National University of Computer and Emerging Sciences

Lahore Campus

0208

Data Structures (CS2001)

Midterm 2 Exam

Date: Mon, 4 Nov 2024

Course Instructor(s)

ZA, SK, MN, SF, AK, MM, UN, UH, AK, FA

Total Time (Hrs.): 1

Total Marks: 20

Total Questions: 3

23L-0501
Roll No

BCS-3E
Section

8+5+5

Student Signature

Note: Solve all questions in the space provided and submit it to the invigilator.

CLO # 1: Demonstrate basic concepts of data structure and algorithms

Q. No 1: **(Marks: 2+8)**

You are given two integer ***Binary Trees (BTs)***, A and B that store ***unique integers***. Your task is to determine whether B is a subtree of A. Specifically, handle the following cases: size (A) ≤ size(B)

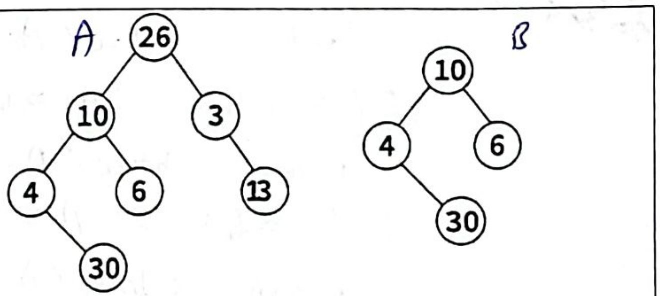
- B is a proper subtree of A (meaning B is part of A, but they are not identical).
- B is an improper subtree of A (where B and A are identical). A = B bool start, ~~[illegible]~~,
- B is not a subtree of A.

The algorithm should have a time complexity of O(n), where n is the number of nodes in A.

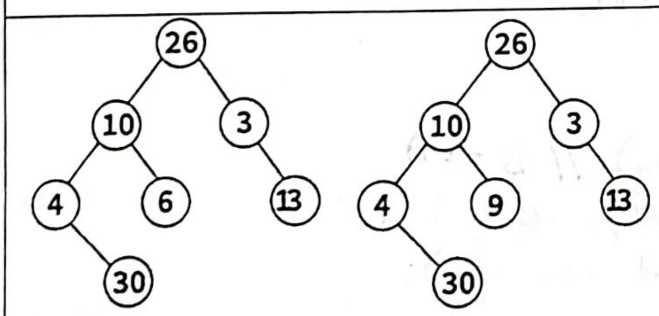
Describe the Algorithm: Provide a step-by-step description of your algorithm in plain English (not more than 3-4 lines).

Implement the Algorithm: Write a time and space efficient ***recursive C++ function*** (member function of class Tree) for the algorithm, including comments explaining each step. You can assume the following implementation of the class Node.

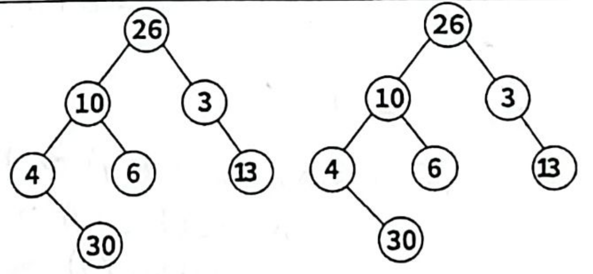
```
class Node {
  private:
    int data;
    Node *left;
    Node *right;
    Node(int x):data(x),left(nullptr),right(nullptr) {}
};
```



