

















Dashboard > Data Structures > Trees > Binary Search Tree : Insertion

Points: 373.91 Rank: 16552

# **Binary Search Tree: Insertion** ■





Problem

Submissions

Leaderboard

Discussions

Editorial

You are given a pointer to the root of a binary search tree and a value to be inserted into the tree. Insert this value into its appropriate position in the binary search tree and return the root of the updated binary tree. You just have to complete the function.

#### **Input Format**

You are given a function,

```
node * insert (node * root ,int value) {
}
```

node is defined as:

```
struct node
{
int data;
node * left;
node * right;
}node;
```

#### **Constraints**

• No. of nodes in the tree  $\leq$  500

## **Output Format**

Return the root of the binary search tree after inserting the value into the tree.

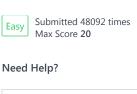
# **Sample Input**



The value to be inserted is 6.

## **Sample Output**







Rate This Challenge:

Download problem statement

Download sample test cases

**Suggest Edits** 



```
Current Buffer (saved locally, editable) & 🗸 🖸
                                                                                             Python 2
                                                                                                                               \Diamond
   ....
 1
   Node is defined as
 2
   self.left (the left child of the node)
   self.right (the right child of the node)
    self.data (the value of the node)"""
 6
 7 ▼ def insert(r,val):
 8 ▼
        if r==None:
 9
            r=Node(val)
10
             return r
        elif val>r.data:
11 ▼
            r.right=insert(r.right,val)
12
13
            return r
14 ▼
15
            r.left=insert(r.left,val)
16
            return r
17
18
                                                                                                                     Line: 16 Col: 17
```

# **1** Upload Code as File

Run Code

Submit Code

Testcase 0 🗸

# Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

### Input (stdin)

```
5
4 2 3 1 7
6
```

Your Output (stdout)

Righ	nt Answer!	
Expected Output		
Righ	nt Answer!	

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature