
Problem A. Highway Buses

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 512 megabytes

There are n bus stations in Harbin, connected by m bidirectional highways. It is always possible to reach any bus station using these highways wherever you are. These bus stations are labeled by $1, 2, \dots, n$.

If you want to take a bus from the i -th station to the j -th ($1 \leq i, j \leq n$) station, you should first buy such a ticket for your trip and then enter the corresponding bus. The driver will always choose the shortest route to your destination, which is the route consisting of the minimum number of highways. But if your destination is too far away, you will not be able to buy such a ticket. Specifically, you can buy a ticket from the i -th station to the j -th station if and only if you are at the i -th station and $dis(i, j) \leq f_i$, where $dis(i, j)$ denotes the number of highways on the shortest route between i and j .

Alice is at the first station, and she wishes to visit her best friend Bob, who is at the k -th ($1 \leq k \leq n$) station. She needs to take several buses to reach the k -th station **in one day**. Initially, buying a ticket at the i -th station will cost Alice c_i dollars. Buying a ticket on the second day will cost her $c_i + w_i$ dollars. And if she wants to buy a ticket on the third day, she'll need to pay $c_i + 2w_i$ dollars, and so on. In short, the cost changes by w_i every day.

To save money, Alice needs to select the best date T , where $1 \leq T \leq T_{\max}$, and take several buses all on that day to reach the k -th station, such that the total cost for the tickets is minimized. Note that initially $T = 1$, so she needs to pay $c_i + (T - 1)w_i$ dollars on the T -th day.

Bob hasn't told Alice where he is yet. Please write a program to help Alice find the cheapest way for all possible destinations $k = 1, 2, \dots, n$.

Input

The input contains only a single case.

The first line of the input contains three integers n, m and T_{\max} ($1 \leq n \leq 200\,000$, $n - 1 \leq m \leq n + 50$, $1 \leq T_{\max} \leq 10^6$), denoting the number of bus stations, the number of highways and the parameter T_{\max} .

Each of the next n lines contains three integers f_i, c_i and w_i ($1 \leq i \leq n$, $1 \leq f_i \leq n$, $1 \leq c_i \leq 10^9$, $|w_i| \leq 10^9$), denoting the parameters of the i -th bus station. It is guaranteed that the cost for each ticket will never be negative and will never exceed $2 \cdot 10^9$.

Each of the next m lines contains two integers u_i and v_i ($1 \leq i \leq m$, $1 \leq u_i, v_i \leq n$, $u_i \neq v_i$), denoting a bidirectional highway connecting the u_i -th station and the v_i -th station. It is guaranteed that it is always possible to reach any bus station using these highways wherever you are. Note that there may be multiple highways between the same pair of bus stations.

Output

Print n lines, where the k -th ($1 \leq k \leq n$) line contains a single integer, the minimum number of dollars that Alice needs to pay if Bob is at the k -th station.

Example

standard input	standard output
6 6 2	0
1 50 -40	10
1 2 100	52
2 1 100	52
2 4 100	52
3 1 100	10
1 1 100	
1 2	
2 3	
3 4	
4 2	
2 5	
6 1	

Note

In the sample test:

- When $k = 1$, the answer is obviously 0.
- When $k = 2$, the best date T is 2.
- When $k = 3$, the best date T is 1.
- When $k = 4$, the best date T is 1.
- When $k = 5$, the best date T is 1.
- When $k = 6$, the best date T is 2.