Bar hopping

Input file:	*standard input*
Output file:	*standard output*
Time limit:	6 second
Memory limit:	1024 MB

After even more Covid-19 related restrictions have been lifted, Lucia can go back to one of her favourite hobbies - bar hopping. The city where she wants to do that has all interesting locations placed in a way such that the roads between them form a tree. Moreover, each road i has a wealth index w_i. Now, pub owners are quite the elitists, so in order to be able to go from a bar a to a bar b through road i, it must hold that she has a wealth index greater or equal than w_i.

Lucia is no stranger to bar hopping. In fact, there are combinations of pubs that she has tried before the pandemic. Therefore, she wants to make things more interesting.

For a given path from X to Y, Lucia is now wondering how much wealth she would need in order to visit K other pubs that are not included in said path, starting from X. Since she is quite adventurous, she does not ask herself this only once - she does this Q times.

Input

The first line of input will contain two integers, N ($1 \leq N \leq 10^5$) and Q ($1 \leq Q \leq 10^5$). The following $N-1$ lines will contain three integers x, y and w ($1 \leq w \leq 10^9$), signifying that there is a road from x to y with a wealth index of w. The next Q lines of input shall also contain three integers x, y, k ($k \geq 1$), representing a query: given a path from x to y (not necessarily distinct), what is the minimum wealth index that Lucia requires in order to go to k other pubs which do not belong to this path? It is guaranteed that there are at least k nodes outside the path.

Output

The output shall have a line containing Q different integers $wealth_i$. Integer $wealth_i$ represents the answer to query i.

Example

standard input	standard output
8 3 2 1 7 3 1 4 4 3 8 5 1 10 6 4 2 7 4 5 8 2 6 4 5 4 3 7 4 2 3 3	8 8 8