PA 6 Part 1: BST Worksheet

DSC 30 Spring 2024

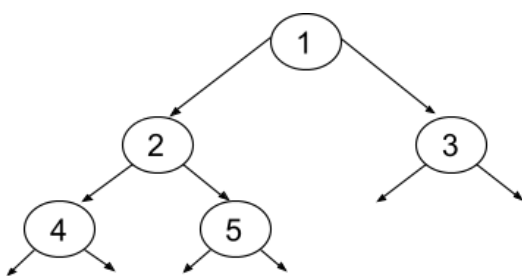| Name	Clayton Hammen Tan
PID	A17819097

First, you will perform several operations by hand so that you become familiar with the BSTs before coding them. For each part you should have solutions in two forms: hand-written and Java function (similar to the PA5 part 1).
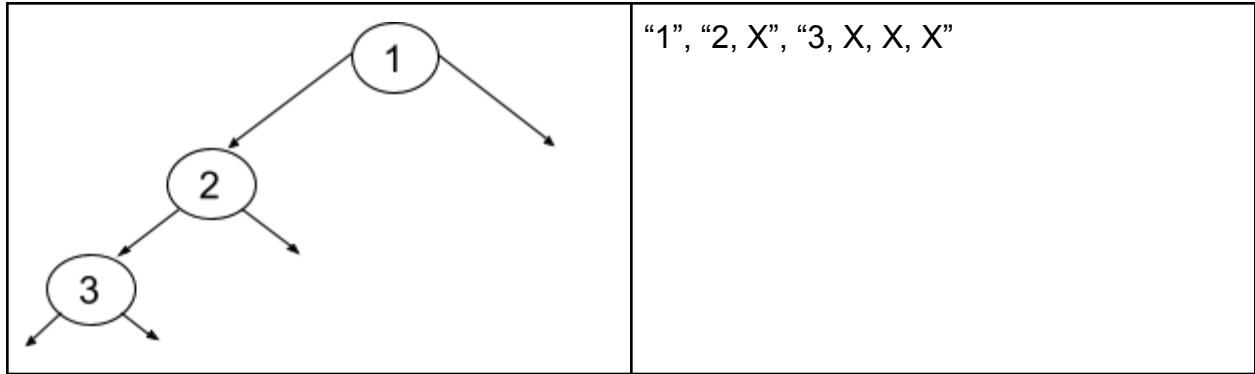
To represent a BST as a set of strings, you should treat each level as one string with the appropriate number of nodes per level.

- 1 node per level (level 0)

- 2 nodes per level (level 1)

- 4 nodes per level (level 2)
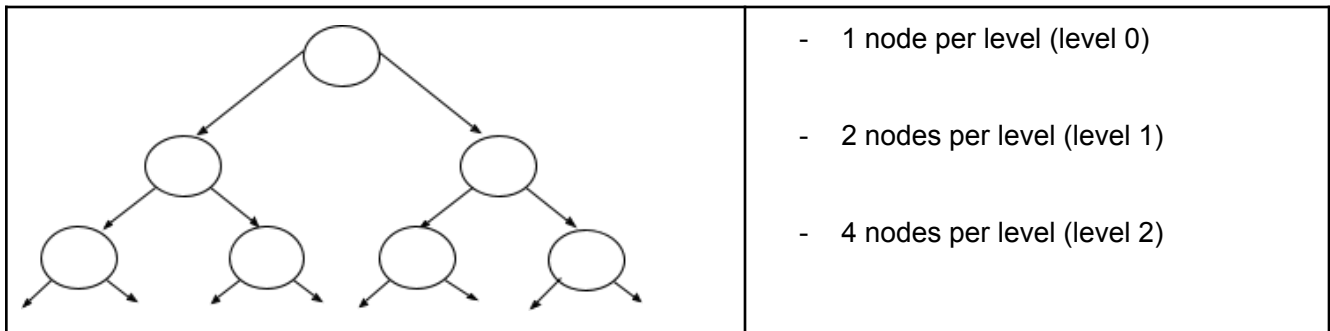
If the node is missing in a level, then use X to represent the missing node.

