Dark Streets

🔒 locked

Problem | Submissions | Leaderboard | Discussions

Bob fears the dark and unfortuantely many street lights in his neighborhood are broken. He wants to visit some of his friends today as he feels so lonely so he needs some help. The streets are represented as a tree with root at node 1 and N Nodes, the leaves of the tree contain his friends. He needs to know which friend he can visit, given that he can't take a path that have more than M consecutive dark streets. Count the number of friends he can visit.

Notes : The tree is not binary, so the node may have multiple children, also it is not guaranteed that the edges will be given from parent to child so if node B has a parent A, the edge in the input could be B A or A B

Input Format

The first line contains two integers, N and M — the number of nodes of the tree and the maximum number of consecutive dark nodes.

The second line contains N numbers either equals to 1 if the node is dark, or 0 otherwise.

Next N- 1 lines contain the edges of the tree in the format "xi yi" (without the quotes) ($1 \le$ xi, yi $\le$ N, xi $\ne$ yi), where xi and yi are the nodes of the tree, connected by an edge.

It is guaranteed that the given set of edges specifies a tree.

Constraints

($2 \le N \le 10^5, 1 \le M \le N$)

Output Format

The number of distinct leaves of the tree with a valid path

Sample Input 0

```
4 1
1 1 0 0
1 2
1 3
1 4
```

Sample Output 0

```
2
```

Sample Input 1

```
7 1
1 0 1 1 0 0 0
1 2
1 3
2 4
2 5
3 6
3 7
```

Sample Output 1

```
2
```

C++14 ▼

```cpp
1  #include <cmath>
2  #include <cstdio>
3  #include <vector>
4  #include <iostream>
5  #include <algorithm>
6  using namespace std;
7
8
9  int main() {
10     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
11     return 0;
12 }
13
```

Line: 1 Col: 1

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