B is a proper subtree A



B is not a subtree of A



B is a improper Subtree of A

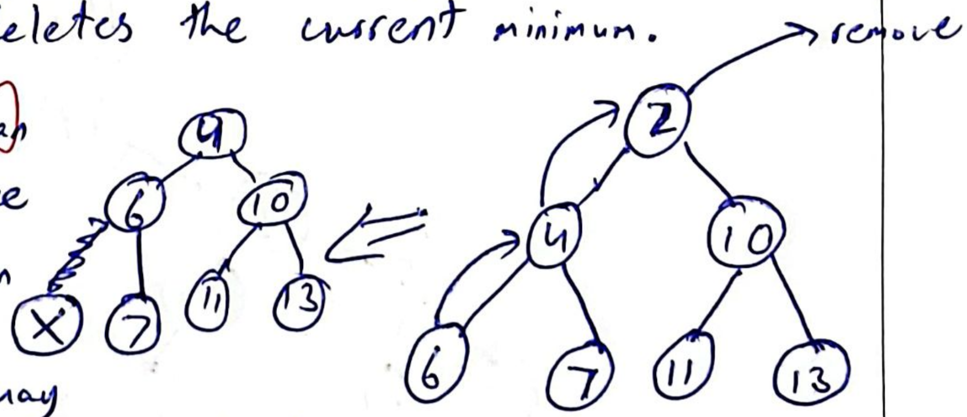
CLO # 4: Determine bugs in programs and recognize required operations with data structures.

Q2: **(Marks: 5)**

Suppose we modify the extract-min operation in a min-heap. Instead of moving the last element of the heap to the root and then performing the heapify operation, we move the minimum child of the root node to the root. We then recursively apply this process to the affected subtree, moving the minimum child to the root of that subtree and continuing until the entire heap is heap-ordered. Explain why this approach is incorrect and give a sample heap where this approach fails to correctly perform the extract-min operation.

Supposing this procedure is ~~so~~ like a bubble down modified function which finds the next minimum in the heap and replaces it with the current, then deletes the current minimum.

From the given example, we can see that there is no guarantee that the next minimum value in the heap exists at the last element of the heap, which may cause the continuity of the heap in the leaf nodes to break.



National University of Computer and Emerging Sciences

Lahore Campus

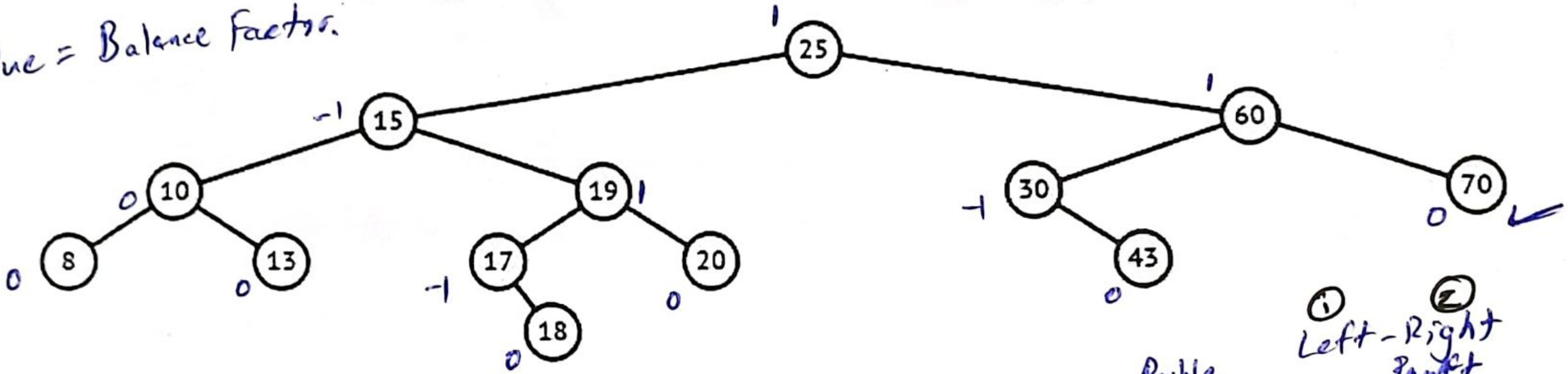
0208

CLO # 1: Demonstrate basic concepts of data structure and algorithms

Q3: **(Marks: 5)**

You are given an AVL tree. Your task is to delete the node with the value 70 from the tree. During the deletion process, write down the balance factor of each node at every step. Clearly indicate any rotation cases that occur to maintain the AVL tree's balance after the deletion and show all intermediate steps.

Blue = Balance Factor.



① ② Left-Right

Double

② (Rotate ~~right~~ Left at)