Problem E. Excellent HLD

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	256 mebibytes

You are given a rooted tree consisting of n vertices. Its root is vertex 1. Let us consider a *heavy-light decomposition* of the tree, where each edge is either a *heavy edge* or a *light edge*. For each vertex, among all edges connecting the vertex with its children, at most one edge can be a heavy edge.

In this problem, we have a multiset of simple paths T, which is initially empty. We will assign each edge to be a heavy edge or a light edge according to T, satisfying the condition above.

Each time an update is done on T, your task is to find an assignment of edges that minimizes the sum of the number of light edges on every path in T.

There are q updates given. Each update consists of three integers: s, e, and k. They mean that k copies of the simple path from s to e are inserted into T. After each update, find the minimum sum of the number of light edges on every path in T.

Input

The first line of input contains two space-separated integers n and q ($2 \le n \le 10^5$; $1 \le q \le 10^5$).

The i-th of the following $n - 1$ lines contains two space-separated integers x_i and y_i, meaning that the i-th edge connects vertices x_i and y_i in the tree ($1 \le x_i, y_i \le n$; $x_i \ne y_i$). It is guaranteed that the given edges form a tree.

The i-th of the following q lines contains three space-separated integers, s, e, and k, describing each update ($1 \le s, e \le n$; $s \ne e$; $1 \le k \le 10^9$).

The updates are processed in the input order. The updates are permanent: the changes made in each update persist in further updates as well.

Output

For each of the q updates, print a line with the answer after this update.

Examples

standard input	*standard output*
3 3 1 2 3 1 1 3 2 1 3 3 1 2 2	0 0 2
5 5 3 4 2 4 1 2 5 3 5 4 2 1 2 4 3 4 1 5 3 4 1 2 2	0 0 0 0 0
8 8 4 6 8 4 1 6 5 1 2 1 3 2 7 3 2 7 1 8 2 1 5 3 6 8 3 5 1 4 2 6 7 1 5 6 4 6 2 3	0 1 7 12 14 15 23 26