

Problem O. Towns and Roads

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

There are n towns and $n - 1$ bidirectional roads. Towns are numbered from 1 to n , and roads are numbered from 1 to $n - 1$, respectively. Road i connects town a_i and town b_i and has length d_i .

At the beginning, all roads are open, and from any town, you can reach all other towns using one or more roads.

There is a robot, initially in town 1.

Your task is to process q queries. Each query has one of the following three types:

- 1 x : move robot to town x . At the time of this query, it is guaranteed that the town where robot is located and town x are directly connected by one open road.
- 2 y : road y is closed. At the time of this query, it is guaranteed that road y is open.
- 3 z : road z is opened again. At the time of this query, it is guaranteed that road z is closed.

In addition, immediately after each query, print the list of towns that are farthest from the town where the robot currently is, if we consider only roads that are open after this query.

Input

The first line of the input contains one integer n , the number of towns ($1 \leq n \leq 2 \cdot 10^5$).

The i -th of the following $n - 1$ lines contains three integers a_i , b_i and d_i : the numbers of cities connected by i -th road and the length of this road, respectively ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$, $1 \leq d_i \leq 10^6$). It is guaranteed that, from any town, you can reach all other towns using one or more roads.

The next line contains one integer q ($1 \leq q \leq 2 \cdot 10^5$).

Then q queries follow. Each query is given on a separate line and has one of the three forms described above.

Output

Print q lines: the i -th line should contain the answer to the i -th query.

For each query, start the line with an integer c_i : the number of towns that are farthest from the town where the robot is after the i -th query. Then print c_i integers on the same line: the numbers of these towns in ascending order.

It is guaranteed that, in each test given to your solution, the sum of all c_i in the correct answer will not exceed $4 \cdot 10^5$.

Examples

standard input	standard output
6 2 4 1 1 2 1 4 6 1 2 3 1 4 5 1 5 2 5 2 3 1 2 3 5 1 4	1 6 2 3 4 3 1 3 4 1 5 2 1 3
5 3 4 1 2 1 1 4 5 1 3 2 1 6 2 2 3 2 1 2 1 3 2 4 1 4	1 1 1 5 1 5 2 1 5 1 5 2 3 5