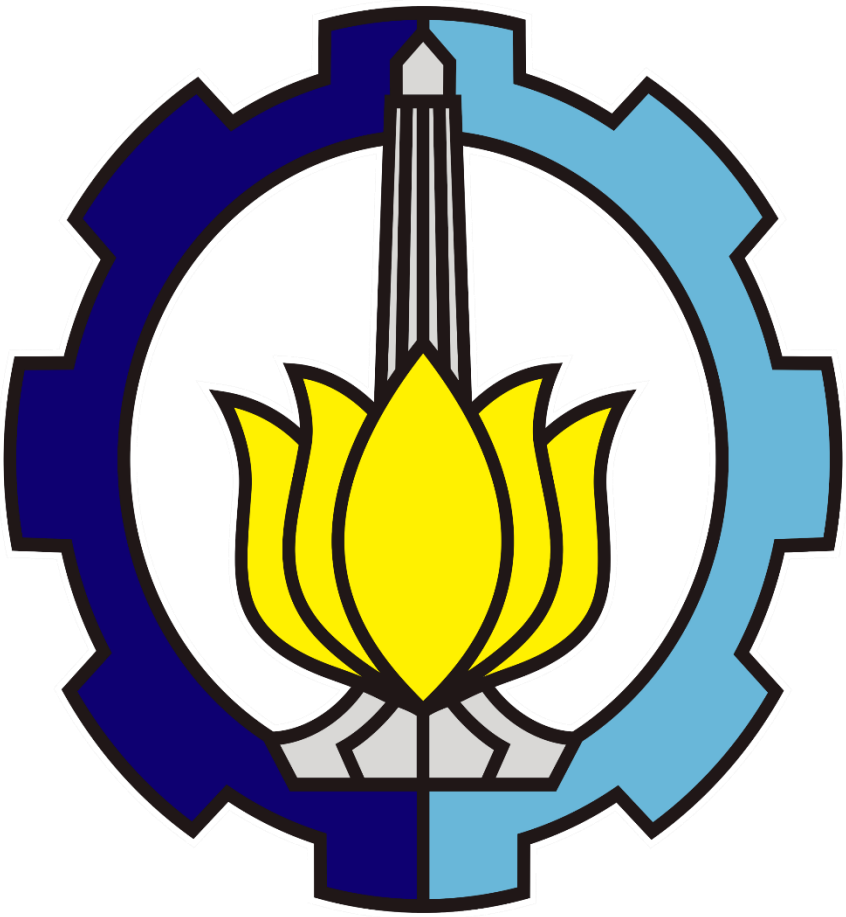
Dijkstra’s Shortest Path

DAA D – Quiz 2



By :

- I Nyoman Yoga Mahottama (05111740000159)
- Faishal Abiyyudzakir (05111740000192)
- Naulia Adam Kuncoro (05111740000155)

DAA-D

Project: Dijkstra's Shortest Path

Design Analysis Algorithms

this program have function to find the shortest distance between cities (vertices). Using Dijkstra's Algorithm to solve this. At first you will be asked to write down the distance(edge) between each cities as written in the program and next you will be asked which one will be the starting city(vertex) and which city will be the destination, then it will compute the shortest distance.

