Lazy White Falcon

White Falcon just solved the data structure problem below using heavy-light decomposition. Can you help her find a new solution that doesn't require implementing any fancy techniques?

There are 2 types of query operations that can be performed on a tree:

1. `1 u x`: Assign x as the value of node u.
2. `2 u v`: Print the sum of the node values in the unique path from node u to node v.

Given a tree with N nodes where each node's value is initially 0, execute Q queries.

Input Format

The first line contains 2 space-separated integers, N and Q, respectively.
The $N-1$ subsequent lines each contain 2 space-separated integers describing an undirected edge in the tree.
Each of the Q subsequent lines contains a query you must execute.

Constraints

- $1 \leq N, Q \leq 10^5$
- $1 \leq x \leq 1000$
- It is guaranteed that the input describes a connected tree with N nodes.
- Nodes are enumerated with 0-based indexing.

Output Format

For each type-2 query, print its integer result on a new line.

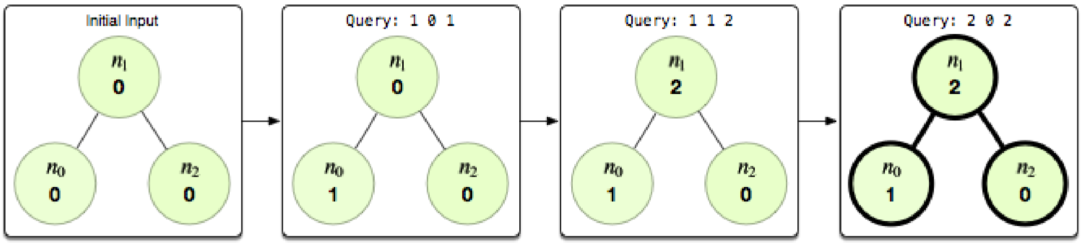
Sample Input

```
3 3
0 1
1 2
1 0 1
1 1 2
2 0 2
```

Sample Output

```
3
```

Explanation



After the first 2 queries, the value of node $n_0 = 1$ and the value of node $n_1 = 2$. The third query requires us to print the sum of the node values in the path from nodes 0 to 2, which is $1 + 2 + 0 = 3$. Thus, we print 3 on a new line.