CS 367 Homework 4

Mingren Shen (mshen32@wisc.edu)

Question 1

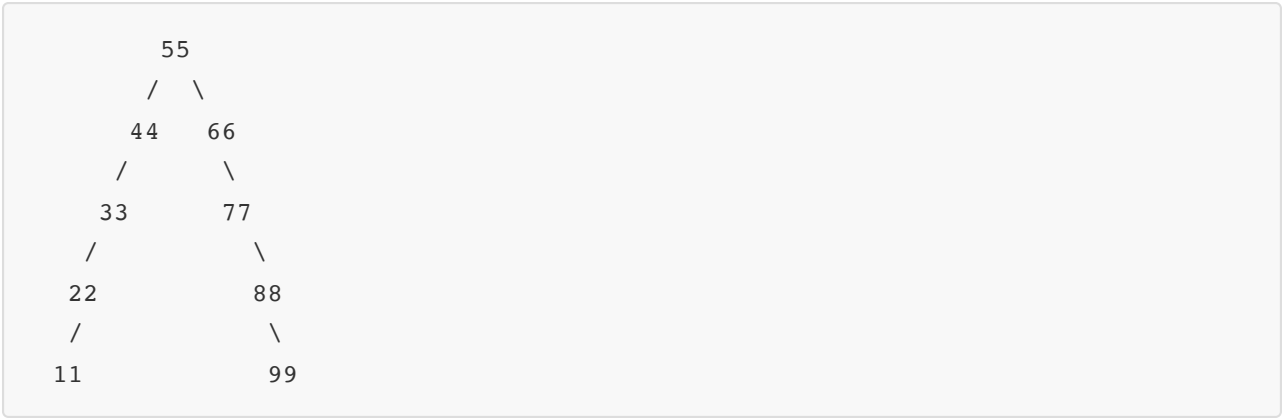
Assume that **general trees** are implemented using a `Treenode` class that includes the following fields and All parts of this question refer to standard binary search trees. The next question will deal with red-black trees, but for this question assume you are using simple binary search trees with no extra balancing logic.

Part A

Show the binary search tree that results from inserting the following sequence of integers into a tree that is initially empty:

```
55  44   33   22   11   66   77   88   99
```

Answer

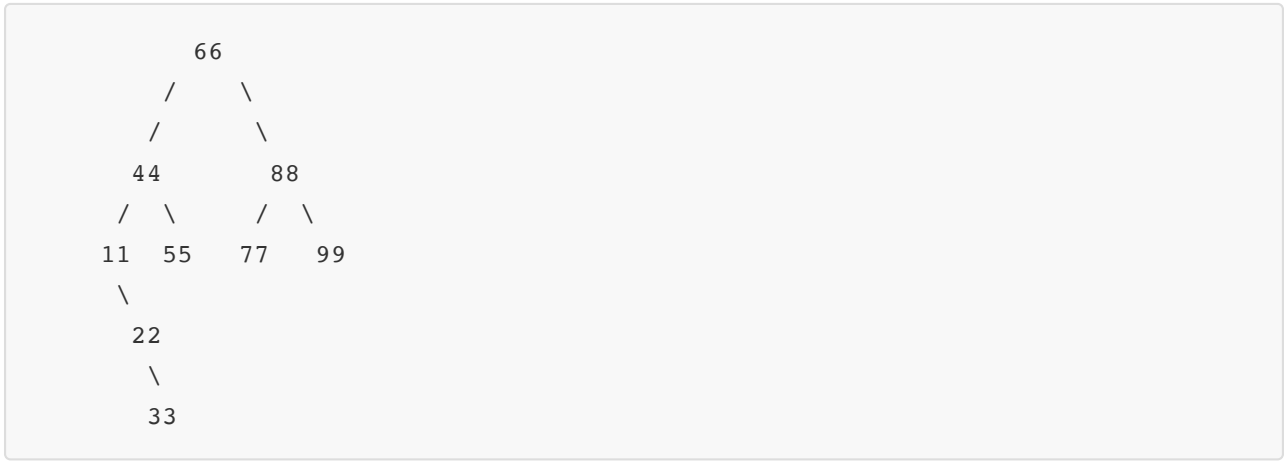


Part B

Show the binary search tree that results from inserting the following sequence of integers into a tree that is initially empty:

```
66   44   88   11   77   99   55   22   33
```

