

4672 New Game

Alice and Bob are fond of a new interesting game. In this game, they play by turns in a Directed Acyclic Graph (DAG). Every node of the graph is assigned a non-negative integer at the beginning. Each round a player must choose a node having a positive integer and transfer 1 to one of its succeeding nodes, i.e. its integer decreases by 1 and certain succeeding node's integer increases by 1. Who cannot find such a node to transfer will lose, i.e. the integer is 0 in every node having at least 1 succeeding node.

Before playing the game, they come to an agreement about the integer assignment. Alice plays first. To be fair, Bob is responsible for assigning an integer to each node. But there is a constraint: the sum of all integers must be a predetermined value that they both agree.

Now, given the sum, Alice wonders how many distinct assignments which will prevent her from winning. Two assignments are different if there is one node assigned different integers in the two assignments.

Alice and Bob are so smart that they can always find the best strategy.

Input

The first line is a number indicating the number of test cases.

The first line of each test case includes three number N , M and S ($2 \leq N \leq 100$, $1 \leq M \leq 10000$, $1 \leq S \leq 10000$) indicating the number of nodes, the number of edges and the sum of all integers respectively. Nodes are numbered by $0 \dots N - 1$.

The following M lines each have two integers, u and v , indicating an edge from u to v ($0 \leq u, v < N$). You can assume there is no cycle in the graph.

Output

For each test case, output a single integer indicating how many distinct assignments in which Alice can NOT win. Because it can be very large, you should output the remainder divided by 1,000,000,007.

Sample Input

```
3
2 1 3
0 1
4 3 3
0 1
1 2
2 3
3 3 5
0 1
1 2
0 2
```

Sample Output

```
2
10
6
```