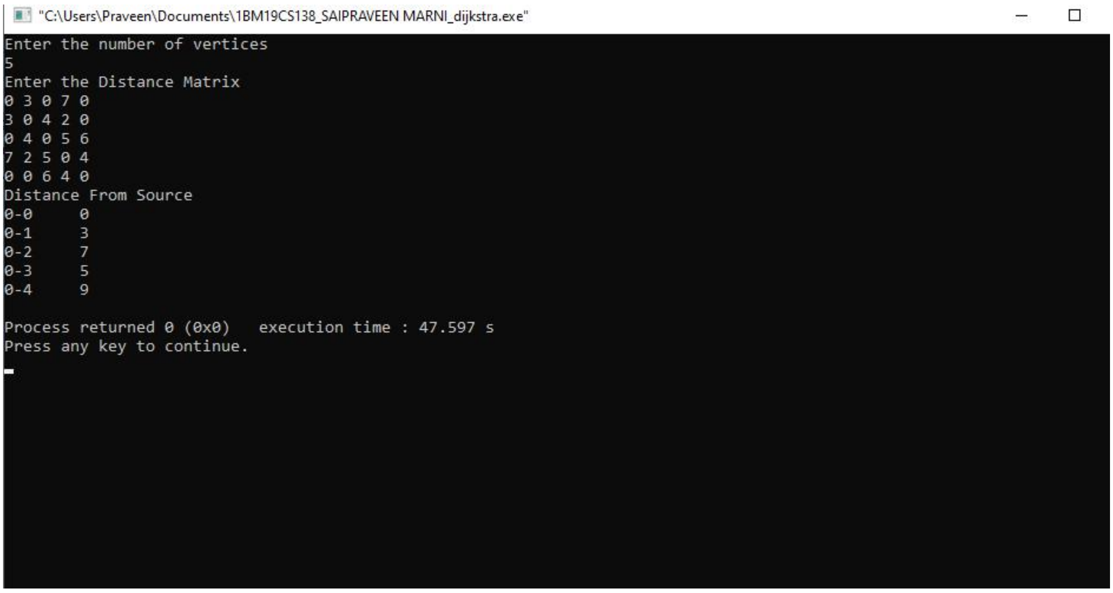
Name: Sai Praveen Marni
USN: 1BM19CS138

Dijkstras algortihm:

```
#include<stdio.h>
#include<limits.h>
#include<stdbool.h>
int V;
int minKey(int key[],bool mstset[])
{
    int min,minIndex;
    min=INT_MAX;
    for(int i=0;i<V;i++)
    {
        if(mstset[i]==false && key[i]<min)
        {
            min=key[i];
            minIndex=i;
        }
    }
    return minIndex;
}
void printmst(int key[])
{
    int sum=0;
    printf("Distance From Source\n");
    for(int i=0;i<V;i++)
    {
        printf("%d-%d\t%d\n",0,i,key[i]);

    }
}
void primst(int graph[V][V])
{
    int parent[V];
    int key[V];
    bool mstset[V];
    int sum=0;
    for(int i=0;i<V;i++)
    {
        parent[i]=0;
        key[i]=INT_MAX;
        mstset[i]=false;
    }
    key[0]=0;
    parent[0]=-1;
    for(int count=0;count<V-1;count++)
    {
        int u=minKey(key,mstset);
        mstset[u]=true;
        for(int v=0;v<V;v++)
        {
            if(graph[u][v] && mstset[v]==false && key[u]!=INT_MAX &&
key[u]+graph[u][v]<key[v])
            {
                parent[v]=u;
                key[v]=key[u] + graph[u][v];
            }
        }
    }
    printmst(key);
}
int main()
{
    printf("Enter the number of vertices\n");
    scanf("%d",&V);
    int graph[V][V];
    printf("Enter the Distance Matrix\n");
    for(int i=0;i<V;i++)
    {
        for(int j=0;j<V;j++)
        {
            scanf("%d",&graph[i][j]);
        }
    }
    primst(graph);
}
```

Output:



"C:\Users\Praveen\Documents\1BM19CS138_SAIPRAVEEN MARNI_dijkstra.exe"

```
Enter the number of vertices
5
Enter the Distance Matrix
0 3 0 7 0
3 0 4 2 0
0 4 0 5 6
7 2 5 0 4
0 0 6 4 0
Distance From Source
0-0     0
0-1     3
0-2     7
0-3     5
0-4     9

Process returned 0 (0x0)   execution time : 47.597 s
Press any key to continue.
```