Task: STR
Strikes

XXIV OI, Stage II, Day one. Source file `str.*` Available memory: 128 MB. *15.02.2017*

Citizens of Bitotia are well-known for their short temper and deep love for democracy (as opposed to calm and peaceful subjects in royalist Bytotia). Whenever they feel the urge to give voice to their discontent with the government's decisions (e.g., to increase the number of bits in a byte) or support one (e.g., to make RAM and cache equal), they gather in the streets and go on strike.

There are n towns in Bitotia. The citizens of each independently decide on starting or ending a strike. Once started, a strike paralyzes a town completely, precluding either entering or leaving it. Unfortunately, the Bitotian road network has minimalist design, in the sense that for every pair of towns, there is a unique way to reach one from the other. Thus, the strikes often disconnect towns that are not themselves striking. More formally, a current set of cities on strike partitions the remaining cities into a number of *components* such that two towns (with no strikes taking place in either) are connected if and only if they are in the same component.

Named the commissioner for transport in Bitotia, your task is to write a program that, fed up-to-date information about currently on-going strikes, determines the number of components of the road network.

All roads in Bitotia are bidirectional, each having either end in some town.

Input

The first line of the standard input contains a single integer n ($n \geq 2$) that specifies the number of towns in Bitotia. The towns are numbered from 1 to n. Each of the $n-1$ lines that follow describes a single road segment by two integers a and b ($1 \leq a < b \leq n$), which are the numbers of two cities directly linked by this segment. Every pair of towns in Bitotia is connected by some route composed of a number of direct road segments.

The next line contains a single integer m ($m \geq 1$) which specifies the number of strike updates. Each of the following m lines contains a single integer z such that $1 \leq |z| \leq n$. If $z > 0$, this stands for a beginning of a strike in town no. z, whereas $z < 0$ stands for an end of a strike in town no. $-z$. You may assume that in every town at any time there is either no strike or exactly one; formally, no strike can start in a town where there already is one, and similarly no strike can end in a town where there is none. Initially, there is no strike in any town.

Output

Your program should print m lines to the standard output. The i-th line, for $1 \leq i \leq m$, should contain a single integer equal to the number of components of the Bitotian road network after the i-th update from the input. At a time when citizens of all town are on strike, there are 0 components.

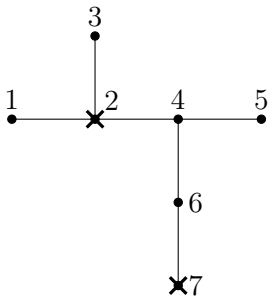
Example

For the following input data:

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

the correct result is:

```
3
3
4
3
```



Explanation for the example: The figure depicts the Bitotian road network after the strikes in town no. 2 and 7 began. The network breaks down into three components.

Sample grading tests:

`1ocen:` $n = 2$ towns, the strikes in these towns alternately begin and end ($m = 10$);

`2ocen:` a path of length $n = 1000$, each town in sequence begins its strike ($m = n$);

`3ocen:` a path of length $n = 500\,000$, successive even-numbered towns go on strike, then end their strikes in the same order.

The road network forms a *path* if for every $a = 1, \ldots, n-1$ the towns no. a and $a+1$ are directly linked.

Grading

The set of tests consists of the following subsets. Within each subset, there may be several test groups.

Subset	Property	Score
1	$n, m \leq 1000$	24
2	$n, m \leq 500\,000$, a path	17
3	$n, m \leq 500\,000$, all numbers on input are positive	17
4	$n, m \leq 500\,000$	42