

Competitive Programming SS24

Submit until end of contest



Problem: storerun (1.0 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.

Another lock down is on the horizon but you still need to get your weekly groceries. However hoarders are also running to the stores and are buying up all available stock.

The stores are connected by bidirectional tram lines. You know the city by heart, so you know exactly which stores are connected. Also you found a local website where the stock of groceries of all stores is updated in real time and a very accurate estimation, for when all stock of a given grocery is bought up, is given.

To avoid contamination you want to find the minimum number of tram rides between neighboring stores, which allow you to get all your groceries and return back home.

Input In the first line you get three numbers n, m and k ($1 \leq n \leq 10^3$, $0 \leq m \leq 10^3$, $1 \leq k \leq 10$) the number of all stores, the number of all tram lines and the number of different groceries you need. You live and start right next to the first store.

In the next line you get k space separated identifiers¹ for the groceries you need.

Then m lines follow each with two numbers a, b ($1 \leq a, b \leq n$, $a \neq b$) meaning that the a -th and b -th store are connected directly by a tram line.

Then n lines follow. The i -th line describes the stock of the i -th store. The line starts with a number s ($0 \leq s \leq 30$) the amount of different items the store is offering. This number is followed by s pairs of an identifier and a number t ($0 \leq t \leq 10^9$). Meaning that the store will have the specified item in stock as long as you did less than t tram rides.

Output If its possible to get all your groceries give the minimum number of tram rides you need to buy all of them and get to where you started. Otherwise print "-1" (without the quotes).

¹An identifier is a string of non-whitespace characters, with length at most 20.

Sample input

```
5 4 2
toiletpaper soap
1 2
2 3
3 4
2 5
0
1 soap 2
2 soap 1 toiletpaper 2
1 toiletpaper 5
1 toiletpaper 2
```

Sample output

```
6
```

```
3 3 4
a b c d
1 2
2 3
1 3
1 a 0
3 b 3 c 2 d 7
1 a 3
```

```
3
```

```
5 4 2
the-answer-to-life 42
1 2
2 3
3 4
2 5
2 the 0 universe 12
2 and 139 everything 95
0
1 42 5
1 the-answer-to-life 2
```

```
-1
```