Cancer IQ Coding Challenge

This is the Cancer IQ coding challenge for the Software Engineering Intern role.

Some aspects of CancerIQ's software deals with the genetics of cancer in families. The family tree is similar in some ways to computer science trees (and also quite different in some notable ways). Therefore our coding challenge today deals with trees and paths through trees.

Problem Description

———————————————————————————————————

Given a rooted tree of ***N*** nodes, where each node is uniquely numbered in between ***[1..N]***. The node 1 is the root of the tree. Each node has an integer value which is initially 0.

You need to perform the following two kinds of queries on the tree:

- ***add t value***: Add ***value*** to all nodes in subtree rooted at ***t***
- ***max a b***: Report maximum value on the path from ***a*** to ***b***

Input Format

First line contains ***N***, number of nodes in the tree. Next ***N-1*** lines contain two space separated integers ***x*** and ***y*** which denote that there is an edge between node ***x*** and node ***y***.
Next line contains ***Q***, the number of queries to process.
Next ***Q*** lines follow with either ***add*** or ***max*** query per line.

Constraints

$1 \le N \le 100{,}000$
$1 \le Q \le 100{,}000$
$1 \le t,a,b,x,y \le N$
$x \ne y$
$-10{,}000 \le value \le 10{,}000$

Output Format

For each *max* query output the answer in a separate line.

Sample Input

```
5
1 2
2 3
2 4
5 1
6
add 4 30
add 5 20
```

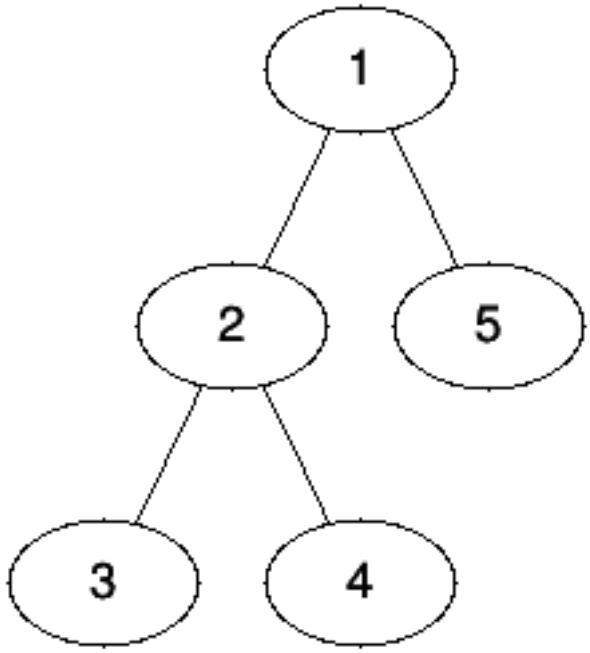
```
max 4 5
add 2 -20
max 4 5
max 3 4
```

Sample Output

```
30
20
10
```

Explanation

In the test case we have the following tree:



Initially all node values are zero.
Queries are performed in the following way:

add 4 30 // add 30 to node 4
add 5 20 // add 20 to node 5
max 4 5 // for the nodes 4,2,1,5 the maximum is **30** because node 4 is 30
add 2 -20 // subtract 20 from nodes 2,3,4
max 4 5 // for the nodes 4,2,1,5 the maximum is now **20** because node 5 is 20
max 3 4 // for the nodes maximum 3,2,4 the maximum is **10** because node 4 is 10

———————————————————————————————————

Instructions

You may use any coding language to solve this challenge.

Attached to this email are sample inputs (named input##.txt), sample outputs (named output##.txt), and two "challenge" inputs. The sample outputs correspond to the sample inputs. The purpose of the sample inputs and outputs are for you to test your code and verify that it's working correctly.

We want your two solutions to the two "challenge" inputs, named **input_easy.txt** and **input_hard.txt**.

IMPORTANT
Please attach your solutions as text files named **output_easy.txt**, and **output_hard.txt.** Please attach your **source code** also.

If you cannot do **output_hard.txt**, please still attach **output_easy.txt**. You will get partial credit.

If you cannot do **output_easy.txt**, please still submit your source code, you will get some partial credit based on how well written and well designed and readable your source code is. Software engineering is about simplicity, readability, modularity, not just giant 2000-line spaghetti code scripts that work but are unmaintainable.