University of Colombo School of Computing
SCS 1208 - Data Structures and Algorithms II

Lab Sheet 1

---

1. Develop an algorithm to check whether there is a path from the first given vertex to the second given vertex in a directed graph.

2. Write an algorithm to check whether a given directed graph contains a cycle or not.

3. Develop an algorithm to count the total number of paths that exist between two vertices in a directed graph. Assume those directed graphs don't contain cycles.

4.

   You have to write a program that predicts when every user will be linked and mimics the expansion of a social network. A list of friendships between users, represented by pairs of IDs, is provided to you. Creating a function that calculates the bare minimum of friend connections required for every user in the network to be linked is your task.

   Create the function minFriendConnections and provide it the aforementioned information:

   n: An integer (IDs ranging from 0 to n-1) that indicates the number of users.
   friendships: A collection of tuples that show a user's friendships with other users. Two integers that reflect the IDs of friends' users are contained in each tuple.
   The function needs to yield the bare minimum of extra friend connections required in order for every user to be linked in the network.

   Consider using a graph-based approach (such as Union-Find or graph traversal algorithms like DFS or BFS) to solve this problem efficiently.