Data Structures and Algorithms
CSE2001
Lab - 7 - Assignment - 2

Yashwanth Reddy
19BCE7362
Date- 20thJuly2021

Problem : Write a Program to Implement Dijkstra Shortest Path Algorithm

Code

```java
class Graph_Shortest_Path {

  static final int num_Vertices = 6;
  int minDistance(int path_array[], Boolean sptSet[])  {


    int min = Integer.MAX_VALUE, min_index = -1;
    for (int v = 0; v < num_Vertices; v++)
      if (sptSet[v] == false && path_array[v] <= min) {
        min = path_array[v];
        min_index = v;
      }
    return min_index;
  }
  void printMinpath(int path_array[])  {
    System.out.println("Vertex# \t Minimum Distance from Source");
    for (int i = 0; i < num_Vertices; i++)
      System.out.println(i + " \t\t\t " + path_array[i]);
```

```java
    }

    void algo_dijkstra(int graph[][], int src_node) {
        int path_array[] = new int[num_Vertices];
        Boolean sptSet[] = new Boolean[num_Vertices];
        for (int i = 0; i < num_Vertices; i++) {
            path_array[i] = Integer.MAX_VALUE;
            sptSet[i] = false;
        }
        path_array[src_node] = 0;
        for (int count = 0; count < num_Vertices - 1; count++) {
            int u = minDistance(path_array, sptSet);
            sptSet[u] = true;
            for (int v = 0; v < num_Vertices; v++)
                if (!sptSet[v] && graph[u][v] != 0 && path_array[u] !=
                        Integer.MAX_VALUE && path_array[u]
                        + graph[u][v] < path_array[v])
                        path_array[v] = path_array[u] + graph[u][v];
        }
        printMinpath(path_array);
    }
}
class Dijkstra_Algo{
    public static void main(String[] args)
    {
        int graph[][] = new int[][] { { 0, 2, 1, 0, 0, 0},
                                      { 2, 0, 7, 0, 8, 4},
                                      { 1, 7, 0, 7, 0, 3},
                                      { 0, 0, 7, 0, 8, 4},
                                      { 0, 8, 0, 8, 0, 5},
                                      { 0, 4, 3, 4, 5, 0} };
```

```
        Graph_Shortest_Path g = new Graph_Shortest_Path();

        g.algo_dijkstra(graph, 0);
    }
}
```

Output

```
C:\Users\yashw\Desktop\Summer\Labs>javac Dijkstra_Algo.java --release 8

C:\Users\yashw\Desktop\Summer\Labs>java Dijkstra_Algo
Vertex#         Minimum Distance from Source
0                       0
1                       2
2                       1
3                       8
4                       9
5                       4

C:\Users\yashw\Desktop\Summer\Labs>
```