Pooja Srinivasan
5'B'
1BM18CS069

Lab-9 WAP to implement Dijkstra's algo TF shortest path

```
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);
```

Date
Page

```
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);
}
```