Assignment No-8

Name- PRAFUL SHEWALE
Roll no=SI- 48   Class=SE-IT

Aim- DIJIKSTRA'S ALGORITHM

```
#include <iostream>
using namespace std;
#include <limits.h>
#define V 9
int minDistance(int dist[], bool sptSet[])
{
 int min = INT_MAX, min_index;// INT_MAX is macros having Maximum value
2147483647 for 32 bit
 for (int v = 0; v < V; v++)
 if (sptSet[v] == false && dist[v] <= min)
 min = dist[v], min_index = v;
 return min_index;
}
void printSolution(int dist[])
{
 cout << "Vertex \t\t\tDistance from Source" << endl;
 for (int i = 0; i < V; i++)
 cout << i << " \t\t\t\t" << dist[i] << endl;
}
void dijkstra(int graph[V][V], int src)
{
 int dist[V]; // The output array. dist[i] will hold the
 bool sptSet[V]; // sptSet[i] will be true if vertex i is
for (int i = 0; i < V; i++)
 dist[i] = INT_MAX, sptSet[i] = false;
 dist[src] = 0;
 for (int count = 0; count < V - 1; count++) {
 int u = minDistance(dist, sptSet);
 sptSet[u] = true;
 for (int v = 0; v < V; v++)
if (!sptSet[v] && graph[u][v]
 && dist[u] != INT_MAX
 && dist[u] + graph[u][v] < dist[v])
 dist[v] = dist[u] + graph[u][v];
 }
 printSolution(dist);}
int main(){
 /* Let us create the example graph discussed above */
 int graph[V][V] = { { 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 }  };
dijkstra(graph, 0);
 return 0;}
```

OUTPUT

```
Vertex                  Distance from Source
0                       0
1                       4
2                       12
3                       19
4                       21
5                       11
6                       9
7                       8
8                       14


=== Code Execution Successful ===
```