CS F222 Discrete Structures for CS
I Semester 2022-2023
BITS PILANI, Hyderabad Campus
Programming Assignment

Part A Deadline: Mon, Dec 5, 11pm.

Objective:

The main objective of the assignment is to report if two undirected graphs are isomorphic or not.

The assignment will consist of two parts.

You can assume that the given graphs are not multigraphs. Your programs must be in C.

Part A:

Given a graph, you need to output the degree sequence of the graph.
The *degree sequence* is obtained by determining the degrees of each node in the graph and arranging it as a sequence in non-increasing order. Recall from class that the degree sequence is a *graph invariant.*

You would be given the name of the input file as a command-line argument.

The first line of the input would be an integer n - the number of nodes.
Second line of the input would be an integer e - the number of edges
The next e lines contain a pair x and y ($1 \leq x,y \leq n$ and $x \neq y$) , to indicate that there is an edge between nodes numbered x and y.

You need to output the degree sequence in non-increasing order.
Output should be printed in standard output.

You can assume the value of $n \leq 20$, $e \leq 45$.

Test Case 1:

Input (a.txt):

```
5
6
1 2
1 3
4 5
3 5
```

2 3
1 4

Output (out-a.txt):

3 3 2 2 2

Test Case 2:

Input (b.txt):

7
10
1 2
1 5
1 4
2 3
3 4
4 8
5 6
5 8
6 7
7 8

Ouput (out-b.txt):

3 3 3 3 2 2 2 2

General Instructions:

- The code must be written in C only.
- This assignment will be done in groups of two students.
- The name of the file should be TeamX.c, where X is your Team number, as entered in the Team Sign-up spreadsheet.
- Comment the name and ID of both team members at the starting of the program.
- Your program would be tested against more test cases other than the samples provided here. So you should take care of the edge cases.

Some Suggestions:

- You can use adjacency matrix as a data structure for your graph.
- You can use different functions to organise different tasks so that your program becomes more readable, simpler and more modular.