Cliques

Input file: `standard input`
Output file: `standard output`
Time limit: 5 seconds
Memory limit: 512 megabytes

Given is a tree $\mathcal{T}$ with n vertices numbered with consecutive natural numbers from 1 to n. Using it, we will create an undirected, initially empty graph $\mathcal{G}$. There are two types of operations to be performed:

- `+` v w ($1 \le v \le w \le n$) – a vertex is added to the graph $\mathcal{G}$, which is labeled with a pair of numbers (v, w).
- `-` v w ($1 \le v \le w \le n$) – one vertex labeled with the pair of numbers (v, w) is removed from the graph $\mathcal{G}$.

The pairs of numbers written in the vertices of graph $\mathcal{G}$ correspond to paths from the given tree $\mathcal{T}$ – these two numbers indicate the indices of the two ends of such a path, and they can be equal if the path consists of a single vertex.

At any given time, two vertices of the graph $\mathcal{G}$ are connected by an edge if the paths from $\mathcal{T}$ corresponding to them have at least one vertex in common. After adding a new vertex to the graph $\mathcal{G}$, edges are attached to it according to this rule, and when a vertex is removed, all the edges incident to it are also removed.

Your task is, after each operation, to output the number of non-empty subsets of vertices of the graph $\mathcal{G}$ that form cliques. A clique is a subgraph in which every pair of vertices is connected by an edge. Since this number can be very large, it's sufficient to output its remainder when divided by $10^9 + 7$.

Input

The first line of standard input contains one integer n ($2 \le n \le 200\,000$), indicating the number of vertices of the tree $\mathcal{T}$.

Each of the next $n - 1$ lines contains two integers. The numbers in the i-th of these lines are a_i and b_i ($1 \le a_i, b_i \le n$), indicating the existence in the tree $\mathcal{T}$ of an edge connecting vertices numbered a_i and b_i. It is guaranteed that the given edges describe a valid tree.

The next line contains one integer q ($1 \le q \le 50\,000$), indicating the number of modifications to the graph $\mathcal{G}$.

Each of the next q lines is of one of the two possible types:

- `+` v w ($1 \le v \le w \le n$) – a vertex is added to the graph $\mathcal{G}$, corresponding to the path between vertices v and w of the tree $\mathcal{T}$.
- `-` v w ($1 \le v \le w \le n$) – one vertex corresponding to the path between vertices v and w of the tree $\mathcal{T}$ is removed from the graph $\mathcal{G}$.

Multiple vertices in the graph $\mathcal{G}$ can have the same pair of numbers written in them. It's guaranteed that when instructed to remove a vertex with a certain pair of numbers, at least one such vertex exists in the graph $\mathcal{G}$. When instructed to remove, only one vertex with the corresponding path should be removed, even if more of them currently exist.

Output

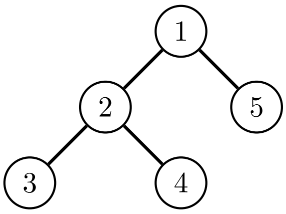
The output should contain q lines – the i-th of them should contain one integer, the number of non-empty subsets of vertices of the graph $\mathcal{G}$ that form cliques after the i-th modification. This number should be given as a remainder when divided by $10^9 + 7$.

Example

standard input	standard output
5	1
1 2	3
5 1	7
2 3	3
4 2	7
6	9
+ 4 5	
+ 2 2	
+ 1 3	
- 2 2	
+ 2 3	
+ 4 4	

Note

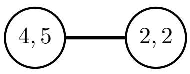
The tree $\mathcal{T}$ from the sample test looks as follows:



The following figures show the graph $\mathcal{G}$ after consecutive modifications.

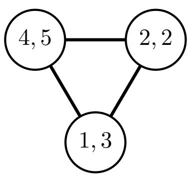
The graph $\mathcal{G}$ after the first modification:

The graph $\mathcal{G}$ after the second modification:

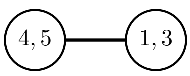


Both vertices in $\mathcal{G}$ are connected by an edge because the common vertex in $\mathcal{T}$ for both paths is vertex number 2.

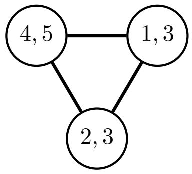
The graph $\mathcal{G}$ after the third modification:



The graph $\mathcal{G}$ after the fourth modification:



The graph $\mathcal{G}$ after the fifth modification:



The graph $\mathcal{G}$ after the last modification: