Lab-9

Pranjal Sinha
113216050

```
#define V 9
int minDistance(int dist[], bool sptSet[])
{
    int min = INT_MAX, min_index

    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[])
{
    printf("Vertex \t\t Distance from Source\n");
    for(int i=0; i<V; i++)
        printf("%d \t\t %d\n", i, dist[i]);
}
void dijkstra(int graph[V][V], int src)
{
    int dist[V];
    bool sptSet[V];
    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() {
    int graph[V][V];
    cout << "Enter graph" << endl;
    for (int i = 0; i < V; i++)
    {
        for (int j = 0; j < V; j++)
            cin >> graph[i][j];
    }
    dijkstra(graph, 0);
    return 0;
}
```