```
#include <stdio.h>
#include <limits.h>

#define V 5


void dijkstra(int graph[V][V],int s)
{

    int dist[V];
    int visited[V];
    int parent[V];

    for (int i = 0; i < V; i++)
       {
        dist[i] = INT_MAX;
        visited[i] = 0;
    }


    dist[s] = 0;



    for (int count = 0; count < V - 1; count++) {

        int u,min= INT_MAX;

        for (int i = 0; i < V; i++) {
        if (visited[i] == 0 && dist[i] < min) {
            min = dist[i];
            u = i;
        }
    }


        visited[u] = 1;


        for (int i = 0; i < V; i++) {

            if (graph[u][i]!=0 && visited[i] == 0 && dist[u]+graph[u][i] <
dist[i]) {
                dist[i] = dist[u]+graph[u][i];
                parent[i]=u;

            }
        }
    }


   for (int i = 0; i < V; i++)
         {
            printf(" %d   =  %d \n", i, dist[i]);
        }
        printf("\n");

        for (int i = 0; i < V; i++)
```

```
          {
             printf(" %d   =  %d \n", i, parent[i]);
          }
}


int main() {

    int graph[V][V] = {
        {0,10,0,30,100},
        {10,0,50,0,0},
        {0,50,0,20,10},
        {30,0,20,0,60},
        {100,0,10,60,0},

    };

    for (int i = 0; i < V; i++) {
        for (int j = 0; j < V; j++) {
            printf("  %d \t", graph[i][j]);
        }
        printf("\n");
    }

    printf("\n");
    dijkstra(graph,0);

    return 0;
}
```