Source Code

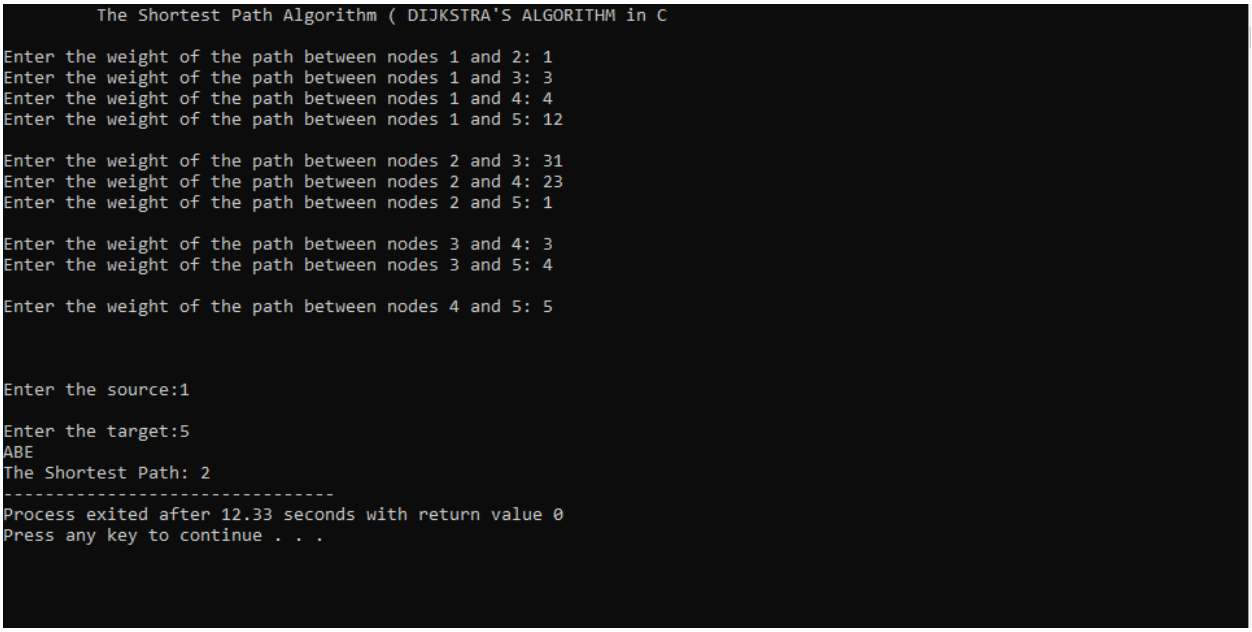
```
#include<stdio.h>
#include<conio.h>
#include<process.h>
#include<string.h>
#include<math.h>
#define IN 99
#define N 5 //change this number the same as the number of the city/vertices you have
int dijkstra(int cost[][N], int source, int target);

int dijsktra(int cost[][N],int source,int target)
{
    int dist[N],prev[N],selected[N]={0},i,m,min,start,d,j;
    char path[N];
    for(i=1;i<= N;i++)
    {
        dist[i] = IN;
        prev[i] = -1;
    }
    start = source;
    selected[start]=1;
    dist[start] = 0;
    while(selected[target] ==0)
    {
        min = IN;
        m = 0;
        for(i=1;i<= N;i++)
        {
            d = dist[start] +cost[start][i];
            if(d< dist[i]&&selected[i]==0)
            {
                dist[i] = d;
                prev[i] = start;
            }
            if(min>dist[i] && selected[i]==0)
            {
                min = dist[i];
                m = i;
            }
        }
        start = m;
        selected[start] = 1;
    }
    start = target;
    j = 0;
    while(start != -1)
    {
        path[j++] = start+64;
        start = prev[start];
    }
    path[j]='\0';
    strrev(path);
    printf("%s", path);
    return dist[target];
}
```

```c
int main()
{
    int cost[N][N],i,j,w,ch,co;
    int source, target,x,y;
    printf("\t The Shortest Path Algorithm ( DIJKSTRA'S ALGORITHM in C \n\n");
    for(i=1;i<= N;i++)
    for(j=1;j<= N;j++)
    cost[i][j] = IN;
    for(x=1;x<= N;x++)
    {
        for(y=x+1;y<= N;y++)
        {
            printf("Enter the weight of the path between nodes %d and %d: ",x,y);
            scanf("%d",&w);
            cost [x][y] = cost[y][x] = w;
        }
        printf("\n");
    }
    printf("\nEnter the source:");
    scanf("%d", &source);
    printf("\nEnter the target:");
    scanf("%d", &target);
    co = dijsktra(cost,source,target);
    printf("\nThe Shortest Path: %d",co);
}
```

How To Use:

Before running the program you have to set how many cities (vertices) you have on line 8(number in front of “N”) , make sure to match it to the amount. After the program run, you will be asked to state the distance between each city (vertices) in line 8 and after that, state the starting city and destination city.

Input and Output:



Analysis:

Based on the picture in Input and Output, we can declare that there are 5 cities (vertices). Each are a letter; A, B, C, D, and E. And there are the declarations of each distance between cities. After being asked which one is the starting city(source) and destination city(target), it calculate the shortest distance it could find and then state the nodes we should go through to achieve that goal.

“By the name of Allah (God) Almighty, herewith we pledge and truly declare that we have solved quiz 2 by ourselves, didn’t do any cheating by any means, didn’t do any plagiarism, and didn’t accept anybody’s help by any means. We are going to accept all of the consequences by any means if it has proven that we have been done any cheating and/or plagiarism.”

Surabaya, 25 March 2020

[I Nyoman Yoga Mahottama]	[Naulia Adam Kuncoro]	[Faishal A]
[05111740000159]	[05111740000155]	[05111740000192]