**CSE221**

**Lab Practice Sheet 05**

**Summer 2024**

**A useful tool for making graphs:**

[**https://csacademy.com/app/graph\_editor/**](https://csacademy.com/app/graph_editor/)

**Task 01**

We all know about the prerequisite course. For example, before taking CSE221, we have to complete CSE220. And prior to taking CSE220, we need to complete CSE111 and CSE110. The course sequence is as follows:

CSE110 → CSE111 → CSE220 → CSE221

In this problem, there are **N** courses in the curriculum. There are **M** prerequisite requirements of the form "Course **A** has to be completed before course **B**".

Your task is to find an order in which you can complete the courses.

1. Solve it using the DFS approach.
2. Solve it using the BFS approach.

**Input:**

The first input line has two integers **N** (1 <= N <= 1000) and **M** (1 <= M <= N2) - the number of courses and prerequisite requirements. The courses are numbered 1,2,3,…,N.

Next, M lines describe the requirements. Each line has two integers **A**, **B** (1 <= A, B <= N)- course A has to be completed before course B.

**Output:**

Print an order in which you can complete the courses. Please note, that there could be multiple correct sequences. You can print any valid order that includes all the courses.

If there are no solutions, print "IMPOSSIBLE".

| Sample Input 1 | Sample Output 1 |
| --- | --- |
| 5 3  3 1  1 2  4 5 | 3 4 1 5 2 |
| Sample Input 2 | Sample Output 2 |
| 6 6  1 2  2 3  4 3  4 5  5 6  6 4 | IMPOSSIBLE |
| Sample Input 3 | Sample Output 3 |
| 8 10  1 2  1 4  2 4  2 5  2 3  4 6  4 5  6 5  5 3  7 8 | 1 7 2 8 4 6 5 3 |

**Task 02**

The problem statement of this problem is the same as **Task 01.**

The only difference is that in this task, you have to find the lexicographically smallest valid course sequence.

If you have two sequences, for example: A: 3 → 1 → 2 → 4 and B: 3 → 1 → 4 → 2. Then path A is lexicographically smaller than path B.

**Input:**

The first input line has two integers **N** (1 <= N <= 1000) and **M** (1 <= M <= N2) - the number of courses and prerequisite requirements. The courses are numbered 1,2,3,…,N.

Next, there are M lines describing the requirements. Each line has two integers **A**, **B** (1 <= A, B <= N)- course A has to be completed before course B.

**Output:**

Print the lexicographically smallest valid course sequence in which you can complete the courses.

If there are no solutions, print "IMPOSSIBLE".

| Sample Input 1 | Sample Output 1 |
| --- | --- |
| 5 3  3 1  1 2  4 5 | 3 1 2 4 5 |
| Sample Input 2 | Sample Output 2 |
| 6 6  1 2  2 3  4 3  4 5  5 6  6 4 | IMPOSSIBLE |
| Sample Input 3 | Sample Output 3 |
| 8 10  1 2  1 4  2 4  2 5  2 3  4 6  4 5  6 5  5 3  7 8 | 1 2 4 6 5 3 7 8 |

**[Strongly Connected Component (SCC) has been excluded from the Summer 2024 final syllabus. We recommend that you self-study the topic later and solve task 03.]**

**Task 03**

You are given a Directed Graph consisting of **N** vertices and **M** edges. You have to find the strongly connected components of the given graph.

**Input:**

The given map will be a directed and unweighted graph.

The first line contains two integers N and M (1 <= N, M <= 100) — the number of vertices and the total number of edges.

The next M lines will contain two integers ui, vi (1 <= ui, vi <= N)— denoting there is a road between ui to vi.

**Output**

Print all the strongly connected components of the given graph. See the output for better understanding.

| Sample Input 1 | Sample Output 1 |
| --- | --- |
| 5 5  1 2  2 3  2 4  3 1  4 5 | 1 2 3  4  5 |
| Sample Input 2 | Sample Output 2 |
| 8 9  1 2  1 6  2 3  3 4  4 5  4 6  5 7  6 2  7 5 | 1  2 3 4 6  5 7  8 |
| Sample Input 3 | Sample Output 3 |
| 4 3  1 2  2 3  2 4 | 1  2  3  4 |