

9.From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's Algorithm

Code:

```
#include<stdio.h>
#include<conio.h>
void dijkstras();
int c[10][10],n,src;
void main()
{
    int i,j;
    printf("\nEnter number of vertices:");
    scanf("%d",&n);
    printf("\nEnter the adjacency matrix:\n");
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=n;j++)
        {
            scanf("%d",&c[i][j]);
        }
    }
    printf("\nEnter the source node:\t");
    scanf("%d",&src);
    dijkstras();
    getch();
}
```

```
void dijkstras()
{
    int vis[10],dist[10],u,i,j,count,min;
    for(j=1;j<=n;j++)
    {
        dist[j]=c[src][j];
    }
    for(j=1;j<=n;j++)
    {
        vis[j]=0;
    }
    dist[src]=0;
    vis[src]=1;
    count=1;
    while(count!=n)
    {
```

```

min=9999;
for(j=1;j<=n;j++)
{
    if(dist[j]<min&&vis[j]!=1)
    {
        min=dist[j];
        u=j;
    }
}
vis[u]=1;
count++;
for(j=1;j<=n;j++)
{
    if(min+c[u][j]<dist[j]&&vis[j]!=1)
    {
        dist[j]=min+c[u][j];
    }
}
}
printf("\nThe shortest distance is:\n");
for(j=1;j<=n;j++)
{
    printf("\n%d to %d=%d ",src,j,dist[j]);
}
}

```

Output:

```

Enter number of vertices:6

Enter the adjacency matrix:
0 25 35 999 100 999
999 0 100 14 999 999
999 999 0 29 999 999
999 999 999 0 999 21
999 999 50 999 0 999
999 999 999 999 48 0

Enter the source node: 1

The shortest distance is:

1 to 1=0
1 to 2=25
1 to 3=35
1 to 4=39
1 to 5=100
1 to 6=60

```