



Binary Search Tree : Lowest Common Ancestor

by [vatsalchanana](#)

Problem

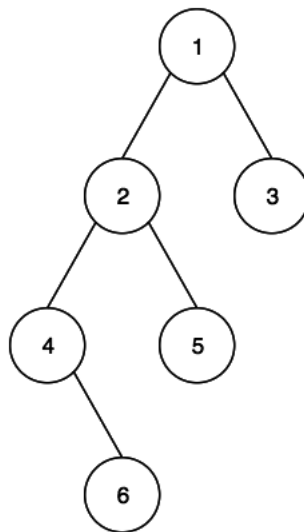
Submissions

Leaderboard

Discussions

Editorial

You are given pointer to the root of the binary search tree and two values v_1 and v_2 . You need to return the lowest common ancestor (LCA) of v_1 and v_2 in the binary search tree.



In the above example, the lowest common ancestor of the nodes **5** and **6** is the node **2**, as **2** is the lowest common node which has both the nodes **5** and **6** as it's descendants.

You only need to complete the function.

Input Format

You are given a function,

```
node * lca (node * root ,int v1,int v2) {  
    }  
}
```

It is guaranteed that v1 and v2 are present in the tree.

Node is defined as :

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

Output Format

Return the LCA of v_1 and v_2 .

Sample Input



$v_1 = 1$ and $v_2 = 7$.

Sample Output

LCA of **1** and **7** is **4** (which is the root).

Return a pointer to the root in this case.

[f](#) [t](#) [in](#)

Submissions: [37224](#)

Max Score: 30

Difficulty: Easy

Rate This Challenge:

☆☆☆☆☆

[More](#)

Current Buffer (saved locally, editable) [🔗](#) [🔄](#)

C++



```
1  /*
2  Node is defined as
3
4  typedef struct node
5  {
6      int data;
7      node *left;
8      node *right;
9  }node;
10
11  */
12
13
14  node *lca(node *root, int v1, int v2)
15  {
16
17
18
19  }
20
21
```

Line: 1 Col: 1

[📁 Upload Code as File](#)

☐ Test against custom input

Run Code

Submit Code

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)