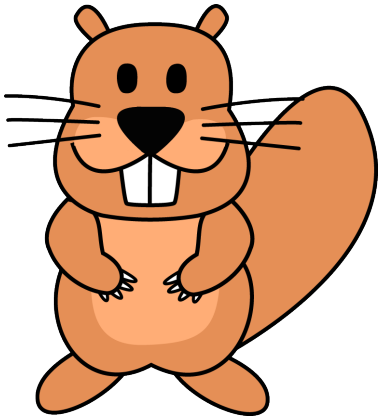
Bitaro's Party



There are N towns of beavers numbered from 1 to N in the descending order of their heights. No two towns have the same height. There are M canals connecting two different towns unidirectionally. The i-th canal ($1 \leq i \leq M$) flows from town S_i to town E_i. These canals flow from high towns to low towns. You cannot move against flow of the canals.

Bitaro, a beaver, has N friends, one of whom lives in each of N towns.

Bitaro is going to have parties Q times, inviting his friends. It is known for the j-th ($1 \leq j \leq Q$) party that Y_j friends are too busy to attend it. The j-th party is held in town T_j, so his friends who cannot go from their towns to town T_j only via canals cannot attend it either. Other friends come to the party.

Each friend come to the town where the party is held via canals. There may be several paths they can take. But Bitaro's friends love canals, so they must choice one of the paths which have the largest number of canals.

Bitaro wonders how many canals the attendant who uses the largest number of canals uses.

Task

Given the indexes of the towns where the parties are held and lists of busy friends for the Q parties, write a program which calculates how many canals the attendant who uses the largest number of canals uses.

Input

Read the following data from the standard input.

- The first line of input contains three space separeted integers N, M, Q. These mean there are N towns of beavers and M canals and Q parties Bitaro has.
- The i-th line ($1 \leq i \leq M$) of following M lines contains two space separated integers S_i and E_i. These mean the canal flows from S_i to E_i unidirectionally.

- The j-th line ($1 \leq j \leq Q$) of following Q lines contains two space separated integers T_j, Y_j and Y_j space separeted integers $C_{j,1}, C_{j,2}, ..., C_{j,Y_j}$. These mean the j-th party is held in town T_j and friends living in town $C_{j,1}, C_{j,2}, ..., C_{j,Y_j}$ are busy.

Output

Output contains Q lines. The j-th line ($1 \leq j \leq Q$) contains the number of canals the attendant who uses the largest number of canals uses for the j-th party. If no one can attend the j-th party, the j-th line contains -1.

Constraints

All input data satisfy the following conditions.

- $1 \leq N \leq 100\,000$.
- $0 \leq M \leq 200\,000$.
- $1 \leq Q \leq 100\,000$.
- $1 \leq S_i < E_i \leq N$ $(1 \leq i \leq M)$.
- $(S_i, E_i) \neq (S_j, E_j)$ $(1 \leq i < j \leq M)$.
- $1 \leq T_j \leq N$ $(1 \leq j \leq Q)$.
- $0 \leq Y_j \leq N$ $(1 \leq j \leq Q)$.
- $1 \leq C_{j,1} < C_{j,2} < \cdots < C_{j,Y_j} \leq N$ $(1 \leq j \leq Q)$.
- $Y_1 + Y_2 + \cdots + Y_Q \leq 100\,000$.

Subtasks

There are 3 subtasks. The score and additional constraints of each subtask are as follows:

Subtask 1 [7 points]

- $N \leq 1\,000$.
- $M \leq 2\,000$.
- $Q = 1$.

Subtask 2 [7 points]

- $Q = 1$.

Subtask 3 [86 points]

There are no additional constraints.

Sample Input and Output

Sample Input 1	Sample Output 1
5 6 3 1 2 2 4 3 4 1 3 3 5 4 5 4 1 1 5 2 2 3 2 3 1 4 5	1 3 0

Among the friends attending the first party (the friends living in town 2, 3 or 4), the friends living in town 2 or 3 go to town 4, where the party is held, via the largest number of canals. This number is 1, so output 1.

Among the friends attending the second party (the friends living in town 1, 4, or 5), the friend living in town 1 goes to town 5, where the party is held, via the largest number of canals. This number is 3, so output 3.

The friend living in town 2 is the only one who attends the third party. He uses no canals, so output 0.

Sample Input 2	Sample Output 2
12 17 10 1 2 2 3 3 4 1 5 2 6 3 7 4 8 5 6 6 7 7 8 5 9 6 10 7 11 8 12 9 10 10 11 11 12 6 3 1 7 12 3 7 1 2 3 4 5 6 7 11 3 1 3 5 9 2 1 9 8 4 1 2 3 4 1 1 1 12 0 10 3 1 6 10 11 8 2 3 5 6 7 9 10 11 8 7 2 3 4 5 6 7 8	1 -1 3 1 3 -1 5 2 4 4