


```
d[src] = 0
```

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

```
    int u = minD(dist, s);
```

```
    s[u] = true;
```

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

```
        if (!d[v] && g[u][v] && d[u] !=
```

```
            INT_MAX && d[u] + g[u][v] <
```

```
            d[v])
```

```
                d[v] = d[u] + g[u][v];
```

```
    }
```

```
} print solution (dist);
```

```
int main()
```

```
{
```

```
    int g[V][V];
```

```
    cout << "enter graph" << endl;
```

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

```
    {
```

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

```
            cin >> g[i][j];
```

```
    }
```

```
    dijkstra(g, 0);
```

```
    return 0;
```

```
}
```

OUTPUT:-

enter graph

0 9 2 5

9 0 6 3

2 6 0 0

5 8 0 0

vertex Distance from source

0

0

1

8

2

2

3

5