Digital Pakistan Speed Programing Competition Online Mock Contest

Instructions

- Do not open the booklet unless you are explicitly told to do so. You can only read these instructions below.
- If you have any question regarding the problems, seek a clarification from the judges using DOMJudge.
- Before submitting a run, make sure that it is executable via command line. For Java, it must be executable via "javac" and for GNU C++ via "g++". Java programmers need to remove any "package" statements and source code's file name must be the same as of main class. C++ programmers need to remove any getch() / system("pause") like statements.
- Do not attach input files while submitting a run, only submit/attach source code files, i.e., *.java or *.cpp or *.py.
- Language supported: C/C++, Java and Python3
- Source code file name should not contain white space or special characters.
- You must take input from Console i.e.: Standard Input Stream (stdin in C, cin in C++, System.in in Java, stdin in Python)
- You must print your output to Console i.e.: Standard Output Stream (stdout in C, cout in C++, System.out in Java)
- Please, don't create/open any file for input or output.
- Please strictly meet the output format requirements as described in problem statements, because your program will be auto judged by computer. Your output will be compared with judge's output byte-by-byte and not tolerate even a difference of single byte. So, be aware! **Pay special attention to spaces, commas, dots, newlines, decimal places, case sensitivity etc.**
- All your programs must meet the time constraint specified.
- The decision of judges will be absolutely final.

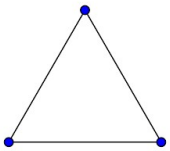
Problem 01: Circuit Design

Time limit: 18 seconds

Imagine you are an electronic engineer working on optimizing digital circuit layouts for integrated chips. Each circuit is represented as a graph where nodes correspond to components (like logic gates) and edges represent the connections between them.

Your task is to identify how often a specific subcircuit also called query subcircuit (a small group of components with a defined topology) appears in the overall circuit. Subcircuits can overlap, sharing components or connections, but each occurrence must have at least one unique component.

Given an undirected graph representing the circuit (components as nodes and connections as edges), identify and count all occurrences of the following pattern within the larger graph.



Input:

The input consists of the following:

- First line contains a pair ***N***, ***M***. The number of components in the circuit, ***N (1 ≤ N ≤ 2000)***. The number of edges in the circuit, ***M (0 ≤ M ≤ (N(N-1)/2))***.
- M lines, each containing two component IDs ***u*** and ***v (0 ≤ u, v ≤ N-1)***, representing an undirected edge from component ***u*** to component ***v*** in the circuit.

Output:

Output a single integer, the count of occurrences of the query sub circuit within the provided overall circuit.

Sample Input	**Sample Output**
6 9 0 1 1 2 1 3 2 3 3 4 4 0 2 5 5 0 5 1	3