

Snail family problems

In this problem we'll be looking at a group of snails. Two snails, numbered **a** and **b** can get married, and some time after that they may even get divorced. Any snail may be married to number of other snails.

The problem arises because snails don't like being married to their ancestors, and they don't know who their ancestors are - they only know the set of snails they're directly related to. A marriage between two snails from which one is an ancestor of the other is called "bad".

If they knew who the oldest snail (the ancestor of everyone else) is, everyone would know their ancestors.

For the given group of **N** snails numbered from 1 to **N**, you'll be given **N-1** pairs of snails (**a**, **b**), indicating that snails a and b are directly related.

Next you'll be given **Q** queries of the form:

Q n - suppose **n** is the oldest snail, how many "bad" marriages are there in the group?

M a b - snails **a** and **b** just got married.

D a b - snails **a** and **b** just got divorced.

No snail can ever get married to himself.

You may assume that no pair will get married twice (if they are already married). You may also assume that no pair will get divorced if they don't get married before that.

Input

The first line of input contains an integer **N** ($2 \leq N \leq 200000$).

Each of the next **N-1** lines contains a pair of integers **a** and **b** ($1 \leq a < b \leq N$).

The next line contains an integer **Q** ($1 \leq Q \leq 300000$).

Each of the next **Q** lines contains a query as defined above.

Output

For each query of type **Q** output a single line containing the number of "bad" marriages if snail numbered **n** was the oldest one.

Example

Input:

```
5
1 2
1 3
2 4
2 5
9
Q 1
M 3 5
```

Q 1
Q 3
M 1 4
Q 3
Q 4
D 3 5
Q 3

Output:

0
0
1
2
1
1