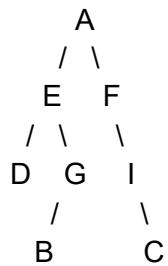


<https://github.com/boshma/Homework2.git>

Bogdan Shmat

1. [3] Given the following pre-order and in-order traversals, reconstruct the appropriate binary search tree. NOTE: You must draw a single tree that works for both traversals. Pre-order: A, E, D, G, B, F, I, C In-order: D, E, B, G, A, F, I, C



2. [3] Starting with an empty BST, draw each step in the following operation sequence. Assume that all removals come from the left subtree when the node to remove has two children.

Insert(5), Insert(10), Insert(2), Insert(9), Insert(1), Insert(3), Remove(5)

$$\begin{array}{c} \text{Step7: } 3 \\ \quad / \quad \backslash \\ \quad 2 \quad 10 \\ \quad / \quad / \\ 1 \quad 9 \end{array}$$

3. [3] Starting with an empty BST, draw each step in the following operation sequence. Assume that all removals come from the right subtree when the node to remove has two children.

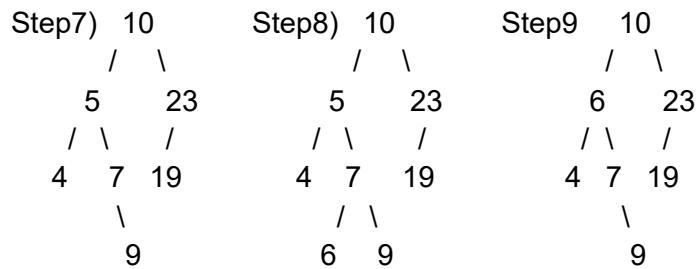
Insert(10), Insert(5), Insert(23), Insert(4), Insert(19), Insert(7), Insert(9), Insert(6), Remove(5).

```

Step1) 10
Step2) 10
Step3) 10
Step4) 10
Step5) 10
Step6) 10

      /   \
     5   23
    / 
   4 
  / \
 4   19
 / \
4   7  19

```



4. Given the following binary tree (where nullptr height == -1):

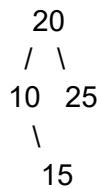
- A. [1] What is the height of the tree? 4
- B. [1] What is the depth of node 90? 3
- C. [1] What is the height of node 90? 1
- D. [3] Give the pre-order, in-order, and post-order traversal of this tree.

In Order: 0001, 0003, 0020, 0050, 0052, 0080, 0083, 0090, 0099, 0100, 0125, 0150, 0152

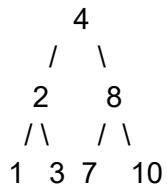
Pre Order: 0100, 0050, 0003, 0001, 0020, 0080, 0052, 0090, 0083, 0099, 0150, 0125, 0152

Post Order: 0001, 0020, 0003, 0052, 0083, 0099, 0090, 0080, 0050, 0125, 0152, 0150, 0100

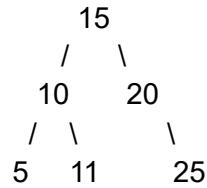
5. [3] Remove 5 from the following AVL tree; draw the results:



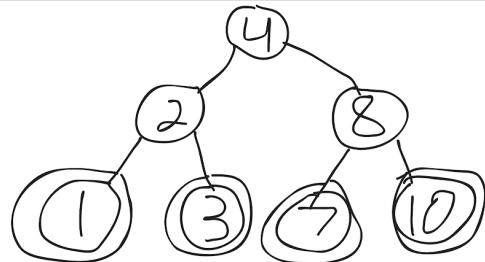
6. [3] Insert the value "8" into the following AVL tree; draw the result:



7. [3] Insert the value "11" into the following AVL tree; draw the result:

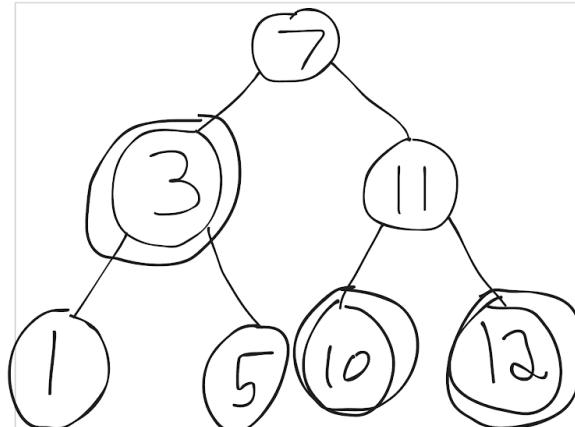


8. [3] Insert the value "8" into the following Red-Black tree; draw the result. Use Double-circle to denote red nodes and single circle to denote black nodes.



 made with
ZITEBOARD

9. [3] Delete the value "2" from the following Red-Black tree; draw the result. Use Double-circle to denote red nodes and single circle to denote black nodes.



 made with
ZITEBOARD

10. [4] Given the following B+ tree ($M = 3$, $L = 3$):

A) Insert 60 into the tree and draw the resulting B+ Tree:

```
[10,      53,      60      ]
 |      |      |
[5,8,9]  [10,11,39]  [53]  [60,98,100]
```

B) Based on the tree resulting from part (A), now remove 10 and draw the new tree:

```
[9,      53,      60      ]
 |      |      |
[5,8,]  [9,11,39]  [53]  [60,98,100]
```