Dijkstra's Algorithm

VYSHNAVI M
1BM18CS125

```
class Graph():
    def __init__(self, vertices):
        self.V = vertices
        self.graph = [[0] * vertices] * vertices

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

    def minDistance(self, dist, sptSet):
        min = sys.maxint
        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.maxint] * self.V
        dist[src] = 0
        sptSet = [False] * self.V
        for count 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[u] >
                    dist[u] + self.graph[u][v]:
                        dist[v] = dist[u] + self.graph[u][v]
```