Ceil in BST

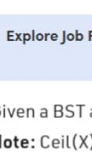
Medium Accuracy: 62.73% Submissions: 37K+ Points: 4

Given a BST and a number **X**, find **Ceil of X**.

Note: Ceil(X) is a number that is either equal to X or is immediately greater than X.

Example 1:

Input:



X = 3

Output: 3

C++ (g++ 5.4) Average Time: 30m Start Timer

```
// } Driver Code Ends


// User function Template for C++

// Function to return the ceil of given number in BST.
int findCeil(Node* root, int input) {
    if(root==NULL) return -1;
    int res = -1;
    while(root != NULL){
        if(root->data == input)
            return root->data;
        else if(root->data < input){
            root = root->right;
        }else{
            res = root->data;
            root = root->left;
        }
    }
    return res;

    // Your code here
}
```

Custom Input | Compile & Run | Submit

Floor in BST

Medium Accuracy: 57.7% Submissions: 6K+ Points: 4

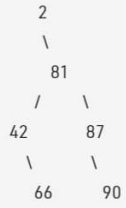
You are given a BST(Binary Search Tree) with **n** number of nodes and value **x**. your task is to find the greatest value node of the BST which is smaller than or equal to x.

Note: when x is smaller than the smallest node of BST then returns -1.

Example:

Input:

n = 7



C++ (g++ 5.4) Average Time: 20m Start Timer

```
// } Driver Code Ends


// Function to search a node in BST.
int floor(Node* root, int x) {
    // Code here
    if(root==NULL) return -1;
    int res=-1;
    while(root!=NULL){
        if(root->data==x)
            return root->data;
        else if(root->data < x){
            res=root->data;
            root=root->right;
        }
        else{
            root=root->left;
        }
    }
    return res;
}
```

Custom Input | Compile & Run | Submit