Lab Question 13: Graph – Shortest Path

Aim:
To write a C program to find the shortest path using Dijkstra's algorithm.

Algorithm:

1. Start the program.
2. Represent graph using adjacency matrix.
3. Initialize distances as infinity except source = 0.
4. Pick minimum distance unvisited vertex.
5. Update its adjacent distances.
6. Repeat until all visited.
7. Stop.

Code (short version):

```
#include <stdio.h>
#define V 5
#define INF 9999

int minDist(int dist[],int vis[]){
   int min=INF,idx=-1;
   for(int i=0;i<V;i++) if(!vis[i]&&dist[i]<min){min=dist[i];idx=i;}
   return idx;
}

void dijkstra(int g[V][V],int src){
   int dist[V],vis[V]={0};
   for(int i=0;i<V;i++) dist[i]=INF;
   dist[src]=0;
   for(int c=0;c<V-1;c++){
      int u=minDist(dist,vis);
      vis[u]=1;
      for(int v=0;v<V;v++)
         if(!vis[v]&&g[u][v]&&dist[u]+g[u][v]<dist[v])
```

```
            dist[v]=dist[u]+g[u][v];
    }
    printf("Vertex\tDistance\n");
    for(int i=0;i<V;i++) printf("%d\t%d\n",i,dist[i]);
}

int main(){
    int g[V][V]={{0,10,0,0,5},{0,0,1,0,2},{0,0,0,4,0},{7,0,6,0,0},{0,3,9,2,0}};
    dijkstra(g,0);
    return 0;
}
```

Output:

- From source 0 distances = [0,8,9,7,5]

Result:
The program finds the shortest path using Dijkstra's algorithm.