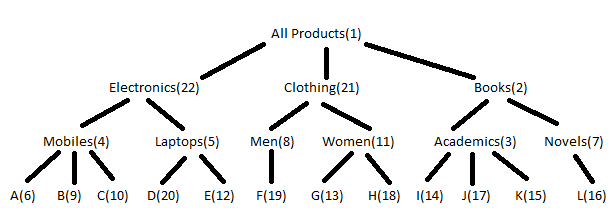
Deal-snapped is a super awesome e-commerce giant. The tech-geeks at Deal-snapped have maintained their databases in the form of a tree, with the hierarchy of items, their categories, sub-categories of the company such that it forms a tree like structure. Each one of them is having a unique key number from 1 to 'n'(count of all of them). But the key numbers are unordered. (Shown below)



With a new sale-season coming up every now-and-then, they have a requirement to simplify the task of easily applying a discount 'd' on any category,sub-category or product.

You also need to tell the discount on a particular product when queried, this will help process user requests about discount.

Input: You are given 'n' - the products,category and sub-categories key value limit. This is followed by n-1 lines containing (u,v) which represents a link between key(u) and key(v), representing the tree structure. Next line contains 'q' - number of queries. Next 'q' lines contains queries of the form:

* U x y - Update products under key(x) with 'y'% discount.
* Q x - Output the discount on product x.

Output: For each query of type - "Q x", output the discount on product x at that particular instant.

Note:

* "All products" always has a key value 1 and all categories, sub-categories and products lie under it.
* Products will be the leaf nodes in the tree.
* Query for discount will always be on products.
* Initially all products have 0% discount.

Constraints:

1<=n,q<=100000

1<=u,v<=n

0<=y<=60

'x' with Q will only be the key of products.

'x' with U can be any key from [1,n].

**Sample Input/Output**

|  |  |
| --- | --- |
| **Input** | **Output** |
| input1.txt | output1.txt |
| input2.txt | output2.txt |
| input3.txt | output3.txt |

**The expected time of completion for this question is 45 minutes**