Assignment 8 – Dijkstra’s

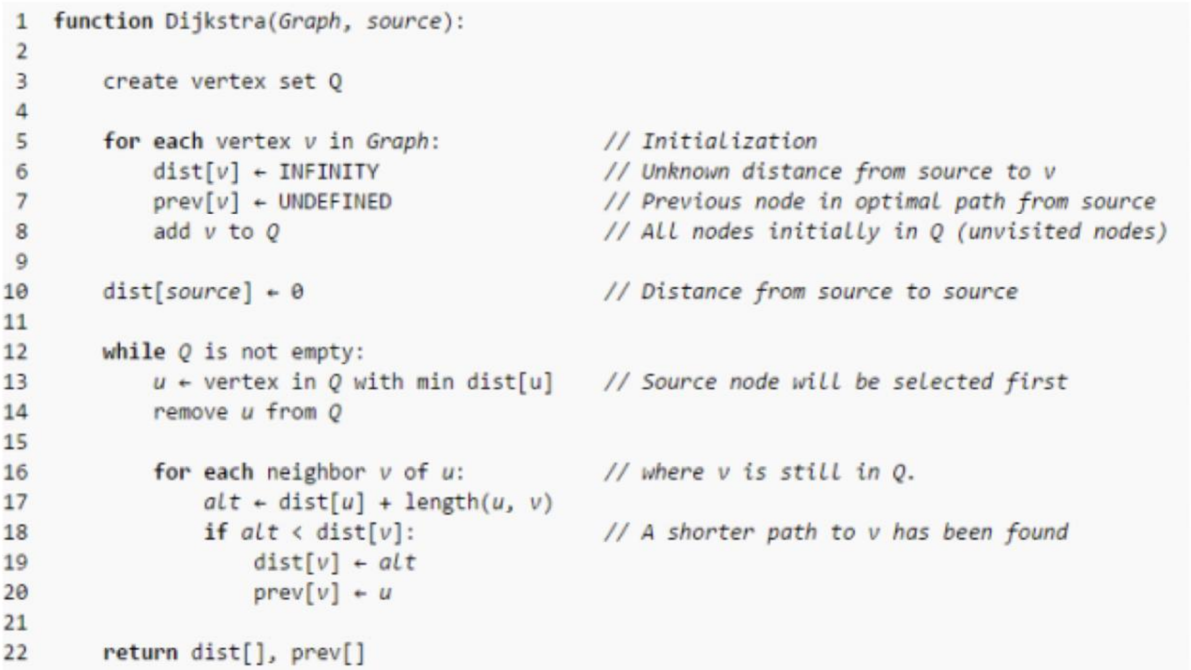
Submission

Submit your code to the Blackboard Assignment by the due date. Please submit a screen shot(s) of your working code in a word or pdf document or submit a short video of your working code.

Details

Write a Java/C/Python program (you may use another language as well) to implement Dijkstra’s algorithm.

Pseudocode for the algorithm shown below:

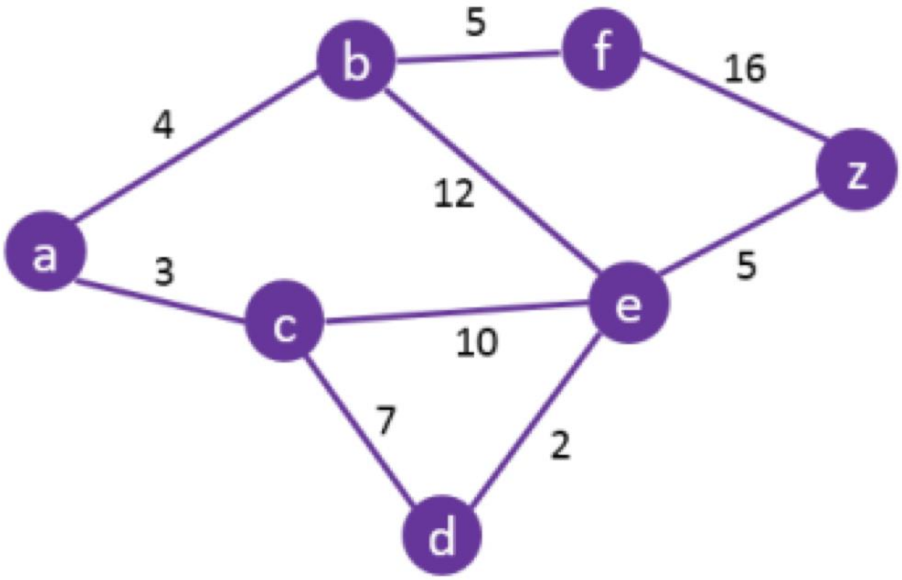


```
function Dijkstra(Graph, source):

    create vertex set Q

    for each vertex v in Graph:            // Initialization
        dist[v] ← INFINITY                 // Unknown distance from source to v
        prev[v] ← UNDEFINED                // Previous node in optimal path from source
        add v to Q                         // All nodes initially in Q (unvisited nodes)

    dist[source] ← 0                       // Distance from source to source

    while Q is not empty:
        u ← vertex in Q with min dist[u]   // Source node will be selected first
        remove u from Q

        for each neighbor v of u:          // where v is still in Q.
            alt ← dist[u] + length(u, v)
            if alt < dist[v]:              // A shorter path to v has been found
                dist[v] ← alt
                prev[v] ← u

    return dist[], prev[]
```

Output:

The output of your program should. be the arrays dist and prev. Recall that the dist[I] is the distance from the start of the vertex to vertex i. prev[i] is the vertex that immediately precedes vertex i on the shortest path from the start vertex to vertex i.

Use the following graph as input to your program



You may number the vertices if you prefer: a=0, b=1, c=2, d=3, e=4, f=5, and z=6.