Competitive Programming SS24

Submit until end of contest

Problem: traderoute (2.0 second timelimit)

Note: This is a problem that is harder to solve than usual. Solve the other problems first before spending too much time on this one.

Jonathan is the CEO of the greatest trading corporation in the Outer Rim. Just recently, he bought the rights to expand trade to a south western system. There are n planets each with their own space station and m directional routes between them, the i-th route costing c_i credits. To conform to the Act of Free Trade Quality, the i-th space station satisfies quality level q_i.

The i-th of k goods are to be traded from space station a_i to b_i. To be able to sell them for a profit, they require a quality r_i. This quality is met if all intermediate space stations have at least quality level r_i. (For some reason, space station a_i and b_i do not require quality r_i.)

Since Jonathan is busy evading taxes, you have to find the minimum cost to transport each good to their destination or determine that they are not worth it.

Input The first line contains n, m and k ($2 \leq n \leq 500,\ 0 \leq m \leq n \cdot (n-1)$, $1 \leq k \leq 10^5$) the number of planets, routes and goods, respectively.

The second line contains n integers $q_1, q_2, \ldots, q_n$ ($0 \leq q_i \leq 10^9$), meaning that the i-th planets space station has quality level q_i.

The next m lines contain three integers u_i, v_i and c_i ($1 \leq u_i, v_i \leq n,\ u_i \neq v_i$, $0 \leq c_i \leq 10^9$) representing a route from u_i to v_i costing c_i. No two routes have the same start and end point.

The next k lines contain three integers a_i, b_i and r_i ($1 \leq a_i, b_i \leq n, 0 \leq r_i \leq 10^9$) representing a good that needs to be shipped from a_i to b_i requiring quality level r_i.

Output Print k integers on k lines, the minimum cost to ship the i-th good, while meeting the requirement, on the i-th line. If a good is not profitable, print -1 instead.

Sample input

```
3 4 5
3 2 3
1 2 1
3 2 5
1 3 4
2 3 2
1 3 2
1 3 3
2 1 0
3 2 4
2 2 100
```

Sample output

```
3
4
-1
5
0
```