Puzzle Competition

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

The annual CCPC (*Cipher & Code Penetrating Competition*) is approaching, and this year's competition has a unique way of releasing the problems.

This competition has a total of n problems, numbered from 1 to n. However, unlike previous years, this year's problems are not all unlocked at the beginning of the competition; instead, they are unlocked gradually. Specifically, the organizing committee has determined a directed graph with n nodes based on the relationships between the problems, where the nodes are numbered from 1 to n and node i represents the i-th problem. Initially, each problem has an energy level of 0, and each problem has a parameter a_i, which indicates that if the energy level of this problem is greater than or equal to a_i, the problem will be immediately unlocked. Once unlocked, all directed edges originating from that node will **simultaneously** transfer 1 point of energy to the corresponding destination problem, which takes w_i seconds.

However, the organizing committee has discovered that some problems may never be unlocked, which is something they want to avoid. Therefore, they designed k forced refreshers to help participants progress. Each forced refresher controls some problems and will be activated at time t_i seconds, immediately modifying the parameter a_i of the problems it controls to 0.

Now, for each problem, you want to know its earliest unlock time or determine if it can never be unlocked.

Input

The first line contains three integers n, m, k ($3 \le n \le 10^5, 0 \le m \le 10^6, 0 \le k \le 10^5$), representing the number of problems, the number of directed edges, and the number of forced refreshers.

The second line contains n integers a_i ($0 \le a_i \le 10^5$), representing the parameters of each problem. It is guaranteed that there is at least one i such that $a_i = 0$.

The next k lines contain the j-th line with two integers t_j, sc_j ($0 \le t_j \le 10^9, 1 \le sc_j \le n$), indicating that the j-th forced refresher is activated at t_j seconds and controls sc_j problems, followed by sc_j integers $id_1, \ldots, id_{sc_j}$ ($1 \le id_l \le n$), representing the problem numbers controlled by this forced refresher. It is guaranteed that the problems controlled by a forced refresher are all distinct.

The next m lines each contain three integers u, v, w ($1 \le u, v \le n, 0 \le w \le 10^9, u \ne v$), indicating that there is a directed edge from problem u to problem v, which takes w seconds to transfer 1 point of energy.

It is guaranteed that $\sum sc_i \le 10^6$.

Output

Output a single line containing n integers, representing the shortest time required to unlock each problem. If a problem can never be unlocked, output -1.

Examples

standard input	standard output
6 9 0 0 2 1 1 1 4 1 2 1 2 3 1 3 4 1 4 5 1 5 2 1 2 6 1 3 6 1 4 6 1 5 6 1	0 -1 -1 -1 -1 -1
6 9 1 0 2 1 1 1 4 100 2 3 5 1 2 1 2 3 1 3 4 1 4 5 1 5 2 1 2 6 1 3 6 1 4 6 1 5 6 1	0 101 100 101 100 102
4 3 0 1 0 1 1 3 1 10 1 2 100 2 4 1000	-1 0 -1 1000

Note

Note: The input size for this problem is very large; please use a faster input method for reading.