

ADA

Lab test - 2

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

```
int minDistance int number[10];
```

```
int pickIndex (int n, int picked[n], int cost[n])
```

```
{
```

```
    int mini = 0;
```

```
    int min = 999;
```

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

```
{
```

```
    if (picked[i] != 1 && cost[i] < min)
```

```
{
```

```
        min = cost[i];
```

```
        mini = i;
```

```
    }
```

```
}
```

```
    return mini;
```

```
}
```

```
void updateDistances (int n, int row, int index, int cost[n][n],  
                      int adj[n][n])
```

```
{
```

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

```
{
```

```
    if (adj[index][i] != 999 && adj[index][i] + cost[row][indexi] < cost[rowrow][ii])
```

```
    {  
        number[i] += 1;
```

```
        cost[row][i] = adj[index][i] + cost[row][index];
```

```
    }  
}
```

```
int dijkstra (int n, int adj[n][n])
```

```
{ int number[n];
```

```
int nodes[n];
```

```
int cost[n][n];
```

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

```
{
```

```
nodes[i]=0; number[i]=0;
```

```
cost[i][0]=0;
```

```
for (int j=0; j<n; j++)
```

```
{
```

```
cost[i][j]=999;
```

```
}
```

```
}
```

```
int row=0;
```

```
while (row < n)
```

```
{
```

```
int index = getIndex (n, nodes, cost[row]);
```

```
row ++;
```

```
nodes[index]=1;
```

```
updateDistances (n, row, index, cost, adj);
```

```
updateRow (row, n, cost);
```

```
}
```

```
print (n, cost[n-1]);
```

```
}
```

```
int main()
```

```
{  
int arr[5][5] = { { 0, 2, 0, 0, 0 }, { 2, 0, 3, 8, 5 }, { 0, 3, 0, 0, 7 },  
{ 6, 8, 0, 0, 9 }, { 0, 5, 7, 9, 0 } };
```

```
dijkstra (5, arr);
```

```
}
```

```
void print (int n, int cost[n])
```

```
{  
    for (int i=0; i<n; i++)
```

```
{  
    printf ("%d \t", cost[i]);
```

```
} printf ("\n");
```

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

```
{  
    printf ("%d \t", members[i]);
```

```
}
```

```
}
```