**Strongly Connected Components (Kosaraju's Algo):-**

Given a graph with **N** nodes and **M** directed edges. Your task is to complete the function **kosaraju()** which returns an integer denoting the number of [strongly connected components](https://en.wikipedia.org/wiki/Strongly_connected_component) in the graph.

**Input:**  
The first line of input contains an integer **T**. Then T test cases follow. Each test case contains two integers **N** and **M**. In the next line there are **M** space-separated values u,v denoting an edge from u to v.

**Output:**  
For each test case in a new line output will an integer denoting the no of strongly connected components present in the graph.

**Your Task:**  
You don't need to read input or print anything. Your task is to complete the function **kosaraju()** which takes the number of vertices V and adjacency list of the graph as inputs and returns an integer denoting the number of strongly connected components in the given graph.

**Expected Time Complexity:** O(N + M).  
**Expected Auxiliary Space:** O(N).

**Constraints:**  
1 <= T <= 100  
1 <= N <= 5000  
0 <= M <= (N\*(N-1))  
0 <= u, v <= N-1  
Sum of M over all testcases will not exceed 25\*106

**Example:  
Input**:  
2  
5 5  
1 0 0 2 2 1 0 3 3 4  
3 3  
0 1 1 2 2 0

**Output:**  
3  
1

**Explanation:  
Testcase 1:**  
There is a connected subgraph that includes 0-1-2 which satisfy the condition of strongly connecting components i.e each node is reachable from every other nodes.

Another subgraph includes individual nodes 4 and 3. That gives us a total of 3 subgraphs satisfying the condition of strongly connected components.