**From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm. Write the program in C.**

#include<stdio.h>

int minimum(int,int);

int main()

{

int cost[20][20],s[20],d[20]; int source,n,mini,u;

int i,j,v;

printf("\n\n\t Dijkstra's Algorithm \nEnter the number of verticies : "); scanf("%d",&n);

printf("Enter the weights of the graph\n");

printf("If no connection enter 999 and for self loop enter 0\n"); for(i=1;i<=n;i++)

for(j=1;j<=n;j++)

scanf("%d",&cost[i][j]);

printf("Enter the source node : "); scanf("%d",&source);

for(i=1;i<=n;i++)

{

s[i]=0;

d[i]=cost[source][i];

}

s[source]=1;

for(i=1;i<=n-1;i++)

{

mini=999; u=0;

for(j=1;j<=n;j++)

if(d[j]<mini && s[j]==0)

{

mini=d[j]; u=j;

} s[u] = 1;

for(v=1;v<=n;v++)

if(s[v]==0)

d[v]=minimum(d[v],d[u]+cost[u][v]);

}

for(i=1;i<=n;i++)

printf("Shortest Path From %d to %d is = %d\n",source,i,d[i]);

}

int minimum(int a,int b)

{

return((a<b)?a:b);

}

Output:

Enter the number of verticies : 4

Enter the weights of the graph

If no connection enter 999 and for self loop enter 0

1 0 9 3

0 0 999 2

0 999 3 4

999 1 2 3

Enter the source node : 1 Shortest Path From 1 to 1 is = 1

Shortest Path From 1 to 2 is = 0

Shortest Path From 1 to 3 is = 4