Play with the Nincompoops

Raaghav and Bhaskar have just learnt game theory, so to test them Nimay gave them a game with a fixed set of rules. The rules are:

1). You are given a tree (rooted at 1) and both players take turns alternatively starting with Bhaskar.

2). In each turn a player has to choose a perfect binary tree from the existing tree and remove it and the resulting tree must remain connected. Even a single node qualifies as a perfect binary tree.

3). Whoever is forced to select the root loses the game.

Let's assume these nincompoops are extremely smart and play to win. Considering both play ideally, output 0 if Bhaskar wins, otherwise 1.

The first line of the input consists of **n**, the number of nodes in the tree. The next **n-1** lines consist of **n-1** edges each stating the endpoints of the edge. Check the definition of perfect binary tree on google. Check the sample testcases for more clarity.

Input Format

n

a_1, b_1

a_2, b_2

...

a_{n-1} b_{n-1}

Constraints

1 <= n <= 10

Output Format

Output of the game as described in the question

Sample Input 0

```
1
```

Sample Output 0

```
1
```

Explanation 0

Bhaskar is forced to select the root and so he loses

Sample Input 1

```
3
1 2
1 3
```

Sample Output 1

```
1
```

Explanation 1

Bhaskar will choose a leaf node, Raaghav will choose another leaf node and then Bhaskar will be forced to choose the root. In the first move Bhaskar can even select the entire tree but then he will directly lose.

Sample Input 2

```
2
1 2
```

Sample Output 2

```
0
```

Explanation 2

Bhaskar chooses a leaf node and Raaghav is forced to choose the root, and losing.

Contest ends in a day

Submissions: 107
Max Score: 35
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable) C++14

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code