

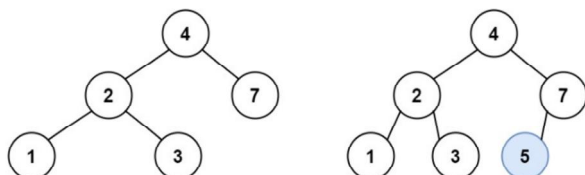
701. Insert into a Binary Search Tree

01 April 2022 11:29 AM

You are given the `root` node of a binary search tree (BST) and a `value` to insert into the tree. Return *the root node of the BST after the insertion*. It is **guaranteed** that the new value does not exist in the original BST.

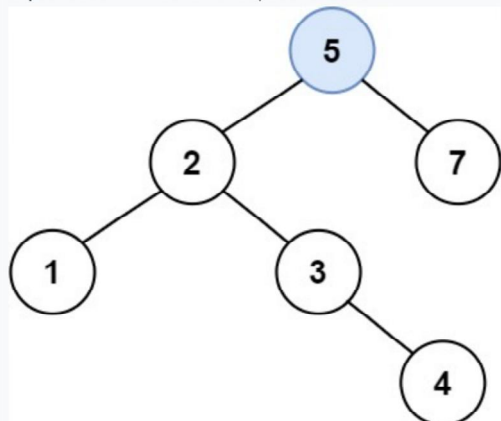
Notice that there may exist multiple valid ways for the insertion, as long as the tree remains a BST after insertion. You can return **any** of them.

Example 1:



Input: root = [4,2,7,1,3], val = 5
Output: [4,2,7,1,3,5]

Explanation: Another accepted tree is:

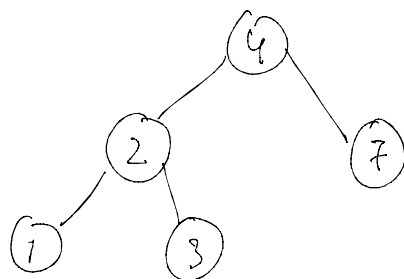


Example 2:

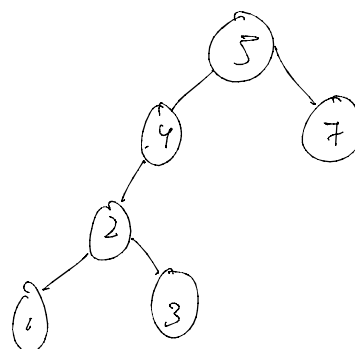
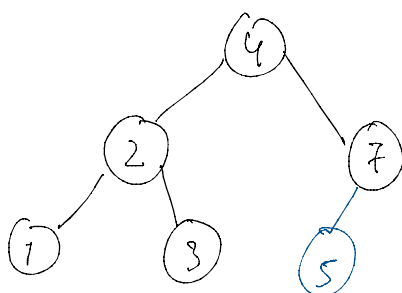
Input: root = [40,20,60,10,30,50,70], val = 25
Output: [40,20,60,10,30,50,70,null,null,25]

node = 5

$L < N < R$ follow this condition



o/p \Rightarrow



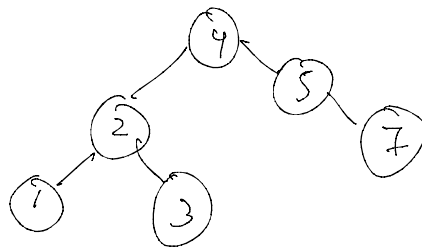
(1)

(3)

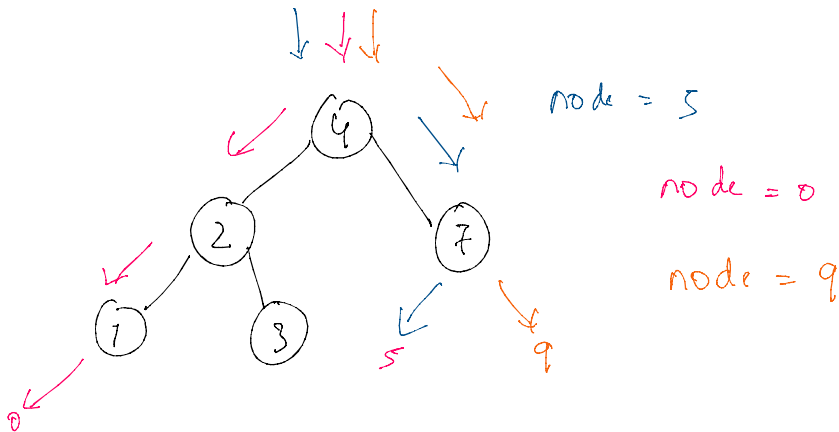
(5)

(1)

(3)

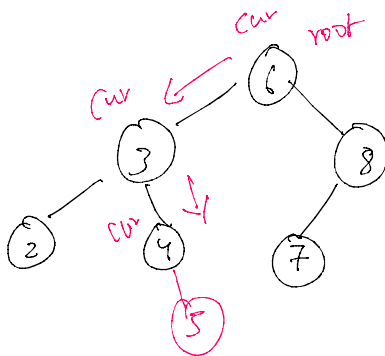


return anyone output



Find where it can be and insert
leaf position

node = 5



```

class Solution {
    public TreeNode insertIntoBST(TreeNode root, int val) {
        if(root == null) return new TreeNode(val);
        // create copy of root because at the end u need to return root itself
        TreeNode cur = root;
        while(true){
            // right side
            if(cur.val <= val){
                if(cur.right != null) cur = cur.right;
            } else {
                // if its null
                cur.right = new TreeNode(val);
                break;
            }
            // left side
            else {
                if(cur.left != null) cur = cur.left;
            } else {
                // if its null
            }
        }
    }
}
  
```

5 is not greater so its like

right comes back now cur = 3

5 > 3

now cur = 4

if its null 5 > 4 and now its null

create new node 5

not break

```

    else {
        if (cur.left != null) cur = cur.left;
        else {
            // if its null
            cur.left = new TreeNode(val);
            break;
        }
    }
}
return root;
}
}

```

W~ ↗ ✓ Check new ⑤
 out break

$$T.C \Rightarrow (\log_2 n)$$