

## Assignment 6

1. Write a program to perform a topological sort on a graph (Note: you may choose the graph on page 32 in Course\_12 as input)
  - a) Implement with adjacency list.
  - b) Implement the routines, e.g., InitializeGraph(), InsertEdge(), InDegree(), OutDegree(), etc.
  - c) Store the graph information in a text file;
  - d) Implement a ReadGraph() routine to call InitializeGraph() and InsertEdge() to construct the graph in memory.
  - e) Output the topological sort result to screen.
2. Write a program to solve the single-source shortest-path problem, i.e., Dijkstra algorithm (Note: you may choose the graph on page 61 in Course\_12 as input and vertex 0 as start vertex)
  - a) Implement with adjacency list.
  - b) Implement the routines, e.g., InitializeGraph(), InsertEdge(), InDegree(), OutDegree(), etc.
  - c) Store the graph information in a text file;
  - d) Implement a ReadGraph() routine to call InitializeGraph() and InsertEdge() to construct the graph in memory.
  - e) Output all shortest paths from vertex 0 to screen.
3. Write a program of the Floyd-Warshall algorithm (Note: you may choose the graph on page 31 in Course\_13 as input).
  - a) Implement with adjacency matrix (i.e., a two-dimensional array).
  - b) Store the graph information in a text file;
  - c) Implement a ReadGraph() routine to construct the graph in memory.
  - d) Output the pairwise short paths together with shortest lengths to screen.

Note:

- Due date: Jan. 2nd, 2019.