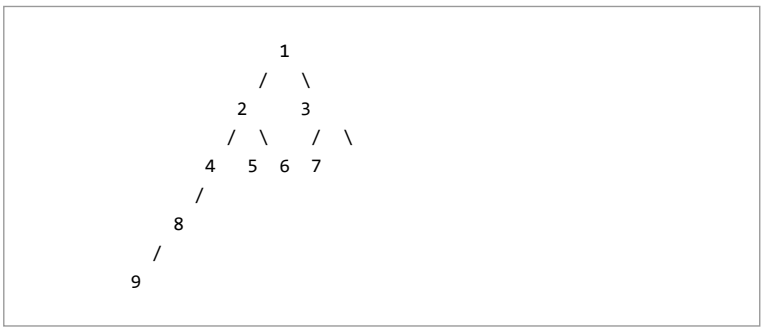
Minimum Leaf Distance

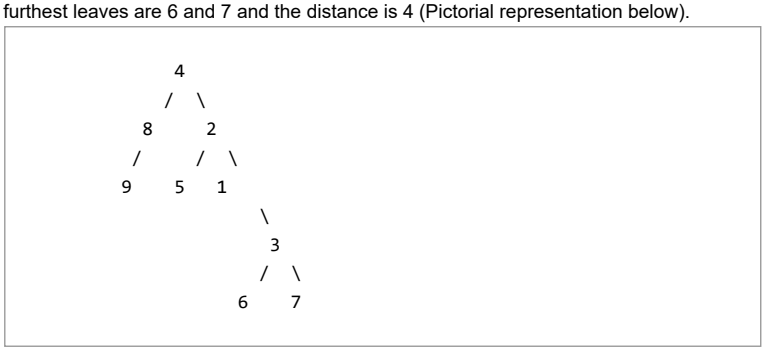
Problem Code: **MINROOTD**

This problem is worth 2 points

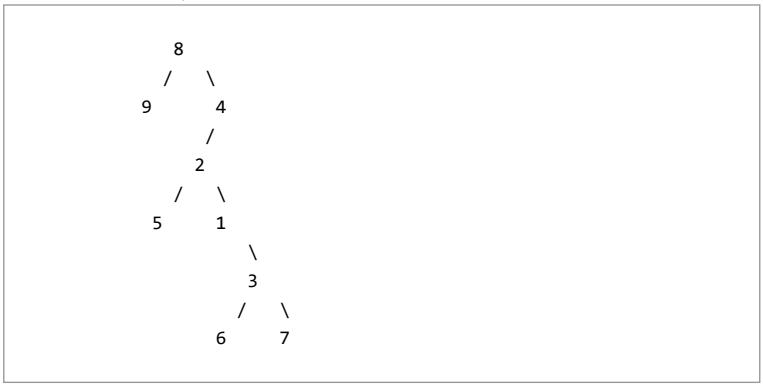
Given a binary tree, you have to find out a node of the binary tree such that when the tree is rooted at this node, the distance from this rooted node to any other leaf in the tree is minimized.



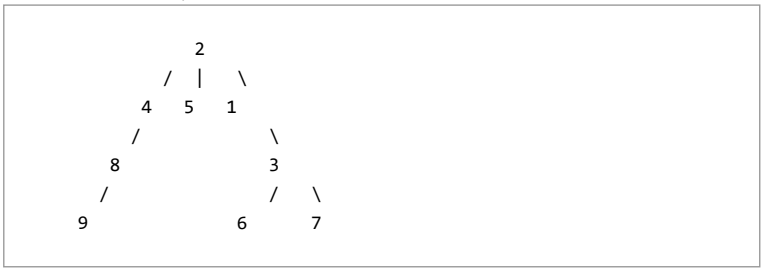
For example, for the above binary tree, if you root the tree at the node labelled 4, the furthest leaves are 6 and 7 and the distance is 4 (Pictorial representation below).



If you root the tree at the node labelled 8, the furthest leaf distance is 5. (Pictorial representation below)



If you root the tree at the node labelled 2, the furthest leaf distance is 3.(Pictorial representation below)



All Submissions
(/DI17R122/status/MINROOTD)

Successful Submissions +

If you root the tree at any other node othan than the node labelled 2, the furthest leaf distance is always greater than 3.
In the above example, when the tree is rooted at the node labelled 2, the maximum leaf distance is minimized and is 3.
Hence, the answer for the given tree is 2.

Note: If there are multiple solutions to the question, then you have find out the node with the lowest label number (label numbers are distinct among all nodes).

It is also possible that after picking a node as root, the tree may no longer remain a binary tree (as is the case with the answer above).

Constraints :

No of nodes N in the binary tree will be < 1000.
Each node has a distinct label number.

Input:

You are provided with a template function which looks like follows:

```
C++ function:
struct node {
        node *left;
        node *right;
        int label;
};

int getRootWhichMinimizesFurthestLeafDistance(node *root) {

}

Java function:
class Node {
        Node left;
        Node right;
        int label;
}

int getRootWhichMinimizesFurthestLeafDistance(Node root) {

}
```

The function has a parameter 'root' which is the reference of the root of a binary tree.
Each node has a leftChild reference and a rightChild reference and a label, which is an integer.
Each node has a distinct label associated with it.

Output:

The template function should return the answer, that is the label of that node of the tree, such that when rooted at that node, the maximum distance from the root to the leaf is minimized.
If there are multiple solutions, return the one with the lowest label number.

Sample Input

You can test your program by giving it the following input in the command line (the template parses this and converts this into a tree, and calls your method) The format is simply: First line is number of test cases. Then each test case has "number of nodes" followed by "label of root node". On the next N lines (number of nodes), there are 3 numbers; "node label" "left child node label" "right child node label". Of course -1 is used as placeholder for no left/right child. Given example is the binary tree from the problem statement.

```
1
9 1
1 2 3
2 4 5
3 6 7
4 8 -1
5 -1 -1
8 9 -1
6 -1 -1
7 -1 -1
9 -1 -1
```

Sample Output

```
2
```

Author:	directi_campus (/users/directi_campus)
Tags:	directi_campus (/tags/problems/directi_campus)
Date Added:	20-10-2012
Time Limit:	10 secs
Source Limit:	50000 Bytes
Languages:	C, CPP 4.3.2, CPP14, JAVA

Comments

Initiatives

Go for Gold (http://www.codechef.com/goforgold)

CodeChef for Schools (http://www.codechef.com/school)

Campus Chapters (http://www.codechef.com/campus_chapter/about)

Domain Registration in India (http://www.bigrock.in/) and Web Hosting (http://www.bigrock.com/web-hosting/) powered by BigRock