Programming and Data Structures Lab (CS2710)
Assignment-10: *Graph*

Lab-work:

1. Implement Topological Sorting / Ordering of Directed Acyclic Graph:
   Topological sorting for Directed Acyclic Graph (DAG) is a linear ordering of vertices such that for every directed edge from a vertex u to a vertex v, the vertex u comes before the vertex v in the ordering. Topological Sorting for a graph is not possible if the graph is not a DAG.
   Inputs: A DAG, G
   Outputs: The topological ordering of vertices of G

2. Determine Strongly Connected Component of a Graph:
   A directed graph is strongly connected if there is a path between all pairs of vertices. A strongly connected component (SCC) of a directed graph is a maximal strongly connected subgraph.
   Inputs: A directed graph, G
   Outputs: All the strongly connected components (subgraphs) of G

- -

Home-work:

1. Detect Cycles in Undirected Graph:
   Given an undirected graph, how to check if there is a cycle in the graph?
   Inputs: An undirected graph, G
   Outputs: Existance of a cycle or not in G
   Hint: We do a DFS traversal of the given graph. For every visited vertex v, if there is an adjacent u such that u is already visited and u is not parent of v, then there is a cycle in graph. If we do not find such an adjacent for any vertex, we say that there is no cycle. The assumption of this approach is that there are no parallel edges between any two vertices.

2. Find all possible words in a board of characters:
   Given a dictionary, a method to do lookup in dictionary and a M x N board where every cell has one character. Find all possible words that can be formed by a sequence of adjacent characters. Note that we can move to any of 8 adjacent characters, but a word should not have multiple instances of same cell.
   Example:

   ```
   Inputs =>
   dictionary[ ] =    { "GEEKS", "FOR", "QUIZ", "GO" };
   board[ ][ ] =      { {'G', 'I', 'Z'},
                        {'U', 'E', 'K'},
                        {'Q', 'S', 'E'} };
   isWord(str): a function that returns true if str is present in dictionary; else false
   Outputs =>         Following words of dictionary are present: GEEKS, QUIZ
   ```

3. Longest Path in a Directed Acyclic Graph (DAG)
   Given a Weighted DAG and a source vertex s in it, find the longest distances from s to all other vertices in the given graph.
   Inputs: A weighted DAG, G, with the source vertex, s
   Outputs: A list of distances of all the vertices in G from s
   Hint: Use topological ordering