For example,

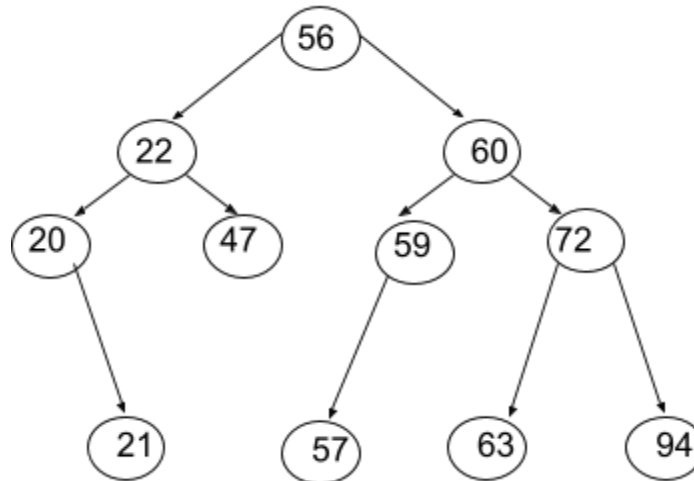| Binary Tree	Corresponding representation
	"1", "2, 3", "4, 5, X, X"

"1", "2, X", "3, X, X, X"

1. **Insert** the following integers in the order presented to an empty BST. Then draw the BST after these insertions in the box below, and then add it in a string format to BSTManual.java. You don't need to show each step. Your Java answer should "mimic" your drawing.

