D. Baggage Claim

time limit per test: 2 seconds
memory limit per test: 256 megabytes

Every airport has a baggage claim area, and Balbesovo Airport is no exception. At some point, one of the administrators at Sheremetyevo came up with an unusual idea: to change the traditional shape of the baggage claim conveyor from a carousel to a more complex form.

Suppose that the baggage claim area is represented as a rectangular grid of size $n \times m$. The administration proposed that the path of the conveyor should pass through the cells $p_1, p_2, \ldots, p_{2k+1}$, where $p_i = (x_i, y_i)$.

For each cell p_i and the next cell p_{i+1} (where $1 \le i \le 2k$), these cells must share a common side. Additionally, the path must be simple, meaning that for no pair of indices $i \neq j$ should the cells p_i and p_j coincide.

Unfortunately, the route plan was accidentally spoiled by spilled coffee, and only the cells with odd indices of the path were preserved: $p_1, p_3, p_5, \ldots, p_{2k+1}$. Your task is to determine the number of ways to restore the original complete path $p_1, p_2, \ldots, p_{2k+1}$ given these $k+1$ cells.

Since the answer can be very large, output it modulo 10^9+7.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 3 \cdot 10^4$). The description of the test cases follows.

The first line of each test case contains three integers n, m, and k ($1 \le n, m \le 1000$, $n \cdot m \ge 3$, $1 \le k \le \left\lfloor \frac{1}{2}(nm-1) \right\rfloor$) — the dimensions of the grid and a parameter defining the length of the path.

Next, there are $k+1$ lines, the i-th of which contains two integers x_{2i-1} and y_{2i-1} ($1 \le x_{2i-1} \le n$, $1 \le y_{2i-1} \le m$) — the coordinates of the cell p_{2i-1} that lies on the path.

It is guaranteed that all pairs (x_{2i-1}, y_{2i-1}) are distinct.

It is guaranteed that the sum $n \cdot m$ over all test cases does not exceed 10^6.

Output

For each test case, output a single integer — the number of ways to restore the original complete path modulo 10^9+7.

Example

input Copy

```
5
2 4 2
1 1
2 2
2 4
1 4 1
1 1
1 4
5 5 11
2 5
3 4
4 5
5 4
4 3
5 2
4 1
3 2
2 1
1 2
2 3
1 4
3 4 4
1 2
2 1
```

Codeforces Round 1021 (Div. 2)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++23 14.2 (64 bit, ms

Choose file: Choose File No file chosen

Submit

→ Last submissions

Submission	Time	Verdict
317466305	Apr/27/2025 16:49	Accepted

→ Problem tags

combinatorics graphs math
No tag edit access

→ Contest materials

- Announcement
- Tutorial (ru)

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

output

```
2
0
2
5
1
```

Note

In the first test case, there are two possible paths:

- $(1,1) \to (2,1) \to (2,2) \to (2,3) \to (2,4)$
- $(1,1) \to (1,2) \to (2,2) \to (2,3) \to (2,4)$

In the second test case, there is no suitable path, as the cells $(1,1)$ and $(1,4)$ do not have a common neighboring cell.