

Problem A. Ants

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 512 mebibytes

You are given a tree with vertices numbered from 1 to n . There are ants in the vertices of the tree. Initially, at each vertex i , there is one ant having index i .

You will be given q queries. Each query j contains an index a_j of an ant that needs help. During query j , each ant goes to an adjacent vertex that is closest to ant a_j , or does not move if they are located at the same vertex. After each query, print the total number of pairs of ants which are located at the same vertex.

Note that the changes persist between queries: for example, when the ants have to move to ant a_2 , they are already in the positions after moving to ant a_1 . Also note that each query requires to move to **ant** a_j , which was at vertex a_j initially, but can be in some other vertex at the time of the query.

Input

The first line of input contains an integer n , the size of the tree ($2 \leq n \leq 10^5$).

Each of the next $n - 1$ lines contains two integers u and v describing an edge of the tree ($1 \leq u, v \leq n$). It is guaranteed that the edges form a tree.

The next line contains an integer q , the number of queries ($1 \leq q \leq 10^5$).

The next q lines contain integers $a_1, a_2, \dots, a_q$, one per line: the numbers of ants in the queries ($1 \leq a_j \leq n$).

Output

For each query, print a single line with the answer to it: the number of pairs of ants which are located at the same vertex after this query.

Examples

<i>standard input</i>	<i>standard output</i>
5 1 2 1 3 2 4 2 5 5 1 1 3 3 5	4 10 10 10 10
8 1 2 1 3 2 4 2 5 3 6 3 7 7 8 6 1 3 4 2 5 6	5 21 28 28 28 28