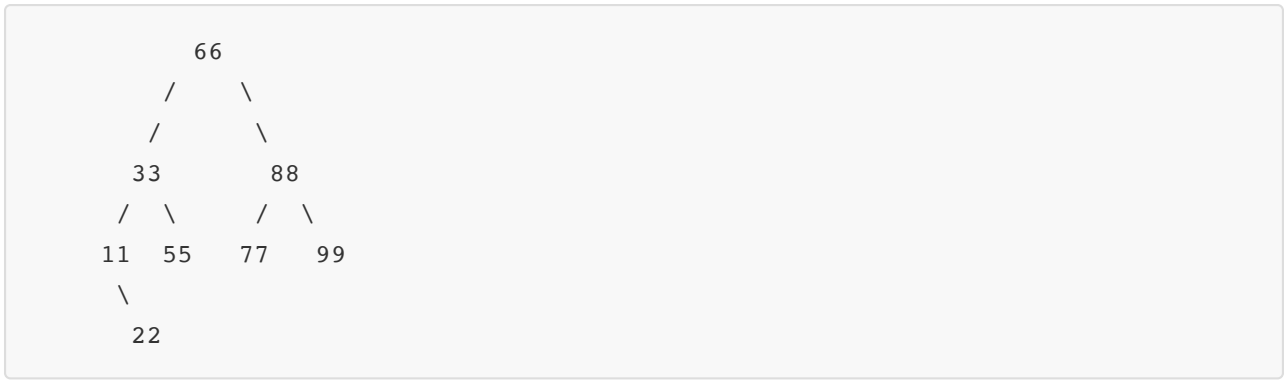
Answer



Part C

Show the binary search tree that results from deleting 44 from the tree in part B using the **in-order predecessor**.

Answer



Part D

Show the binary search tree that results from deleting 66 from the tree in part B using the **in-order successor**.

Answer

```
          77
        /    \
       /      \
      44       88
     /  \       \
    11  55       99
     \
      22
       \
        33
```

Question 2

All parts of this question refer to red-black tree. If you are creating a text-file containing your solution, indicate **red nodes** by using square brackets around the value (e.g., `[44]`) and indicate **black nodes** by not using any brackets around the value (e.g., `44`). If you are creating your solution by hand on a piece of paper, indicate red nodes by drawing a square around the value and indicate **black nodes** by drawing a circle around the value.

Part A

Show the red-black tree that results from inserting the following sequence of integers into a tree that is initially empty:

```
55  44   33   22   11   66   77   88   99
```

Answer

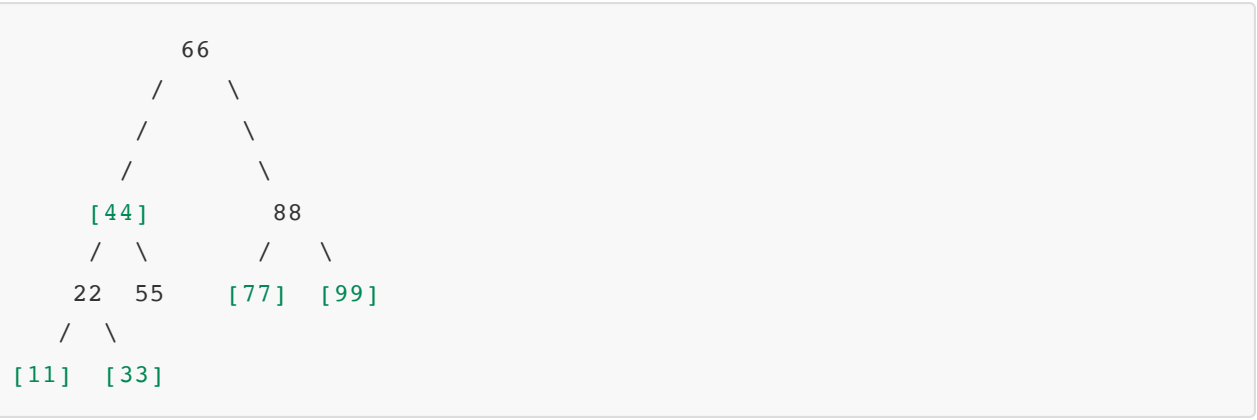
```
           44
         /    \
        /      \
       /        \
      22        [66]
     /  \       /   \
[11]   [33]   55   88
                  /  \
               [77]  [99]
```

Part B

Show the red-black tree that results from inserting the following sequence of integers into a tree that is initially empty:

```
  55  44   33   22   11   66   77   88   99
```

Answer



Question 3

Part A

Assume that a priority queue is implemented using a ***max*** heap. **Show the contents** of the max heap array that results from enqueuing (inserting) the following sequence of integer priorities into a heap that is initially empty:

6 44 20 27 73 34 10 22 89

Assume the array begins with 10 elements. Show your final answer in the form of an array, not as a binary tree, leaving any unused array slots blank.

Answer

index	0	1	2	3	4	5	6	7	8	9
array		89	73	34	44	27	20	10	6	22

Part B

Assume that a priority queue is implemented using a ***min*** heap and the following shows the contents of the array, with slot 0 going unused:

```
index   |0 | 1 | 2  | 3 | 4  | 5  | 6  | 7  | 8  | 9  |
        ---------------------------------------------
array   |  | 3 | 20 | 7 | 24 | 41 | 15 | 32 | 56 | 72 |
        ---------------------------------------------
```

Show the contents of the min heap array after **three dequeue (removeMin)** operations are done. Show your final answer in the form of an array, not as a binary tree, leaving any unused array slots blank.

Answer

```
index   |0 | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  |
        -----------------------------------------------
array   |  | 20 | 24 | 32 | 56 | 41 | 72 |    |    |    |
        -----------------------------------------------
```