Bipartite Graph Matching Problem

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

You are given a bipartite graph. The left part contains n_1 vertices numbered from 1 to n_1, and the right part contains n_2 vertices numbered from 1 to n_2. There are m edges, each connecting one vertex on the left to one vertex on the right.

For every interval $[\ell, r]$ on the right side (where $1 \le \ell \le r \le n_2$), consider the subgraph obtained by:

- keeping all left vertices $1, 2, \ldots, n_1$;
- keeping only the right vertices with numbers in $[\ell, r]$;
- keeping only edges whose endpoints are both kept.

In this subgraph, compute the size of a maximum matching (the maximum number of pairwise disjoint edges). Let this value be denoted by $f(\ell, r)$.

You need to compute the following value:

$$\sum_{1 \le \ell \le r \le n_2} f(\ell, r) \cdot \ell \cdot r \cdot ((\ell \oplus r) + 1),$$

where $\oplus$ denotes the bitwise XOR operation.

Output the value of this sum modulo 998 244 353.

Input

The first line contains a single integer t ($1 \le t \le 2000$), the number of test cases. For each test case:

The first line contains three integers: n_1, n_2, and m ($1 \le n_1, n_2 \le 5000$, $1 \le m \le 10^4$), representing the number of vertices on the left, the number of vertices on the right, and the number of edges.

Each of the next m lines contains two integers, u and v ($1 \le u \le n_1$, $1 \le v \le n_2$), which describe an edge between left vertex u and right vertex v.

There are no multiple edges. The sum of n_1 does not exceed 5000. The sum of n_2 also does not exceed 5000.

Output

For each test case, output a single line containing one integer: the required sum modulo 998 244 353.

Example

standard input	*standard output*
4 3 3 5 1 1 2 1 2 2 2 3 3 3 3 4 6 1 2 1 4 2 3 2 1 3 4 3 3 2 2 1 1 2 4 3 5 1 3 2 1 3 3 4 2 4 1	81 529 12 81