Problem B
Secret Sharing

Input File: *testdata.in*
Time Limit: *5 secs.*

Problem Description

There are two communities, named X-community and Y-community. The X-community is formed by m members $\{1, 2, \ldots, m\}$, and the Y-community is formed by n members $\{1, 2, \ldots, n\}$. Any two members in the same community cannot share a secret. If two members share a secret, then they are from different communities. In particular, any member in one community can share his (her) secret with at least one member in the other community. For convenience, we use a *sharing-secret-relationship* (i, j) to represent that member i in X-community and member j in Y-community can share secret.

One day, a government administrator would like to investigate some event. He(She) needs to find a set I^* with the maximum cardinality such that any two members in I^* cannot share a secret. For example, consider $X = \{x_1, x_2, x_3\}$ and $Y = \{y_1, y_2, y_3\}$ and four sharing-secret-relationships $(x_1, y_1), (x_2, y_1), (x_3, y_1), (x_3, y_2)$, and (x_3, y_3). In this case, we can select members x_1 and x_2 from X and members y_2 and y_3 from Y to form I^*. Hence, $|I^*| = 4$ (Fig. 1). Given m and n and k sharing-secret-relationships, your task is to write a computer program to compute $|I^*|$.

Technical Specification

- $1 \leq m \leq 1000$.
- $1 \leq n \leq 1000$.
- $0 \leq k \leq \max\{m(n-1), n(m-1)\}$.

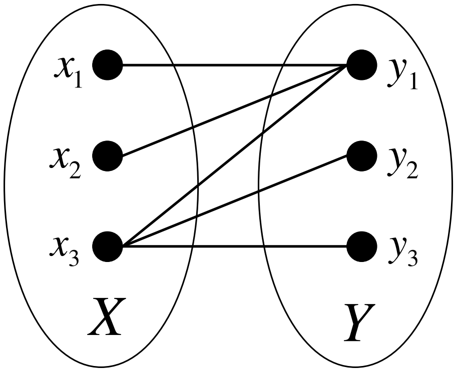

Figure 1: An example of $|I^*| = 4$.

Input File Format

The first line of the input file contains an integer, denoting the number of test cases to follow. For each test case, the the first line contains three positive integers m, n, and k. In the following k lines, each line contains two positive integers separated by one space, which represent one sharing-secret-relationship.

Output Format

For each test case, output $|I^*|$ in one line.

Sample Input

```
2
3 3 5
1 1
2 1
3 1
3 2
3 3
2 2 4
```

```
1 1
1 2
2 1
2 2
```

Output for the Sample Input

```
4
2
```