G. Tree Queries

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a tree consisting of n vertices (numbered from 1 to n). Initially all vertices are white. You have to process q queries of two different types:

1. $1\ x$ — change the color of vertex x to black. It is guaranteed that the first query will be of this type.
2. $2\ x$ — for the vertex x, find the minimum index y such that the vertex with index y belongs to the simple path from x to some black vertex (a simple path never visits any vertex more than once).

For each query of type 2 print the answer to it.

Note that the queries are given in modified way.

Input

The first line contains two numbers n and q ($3 \le n, q \le 10^6$).

Then n - 1 lines follow, each line containing two numbers x_i and y_i ($1 \le x_i < y_i \le n$) and representing the edge between vertices x_i and y_i.

It is guaranteed that these edges form a tree.

Then q lines follow. Each line contains two integers t_i and z_i, where t_i is the type of ith query, and z_i can be used to restore x_i for this query in this way: you have to keep track of the answer to the last query of type 2 (let's call this answer $last$, and initially $last = 0$); then $x_i = (z_i + last) \ mod \ n + 1$.

It is guaranteed that the first query is of type 1, and there is at least one query of type 2.

Output

For each query of type 2 output the answer to it.

Example

input
4 6 1 2 2 3 3 4 1 2 1 2 2 2 1 3 2 2 2 2

output
3 2 1