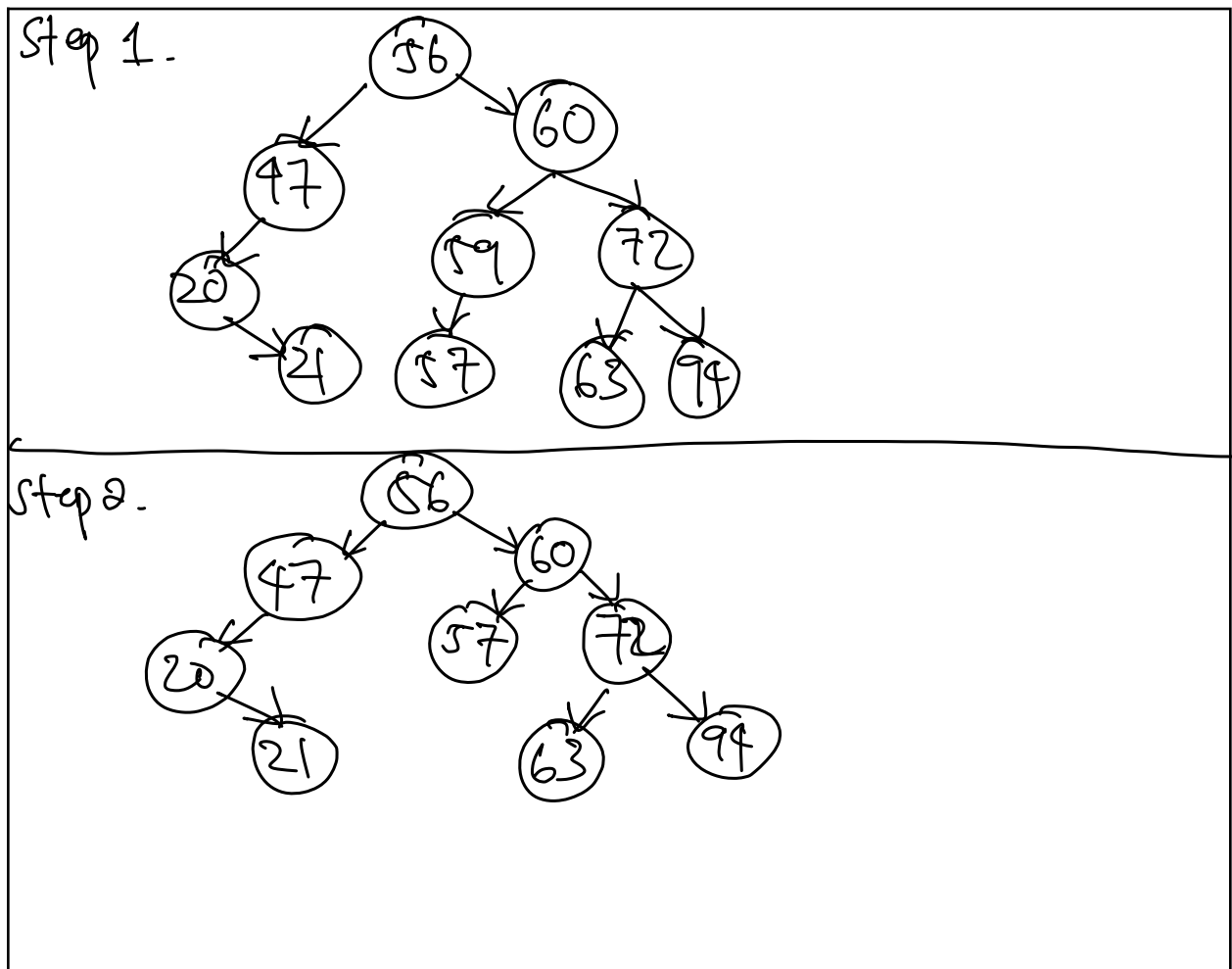[50, 15, 37, 38, 61, 65, 13, 25, 92, 21]
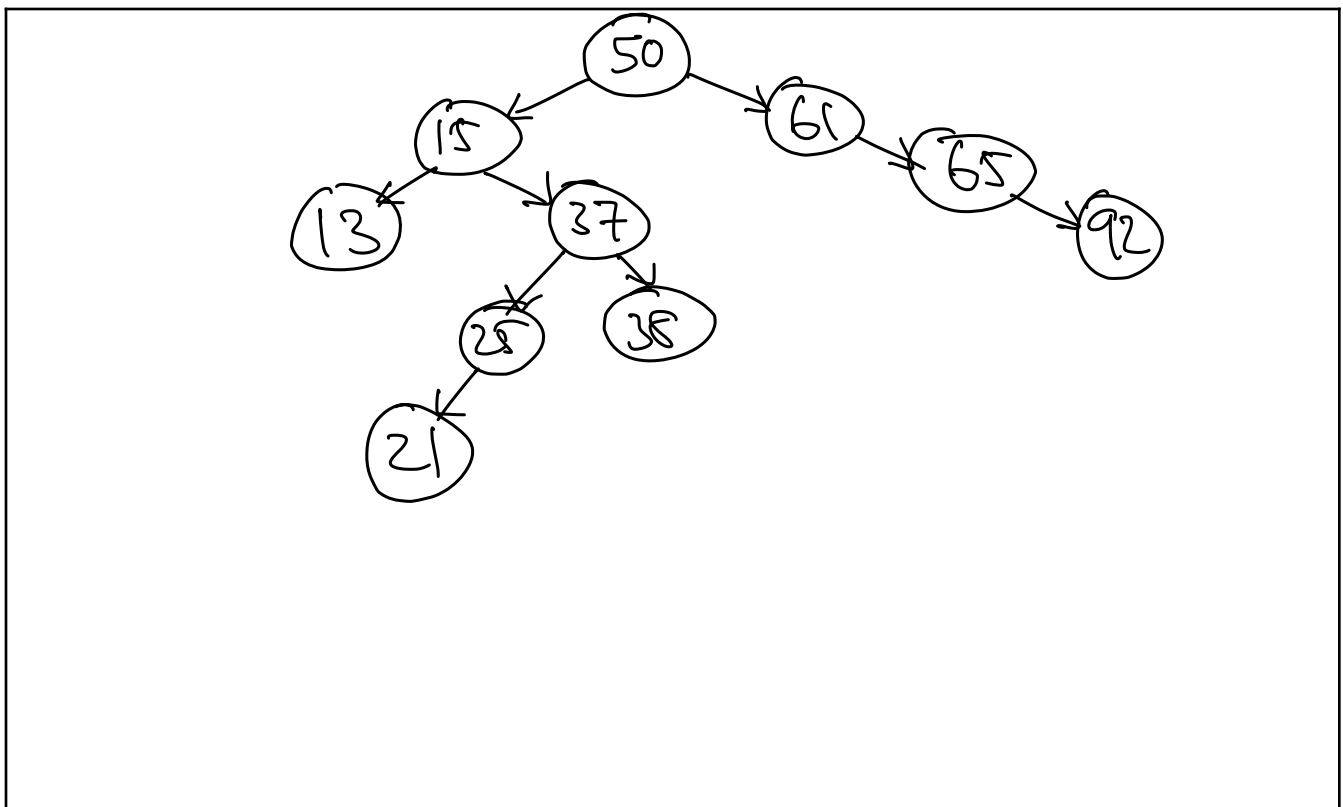
2. **Remove** the following integers in the order presented from the given BST and show each step of removal by drawing the BSTs after each step of removal in the box below (i.e. 5 trees in total). When removing a node with 2 children, you must
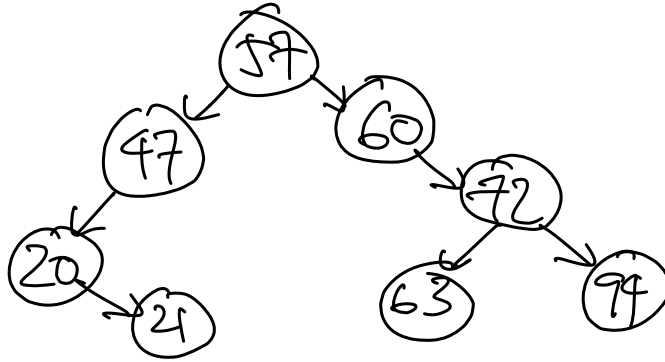
replace it with its **in-order successor** to get credits. Then add the resulting tree in a string format to BSTManual.java.

```
                    56
                   /  \
                 22    60
                /  \   / \
              20   47 59  72
                \      |   / \
                21    57  63  94
```

