**CSC 413/513 Fall 2019 - Homework #6 ----- Due Date: 10/11/2019**

**Problem 1 (30 points)**

Given a directed graph G below with its adjacency lists, draw the graph search results according to the following cases.

adj(A) = [B],

adj(B) = [C, E],

adj(C) = [G, H, I],

adj(D) = [B, E],

adj(E) = [F],

adj(F) = [D]

adj(G) = [E]

adj(H) = [G, I]

adj(I) = [B]

adj(J) = [B, D]

a) Depth First Search, starting from A

b) Breadth First Search, starting from A

c) Depth First Search, starting from D, after changing the above graph to undirected

d) Breadth First Search, starting from D, after changing the above graph to undirected

**Problem 2 (20 pts)**

a) Use the graph in Problem 1, list all tree edges, back edges, forward edges, and cross edges after performing depth first search starting from J

b) Convert the graph in Problem 1 to undirected graph. List all tree edges, and back edges after performing depth first search starting from J

**Problem 3 (30 pts)**

Given a graph G = (V, E).

a) Design an algorithm to print all the shortest path from a starting node s to all other nodes

b) (Mandatory for CSC 513 students, but optional for CSC 413 students. CSC 413 students will receive bonus points if solving this questions)

Implement your proposed algorithm above in your preferred programming languages.

You can choose to represent the edges in such graph by either adjacency matrix or adjacency list.

**Problem 4 (20 pts)**

A graph can contain multiple connected components. Each connected component is a subgraph which is a connected graph (a simple graph with no self-loop and there is a path to connect any two vertices in such graph). Figure 21.1a in page 563 of CLRS textbook displays an example of a graph with four connected components. Design an optimal algorithm based on DFS or BFS to find the number of connected components of an undirected graph G = (V, E).