Exp - 10

VAISHNAVI S
1BM18CS121

Question : Demonstrate Dijkstra algo

Ans :

Code :

```
import sys

class Graph:
    def __init__(self, vertices):
        self.V = vertices
        self.graph = [[0 for column in range(vertices)]
                      for row in range(vertices)]

    def printSolution(self, dist):
        print("Vertex \t Distance from Source")
        for node in range(self.V):
            print(node, "\t", dist[node])

    def minDistance(self, dist, sptSet):
        min = sys.maxsize
        for v in range(self.V):
            if dist[v] < min and sptSet[v] == False:
                min = dist[v]
                min_index = v
        return min_index
```

```
def minDistance(self, dist, sptSet):
    min = sys.maxsize
    for v in range(self.V):
        if dist[v] < min and sptSet[v] == False:
            min = dist[v]
            min_index = v
    return min_index

def dijkstra(self, src):
    dist = [sys.maxsize] * self.V
    dist[src] = 0
    sptSet = [False] * self.V

    for cout in range(self.V):
        u = self.minDistance(dist, sptSet)
        sptSet[u] = True
        for v in range(self.V):
            if self.graph[u][v] > 0 and
               sptSet[v] == False and
               dist[v] > dist[u] + self.graph[u][v]:
                dist[v] = dist[u] + self.graph[u][v]
        self.printSolution(dist)

g = Graph(8)
g.graph = [ [0, 6, 0, 0, 0, 0, 0, 8],
            [4, 0, 8, 0, 0, 0, 0, 1],
            [0, 8, 0, 7, 0, 4, 0, 2],
            [0, 0, 0, 0, 0, 10, 0, 0],
            [0, 0, 4, 14, 10, 0, 2, 10],
            [0, 0, 0, 0, 0, 2, 1, 6],
            [8, 11, 0, 0, 0, 1, 8, 1] ]
g.dijkstra(0)
```