

Dijkstra's Algorithm

Code:

In [10]:

```

INF = 9999999
V = 9
graph = [[0, 4, 0, 0, 0, 0, 0, 8, 0],
          [4, 0, 8, 0, 0, 0, 0, 11, 0],
          [0, 8, 0, 7, 0, 4, 0, 0, 2],
          [0, 0, 7, 0, 9, 14, 0, 0, 0],
          [0, 0, 0, 9, 0, 10, 0, 0, 0],
          [0, 0, 4, 14, 10, 0, 2, 0, 0],
          [0, 0, 0, 0, 0, 2, 0, 1, 6],
          [8, 11, 0, 0, 0, 0, 1, 0, 7],
          [0, 0, 2, 0, 0, 0, 6, 7, 0]]

def printSolution(dist):
    print("Vertex -> Distance")
    for node in range(V):
        print(" " + str(node) + " -> " + str(dist[node]))

def minDistance(dist, selected_node):
    min = INF
    for v in range(V):
        if dist[v] < min and selected_node[v] == False:
            min = dist[v]
            min_index = v

    return min_index

def dijkstra(src):
    dist = [INF] * V
    selected_node = [False] * V
    dist[src] = 0

    for node in range(V):
        u = minDistance(dist, selected_node)
        selected_node[u] = True
        for v in range(V):
            if (graph[u][v] > 0 and selected_node[v] == False and dist[v] > dist[u] + gr
                dist[v] = dist[u] + graph[u][v]

    printSolution(dist)

dijkstra(0)

```

#Output

Vertex -> Distance

```

0 -> 0
1 -> 4
2 -> 12
3 -> 19
4 -> 21
5 -> 11
6 -> 9
7 -> 8
8 -> 14

```

