OCC '19 G6 - Monkeys in a Tree

Time limit: 2.75s **Memory limit:** 512M

In DMOJ forest, there is a tree with N nodes, conveniently numbered from 1 to N, connected by $N-1$ weighted edges. The i^{th} edge connects nodes u_i and v_i with distance w_i. On each node, a monkey lives there. The monkey at node i has a ranking r_i. Multiple monkeys may have the same ranking.

Monkey king plans to host a banana eating competition. He picks up a node X as the location and invites monkeys whose rankings are within $[L, R]$ (i.e. $L \leq r_i \leq R$) to attend this event. Monkey king wants to figure out the sum of distances for all invited monkeys travelling to X. Since monkey king hasn't found out the best location, he will ask you Q queries. In the i^{th} query, he will give you X_i, L_i and R_i. You need to tell him the sum of distances for monkeys with rankings in $[L_i, R_i]$ travelling to X_i. Because monkey king is quite impatient, you must answer his queries online.

Constraints

For all subtasks:

$1 \leq N \leq 150\,000$

$1 \leq Q \leq 200\,000$

$1 \leq M \leq 10^9$

Subtask	Points	Additional constraints
1	19	$N \leq 3000$, $Q \leq 300$, $M \leq 10^9$
2	21	$N \leq 10^5$, $Q \leq 10^5$, $M \leq 20$
3	23	$N \leq 10^5$, $Q \leq 10^5$, $M \leq 10^9$
4	37	No additional constraints.

Input Specification

The first line contains three integers, N, Q, and M ($1 \leq N \leq 150\,000$, $1 \leq Q \leq 200\,000$, $1 \leq M \leq 10^9$), where N is the number of nodes in the tree, Q is the number of queries, and M is used to decode each query.

The second line contains N integers, r_i, ($0 \leq r_i < M$), the ranking of the i^{th} monkey.

Each of the following $N-1$ lines contains three integers, u_i, v_i and w_i, ($1 \leq u_i, v_i \leq N$, $1 \leq w_i \leq 1\,000$), an edge between nodes u_i and v_i with length w_i.

Q lines of input follow. The i^{th} line contains three integers, X_i, l_i and r_i, where l_i and r_i are the encoded ranking interval. You can get the actual L_i and R_i by calculating $L_i = \min((l_i + \text{lastans}) \,\%\, M, (r_i + \text{lastans}) \,\%\, M)$ and $R_i = \max((l_i + \text{lastans}) \,\%\, M, (r_i + \text{lastans}) \,\%\, M)$, where lastans is the answer for the previous query, and specially $\text{lastans} = 0$ for the first query.

Output Specification

Print Q lines. The i^{th} line contains an integer, the sum of distances for the i^{th} query.

Sample Input

```
10 5 10
0 0 7 2 1 4 7 7 7 9
1 2 5
2 3 3
1 4 4
2 5 6
4 6 5
3 7 2
1 8 7
6 9 3
7 10 4
8 9 8
2 8 0
9 3 1
8 0 8
4 2 7
```

Sample Output

```
21
75
61
117
68
```