Da Bian Tree Problem

Description

Do you know YP? YP is a talented musician and programmer.

He love data structures very much, and he always produce some interesting problem for you, such as `"Da Bian Tree Problem"`.

In this problem, YP will give you a tree with N nodes at first, and then there are M operations. Each operation is one of the following three types:

- `1 u` : add node u as `"target point"`
- `2 u` : remove node u from `"target points"`
- `3 x k` : find the k-th distance from node x to all the `"target points"`.

Something you must notice: in this tree, each edge has a weight, and nodes are numbered from 1 to N.

Input

The first line of the input contains two integers N, M. N is the number of nodes, and M is the number of operations.

The following $N-1$ lines describe each edge in the tree. Each line contains three integers u_i, v_i, w_i, it means there is an edge connects u_i and v_i with weight w_i.

The following M lines describe each operation. Each operation is one of the following three types:

- `1 x` : add node u as `"target point"`
- `2 x` : remove node u from `"target points"`
- `3 x k` : find the k-th distance from node x to all the `"target points"`. It guarantees at least one `"target points"` when doing this operation.

The constraints show as follows:

- $1 \leq N, Q \leq 10^5$
- $1 \leq u_i, v_i \leq N$
- $0 \leq w_i \leq 10^9$
- $1 \leq x \leq N$
- $1 \leq k \leq P$, where P is the number of current `"target points"`.

Output

For each `3 x k` operation, output the answer in a line.

Sample Input

```
5 10
3 2 3
1 3 4
5 4 2
4 1 3
1 3
1 4
3 5 2
3 5 1
2 3
1 1
1 2
3 3 3
3 2 2
3 1 1
```

Sample Output

```
9
2
7
7
0
```