You are working as the product manager of *Cafebazaar*, a well-known Iranian Android marketplace. As your one-year plan, you have a list of applications to develop for which you need to assign proper developers. There are n developers in Cafebazaar, some of which are full-time, and the others are part-time developers. Your have m applications in your list, some of which are business-critical and the others are ordinary applications.

Of course, developers have different skills. Some developers are professional in developing some applications, while they are incapable of developing the other applications. If developer i is capable of developing application j, then you will gain $x_{i,j}$ amount of payoff by assigning her to that application in your plan. Obviously, a developer cannot be assigned to an application that she is not capable to develop. To gain maximum time and cost efficiency, you would like to assign each developer to at most one application, and assign at most one developer to each application. A development plan is *proper* if it has the following two properties:

(i) all full-time developers are involved in the plan, so they will not get disappointed, and

(ii) all critical applications are covered, hence you will not lose clients.

In other words, a development plan is proper if each full-time developer is assigned to exactly one application, and exactly one developer is assigned to each critical application. Note that in a proper plan, you may also assign part-time developers to applications, or assign developers to ordinary applications. Your goal is to find a proper plan that yields a maximum amount of payoff.

Input

There are multiple test cases in the input. Each test case starts with a line containing two space-separated integers n and m ($1 \le n, m \le 100$), representing the number of developers and applications, respectively. The next line starts with an integer t ($0 \le t \le n$), the number of full-time developers, followed by t numbers, each of which is the index of a full-time developer. Developers are indexed from 1 to n. The next line starts with an integer s ($0 \le s \le m$), the number of critical applications, followed by s numbers, each of which is the index of a critical application.

Applications are indexed from 1 to m. The next n lines, one for each developer, contain the following information. The i-th line ($1 \le i \le n$) starts with an integer d_i ($0 \le d_i \le m$), the number of applications that developer i is capable to develop, followed by d_i pairs of integers $a_{i,j}$ and $x_{i,j}$ ($1 \le j \le di$, $1 \le a_{i,j} \le m$, $1 \le x_{i,j} \le 10^6$), where $a_{i,j}$ is the index of an application that developer i is capable to develop, and $x_{i,j}$ is the payoff achieved if application $a_{i,j}$ is developed by developer i. Note that each $a_{i,j}$ as the index of an applicaton appears at most once in this list of a developer. The input terminates with a line containing '0 0' which should not be processed.

Output

You should output one line for each test case, containing the maximum payoff you can obtain by a proper plan. If there is no proper plan, you must output '-1' in one line.

Sample Input

```
2 4
1 1
1 3
2 1 8 2 10
3 2 2 3 10 4 50
4 3
3 1 2 4
2 1 3
1 1 200
2 2 700 3 200
2 2 300 3 100
1 1 500
0 0
```

Sample Output

```
20
-1
```