11903   e-Friends

Social networking web sites are very popular these days. I am not mentioning any names because you know better than me. People do have a lot of e-friends now, "you may not know his real identity but he is your friend". Sometimes, there can be rivalries between people, and thus people gets blocked or deleted. And these rivalries don't have to be necessarily commutative. That means x may think y as his enemy, but y may or may not think so.

However, what if you get the opportunity to see all your e-friends live? Since you like social networks very much, you planned to arrange a get-together with your e-friends. So, you invited n of your e-friends in your home. For simplicity, you numbered them from 1 to n. They were your friends, but rivalries existed amongst many of them. You asked them to form a queue such that they can get food from the table. And you will serve foods one by one. But your friends demanded that, no enemy should be the next person in the queue. That means if x thinks that y is his enemy and x's position in the queue is i, then y's position shouldn't be $i - 1$.

So, it became very complex. That's why you decided to add a dissatisfaction index k if one's demand is not fulfilled. And the total dissatisfaction index is the summation of all the dissatisfaction indexes. Now you know the rivalries and the maximum total dissatisfaction index you may allow, you want to find the total number of possible arrangements.

Input

The first line of input will contain T (≤ 30) denoting the number of cases.

Each case starts with 3 integers n ($1 \leq n \leq 12$), k ($0 \leq k \leq 10^6$) and q ($1 \leq q \leq 1000$) which denotes the number of queries. Then there will be n lines. The i-th line contains an integer t_i ($0 \leq t_i < n$) and ids of t_i distinct friends. It means i-th person thinks the t_i listed persons as his enemy. Each of the next q lines contains an integer r ($0 \leq r \leq 10^8$) denoting the total allowable dissatisfaction index.

Output

For each case, print the case number first. Then for each query, print the total number of arrangements possible not exceeding the given total dissatisfaction index.

Sample Input

```
1
2 10 2
1 2
0
10
5
```

Sample Output

```
Case 1:
2
1
```