

LCA of BST

[Send Feedback](#)

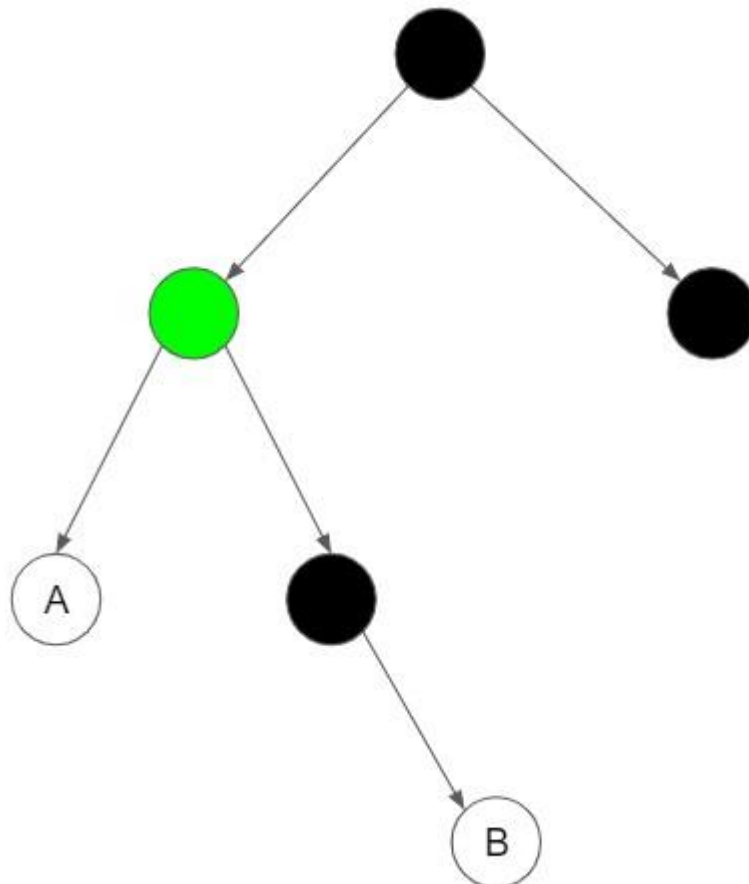
Given a binary search tree and data of two nodes, find 'LCA' (Lowest Common Ancestor) of the given two nodes in the BST.

LCA

LCA of two nodes A and B is the lowest or deepest node which has both A and B as its descendants.

Example:

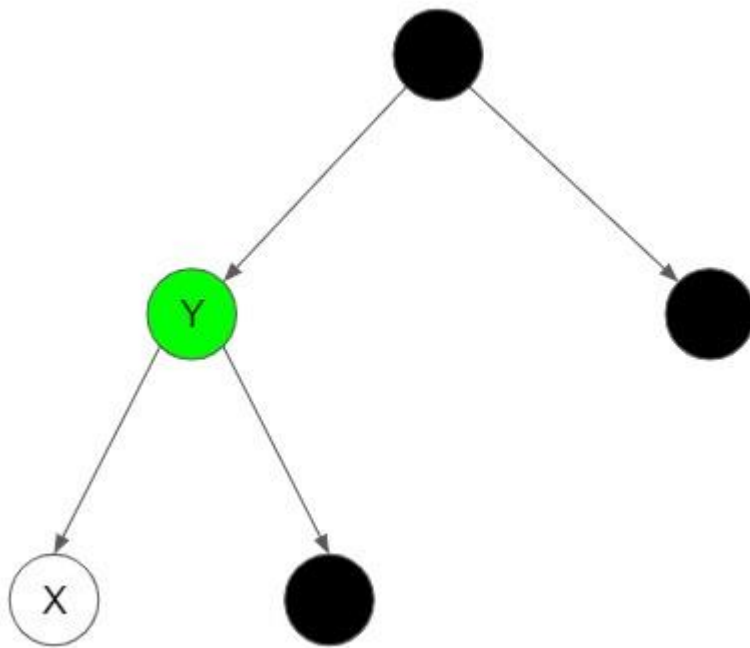
In this example, the green coloured node is the LCA to A and B.



Note:

It is defined that each node is a descendant to itself, so, if there are two nodes X and Y and X has a direct connection from Y, then Y is the lowest common ancestor.

Example:



Note:

1. If out of 2 nodes only one node is present, return that node.
2. If both are not present, return -1.

Input format:

The first line of input contains data of the nodes of the tree in level order form. The data of the nodes of the tree is separated by space. If any node does not have left or right child, take -1 in its place. Since -1 is used as an indication whether the left or right nodes exist, therefore, it will not be a part of the data of any node.

The following line of input contains two integers, denoting the value of data of two nodes of given BST.

Output Format:

The first and only line of output contains the data associated with the lowest common ancestor node.

Constraints:

Time Limit: 1 second

Sample Input 1:

```
8 5 10 2 6 -1 -1 -1 -1 -1 7 -1 -1
2 10
```

Sample Output 1:

```
8
```

Sample Input 2:

```
8 5 10 2 6 -1 -1 -1 -1 -1 7 -1 -1
2 6
```

Sample Output 2:

```
5
```

Sample Input 3:

```
8 5 10 2 6 -1 -1 -1 -1 -1 7 -1 -1
12 78
```

Sample Output 3:

```
-1
```