : [3]



b. Consider the following graph showing some task dependencies:



Perform **DFS** on the graph so that all nodes are visited. Show the discovery time and the finishing time for each node. [3]

c. Determine a valid **topological sorting** of the tasks from question 5(b). [2]