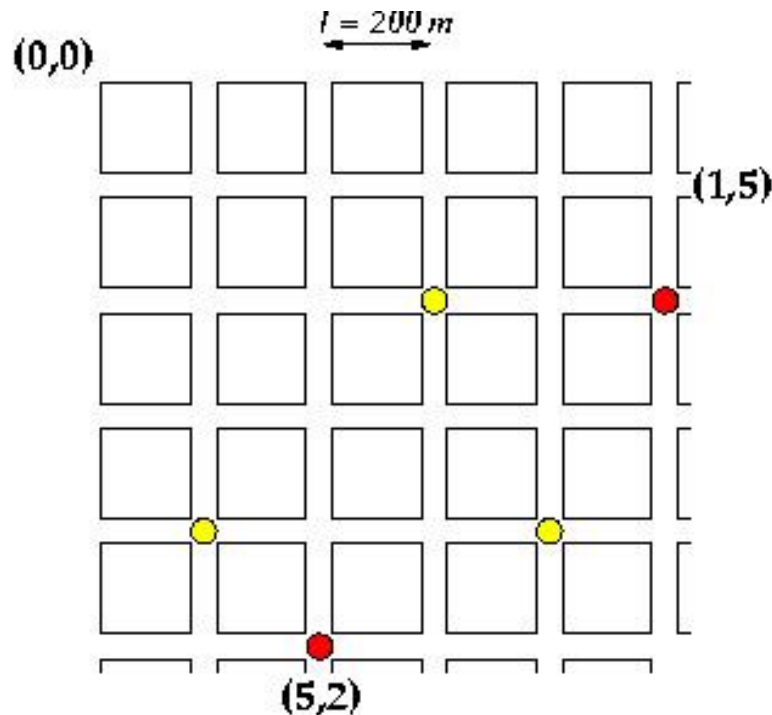


Taxi!

The Problem

Party town is a well planned town. Each town block is 200m x 200m and all of the streets are built as straight lines that intersect at right angles (see diagram).



Although the town is well designed and very pretty, the best thing about Party Town is that the inhabitants love to party! Every month they hold a “mystery venue” party. The venue for the party is only announced a short time before the party starts. The “mystery venue” parties tend to fill up very quickly, so everyone wants to get there as soon as possible. To give everyone a chance, the venue is announced on the radio just before the party starts, so everyone tries to get there as quickly as possible.

This is all great news for the Party Town Taxi Service. As soon as the party venue is announced on the radio, people all over town ring up for a taxi to take them to the party. The problem for the taxi service is that a lot of people ask for a ride at the same time and the taxis have to pick up the people quickly.

You have been asked to help the Party Town Taxi Service figure out how many passengers they can take to the party. You must take into account the following constraints:

- Each taxi can only take one passenger
- Passengers always wait at intersections of roads
- At the time of the radio announcement, all taxis are also waiting at intersections
- The taxi has to reach the passenger within a given time limit (or they will be too late to get into the party)

The Input

The first line contains the number of test cases K ($K \leq 250$). The first line of each test case contains the number of persons P ($1 \leq P \leq 400$), the number of taxis T ($1 \leq T \leq 200$) the speed S ($1 \leq S \leq 2000$) of the taxis in meters per second and the time limit C to collect a person in seconds ($1 \leq C \leq 1000000$). The next P lines contain the positions of the persons. The next T lines contain the position of the taxis in the city.

The Output

For each test case, output the test case number followed by a colon and a space and then the maximum number of persons who can get to the party.

Sample Input

```
1
2 3 10 40
2 5
5 2
2 3
4 1
4 4
```

Sample Output

```
1: 2
```