Elements to remove: [22, 59, 56, 72, 63]

Step 1.

```
              56
             /  \
           47    60
          /      / \
        20      59  72
          \     |   / \
          21   57  63  94
```

Step 2.

```
              56
             /  \
           47    60
          /      / \
        20      57  72
          \          \
          21          94
                     /
                   63
```

Step 3.

Binary search tree with root 57, left child 47, right child 60. 47 has left child 20, 20 has right child 21. 60 has right child 72, 72 has left child 63 and right child 94.

Step 4.

Binary search tree with root 57, left child 47, right child 60. 47 has left child 20, 20 has right child 21. 60 has right child 94, 94 has left child 63.

Step 5.

Binary search tree with root 57, left child 47, right child 60. 47 has left child 20, 20 has right child 21. 60 has right child 94.
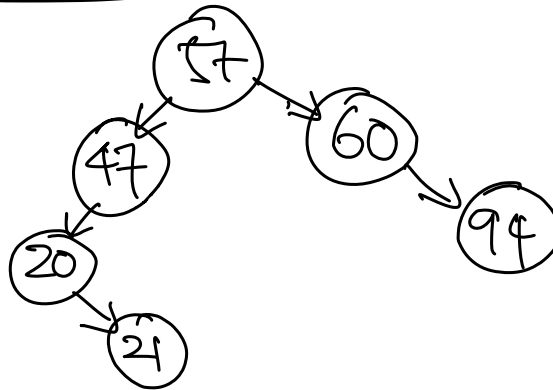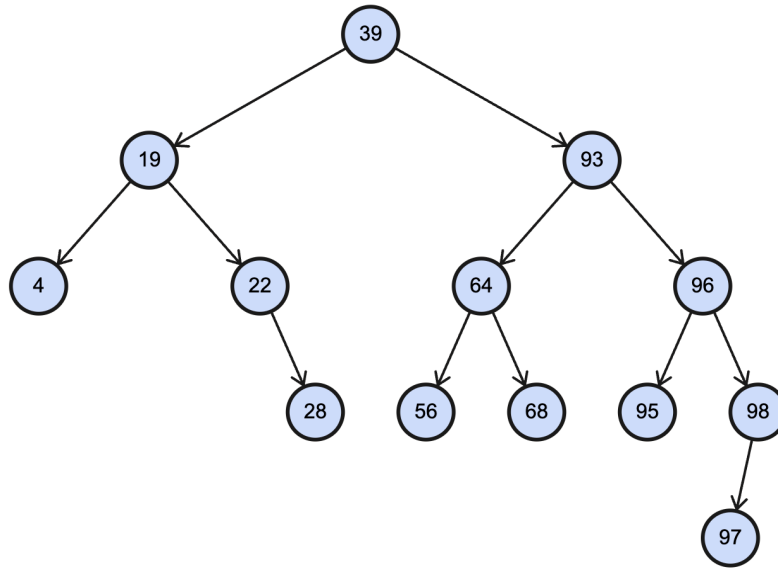
3. Write down the in-order, pre-order, and post-order traversal output of the following BST, then also add your answer to BSTManual.java in a string format: (number, comma, space, number, comma, space etc, like this: "66, 77, 88"

In-order	4,19, 22,28 , 39 , 56 , 64, 68 , 93 , 95 , 96 , 97 , 98
Pre-order	39, 19 , 4 , 22, 28 , 93 , 64 , 56, 68 , 96 , 95 , 98 , 97
Post-order	4, 28 , 22 , 19 , 56 , 68 , 64 , 95 , 97 , 98 , 96 , 93 , 39