

Our Solution(s)

Run Code

Solution 1

Solution 2

```
1 # Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class BST:
4     def __init__(self, value):
5         self.value = value
6         self.left = None
7         self.right = None
8
9     # Average: O(log(n)) time | O(1) space
10    # Worst: O(n) time | O(1) space
11    def insert(self, value):
12        currentNode = self
13        while True:
14            if value < currentNode.value:
15                if currentNode.left is None:
16                    currentNode.left = BST(value)
17                    break
18                else:
19                    currentNode = currentNode.left
20            else:
21                if currentNode.right is None:
22                    currentNode.right = BST(value)
23                    break
24                else:
25                    currentNode = currentNode.right
26        return self
27
28    # Average: O(log(n)) time | O(1) space
29    # Worst: O(n) time | O(1) space
30    def contains(self, value):
31        currentNode = self
32        while currentNode is not None:
33            if value < currentNode.value:
34                currentNode = currentNode.left
35            elif value > currentNode.value:
36                currentNode = currentNode.right
37            else:
38                return True
39        return False
40
41    # Average: O(log(n)) time | O(1) space
42    # Worst: O(n) time | O(1) space
43    def remove(self, value, parentNode=None):
44        currentNode = self
45        while currentNode is not None:
46            if value < currentNode.value:
47                parentNode = currentNode
48                currentNode = currentNode.left
49            elif value > currentNode.value:
50                parentNode = currentNode
51                currentNode = currentNode.right
52            else:
53                if currentNode.left is not None and currentNode.right is not None:
54                    currentNode.value = currentNode.right.getMinValue()
55                    currentNode.right.remove(currentNode.value, currentNode)
56                elif parentNode is None:
57                    if currentNode.left is not None:
58                        currentNode.value = currentNode.left.value
59                        currentNode.right = currentNode.left.right
60                        currentNode.left = currentNode.left.left
61                    elif currentNode.right is not None:
62                        currentNode.value = currentNode.right.value
63                        currentNode.left = currentNode.right.left
64                        currentNode.right = currentNode.right.right
65                else:
66                    # This is a single-node tree; do nothing.
67                    pass
68                elif parentNode.left == currentNode:
69                    parentNode.left = currentNode.left if currentNode.left is not None else None
70                elif parentNode.right == currentNode:
71                    parentNode.right = currentNode.left if currentNode.left is not None else None
72                break
73        return self
74
75    def getMinValue(self):
76        currentNode = self
77        while currentNode.left is not None:
78            currentNode = currentNode.left
79        return currentNode.value
```

Your Solutions

Run Code

Solution 1

Solution 2

Solution 3

```
1 # Do not edit the class below except for
2 # the insert, contains, and remove methods.
3 # Feel free to add new properties and methods
4 # to the class.
5 class BST:
6     def __init__(self, value):
7         self.value = value
8         self.left = None
9         self.right = None
10
11    def insert(self, value):
12        # Write your code here.
13        # Do not edit the return statement of this method.
14        return self
15
16    def contains(self, value):
17        # Write your code here.
18        pass
19
20    def remove(self, value):
21        # Write your code here.
22        # Do not edit the return statement of this method.
23        return self
24
```

Custom Output

Raw Output

Submit Code

**Run or submit code when you're ready.**