

Diamonds

Consider the following board computer game. The game area is an $N \times M$ rectangle divided into 1×1 squares. A player can be in one of these squares in a single turn, while any movement of the player can only take place between turns. During one "in-between" turn, only movement to an adjacent square (horizontally and vertically) is allowed, although a player may not move at all.

At the beginning of each turn a diamond appears on one of the squares of the board, and at the end of the turn it disappears. The diamond can only be taken away if the player is on the same square.

Your task is to write a program that, based on the position of successively appearing diamonds, determines how many of them can be taken away at most. The initial position of the player is arbitrary.

Input

The first line of the input contains a natural number d ($1 \leq d \leq 10$), specifying the number of data sets whose descriptions are placed sequentially on the following lines. The description of a single set is as follows:

The first line contains three natural numbers: the number of rows of the board N ($1 \leq N \leq 10$), the number of columns M ($1 \leq M \leq 10$) and the duration of the game in turns T ($1 \leq T \leq 1000$). The next T lines contain the coordinates of the diamonds that appeared in successive turns given by row n ($1 \leq n \leq N$) and column m ($1 \leq m \leq M$).

Output

Each set on the input should correspond to one line of output. This line should contain the maximum possible number of diamonds to collect.

Example

Input:

```
3
3 3 3
1 1
2 2
1 2
3 3 5
1 1
2 3
3 3
3 1
2 2
3 3 3
1 1
1 1
3 1
```

Output:

```
2
```

