```
#include<iostream>
#include<climits>
using namespace std;

int minimumDist(int dist[], bool Tset[])
{
int min=INT_MAX,index;
for(int i=0;i<6;i++)
{
if(Tset[i]==false && dist[i]<=min)
{
min=dist[i];
index=i;
}
}
return index;
}
void Dijkstra(int graph[6][6],int src)
{int dist[6];
bool Tset[6];

for(int i = 0; i<6; i++)
{
dist[i] = INT_MAX;
Tset[i] = false;
}
dist[src] = 0;
for(int i = 0; i<6; i++)
{
int m=minimumDist(dist,Tset);
Tset[m]=true;
for(int i = 0; i<6; i++)
{

if(!Tset[i] && graph[m][i] && dist[m]!=INT_MAX &&
dist[m]+graph[m][i]<dist[i])
dist[i]=dist[m]+graph[m][i];
}
}
cout<<"Vertex\t\tDistance from source"<<endl;
for(int i = 0; i<6; i++)
{
char str=65+i;
cout<<str<<"\t\t\t"<<dist[i]<<endl;
}
}
int main()
{
int graph[6][6]=
{
{0, 10, 20, 0, 0, 0},
{10, 0, 0, 50, 10, 0},
{20, 0, 0, 20, 33, 0},
{0, 50, 20, 0, 20, 2},
```

```
{0, 10, 33, 20, 0, 1},
{0, 0, 0, 2, 1, 0}};

Dijkstra(graph,0);
return 0;
}
```

Output:-

```
Vertex           Distance from source
A                0
B                10
C                20
D                23
E                20
F                21
```