Preliminary:

• Implement the algorithm and analyze the results using the give input files

• **Deliverables:** Report.pdf file and your code file (**please do not send a zip file**. If you have more than one class in your code, then submit each file separately through Canvas.)

• Homework report must follow the guidelines provided in the sample report uploaded on Canvas

Objectives:

- Implement Dijkstra's Algorithm to find the single source shortest paths for a directed graph.

- Write a detailed report on your program

Problems:

1. Implement Dijkstra's algorithm using the pseudocode algorithm.

2. Write a driver program, which reads input files tinyDG.txt, mediumDG.txt, and largeDG.txt (downloadable from Canvas) and run Dijkstra's algorithm on each of them to find the Single Source Shortest Path within these graphs considering 0 as the source. Your output should be the shortest path

DIJKSTRA(G, w, s)

```
1   INITIALIZE-SINGLE-SOURCE(G, s)
2   S = ∅
3   Q = G.V
4   while Q ≠ ∅
5       u = EXTRACT-MIN(Q)
6       S = S ∪ {u}
7       for each vertex v ∈ G.Adj[u]
8           RELAX(u, v, w)
```

$$\text{RELAX}(u, v, w)$$

1   **if** $v.d > u.d + w(u, v)$
2       $v.d = u.d + w(u, v)$
3       $v.\pi = u$

```
initialize_single_source( Graph g, Node s )
   for each vertex v in Vertices( g )
       d[v] := infinity
       pi[v] := nil
   d[s] := 0;
```

DATA

largeDG.txt

mediumDG.txt

tinyDG.txt

Grading Rubric

Coding	Implementing Algorithms	20 points
	Producing Correct Outputs	20 points
Report	Explaining the algorithms used	10 points
	Displaying the output with a graph or table	20 points
	Comparing the outputs and discussing the time complexity of algorithms	20 points

Correct submissions of the files (ICF, Code.zip, report.pdf)	10 points