

You are working as an HR officer in “Nice Computer Programmer’s Company(NCPC)”. You have to hire a lot of people in a very short period. The hiring process is simple. Interview all the applicants, grade them, select the ones who survived the interview board, then assign them to projects. So, you can see, the hardest job is interviewing the candidates, and find who is capable of working in NCPC. Don’t worry, you don’t need to do that. You will be given a list of candidates, who survived the interview, sorted according to their scores given by the interviewers. You need to to assign them to different projects. Simple, isn’t it? But, unfortunately, this is not as easy as it seems to be.

There are N candidates and M different projects, each project requiring r_i programmers. You know that, not everyone is interested in working in any position. For example, not everyone likes the job of tester, or QA Engineer. Every applicant has given you a list of job, they can work, according to their preferences. Since, NCPC needs to hire a lot of people, they have decided to recruit maximum number of programmers, they can from current candidates.

There can be more than one way to select maximum number of candidates. That is where their grades and preferences comes into play. If there is more than one way to fill the maximum number of spot, give the highest ranked candidate his most preferred position you can. If there is still more than one way, give the second highest ranked candidate, his most preferred position possible, and so on.

Input

First line contains T ($T \leq 61$), the number of test cases. Each test case starts with two integer N ($N \leq 50$) and M ($M \leq 50$), the number of candidates, and the number of available projects. The following line contains M integers r_i ($r_i \leq 50$, $\Sigma r_i \leq 1500$), the number of open spots in project i . The next N lines describe the preferences of candidate i . Each of the lines starts with an integer K ($K \leq M$), followed by K integer p_j ($j = 1 \dots K$), the projects i -th applicant is interested to work, sorted by his/her preference (p_1 is the most preferred project, and p_K is the least preferred). Applicant 1, is the highest ranked applicant, and applicant N is the least ranked. for all candidates, $p_j \leq M$, and no job appear in the list twice

Output

For each test case output the case number, followed by the maximum number of applicant you can recruit, L . This is followed by L lines, each containing two integers, a and p , where a -th applicant is recruited for project p .

Sample Input

```
1
3 3
1 1 1
2 1 2
2 3 2
2 3 2
```

Sample Output

```
Case #1:
3 applicant(s) can be hired.
1 1
2 3
3 2
```