

Problem A. Christmas Tree

Program: tree.(cpp|java)
Input: tree.in
Balloon Color: Pink

Christmas time is almost here! Families are getting ready for this by buying decorated evergreen trees. The christmas tree contains n nodes numbered from 0 to $n - 1$ and rooted at node 0. Alice and Bob are playing around with the new tree and they will play a game using the tree in order to kill their boredom. Alice will carry a coloring marker and Bob will call out two types of instructions:

- $+x$, this means that Alice will color the node numbered x .
- $-x$, this means that Alice will clear the color on the node numbered x .

After each of these instructions, Alice should call out the lowest common ancestor (see note for definition) of all the colored nodes so far. Can you help Alice to reply to Bob's queries as fast as possible ?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer **T**, the number of test cases ($1 \leq T \leq 100$). Followed by **T** test cases. The first line of each test cases will contain one integer **N**, the number of nodes in the tree ($1 \leq N \leq 10^5$). The following **N - 1** lines will each contain a pair of integers **x** and **y** separated by a single space ($0 \leq x, y \leq N - 1$) which means that node x is connected to node y . It's guaranteed that the given edges will form a tree. The following line will contain one integer **Q**, the number of instructions Bob will say ($1 \leq Q \leq 4 \times 10^5$). The following **Q** lines will each contain an instruction that Bob called out in the format of $q_i a_i$ where $q_i \in \{+, -\}$ and ($0 \leq a_i \leq N - 1$). It's guaranteed that the clear instruction will only be applied on a colored node and the color instruction on an uncolored node)

Output

For each test instruction that Bob calls out, print a single line containing an integer corresponding to Alice's response to what is the lowest common ancestor of all the colored nodes or -1 if there are no colored nodes.

Examples

tree.in	standard output
1	2
10	6
1 4	0
5 4	6
1 0	6
6 8	6
6 1	1
1 3	
7 6	
9 7	
9 2	
7	
+ 2	
+ 8	
+ 0	
- 0	
+ 9	
- 9	
+ 1	

Note

In graph theory and computer science, the lowest common ancestor (LCA) of two nodes v and w in a tree or directed acyclic graph (DAG) is the lowest (i.e. furthest from the root) node that has both v and w as descendants, where we define each node to be a descendant of itself.