

LAPORAN STRUKTUR DATA
UAS SEMESTER 2



Au Izaldi Fachril Rahmadani

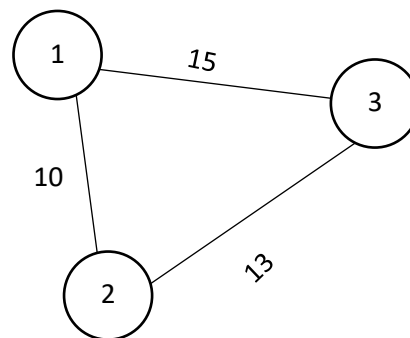
21091397026

PRODI D4 MANAJEMEN INFORMATIKA
MATKUL STRUKTUR DATA
KELAS B
UNIVERSITAS NEGERI SURABAYA

Nomor 1

```
uasstr22.cpp
1 #include<iostream>
2 #define MAX 100
3
4 using namespace std;
5
6 int adjMtx[MAX][MAX];
7
8 void inisialisasiMtx(int v){
9     for(int i = 0; i < v; i++){
10         for(int j = 0; j < v; j++){
11             adjMtx[i][j] = 0;
12         }
13     }
14 }
15
16 void tambahEdge(int u, int v, int w){
17     adjMtx[u][v] = w;
18     adjMtx[v][u] = w;
19 }
20
21 void cetakMtx(int v){
22     for(int i = 1; i <= v; i++){
23         cout<<"\t"<<"(" <<i<<" )";
24     }
25     cout<<endl;
26     for(int i = 1; i <= v; i++){
27         cout<<"(" <<i<<" ) "<<"\t";
28         for(int j = 1; j <= v; j++){
29             cout<<adjMtx[i][j]<<"\t";
30         }
31         cout<<endl;
32     }
33 }
34
35
36 int main(){
37     int vertex = 3;
38
39     inisialisasiMtx(vertex);
40
41     tambahEdge(2, 1, 10);
42     tambahEdge(3, 1, 15);
43     tambahEdge(3, 2, 13);
44
45     cetakMtx(vertex);
46
47 }
```

	(1)	(2)	(3)
(1)	0	10	15
(2)	10	0	13
(3)	15	13	0



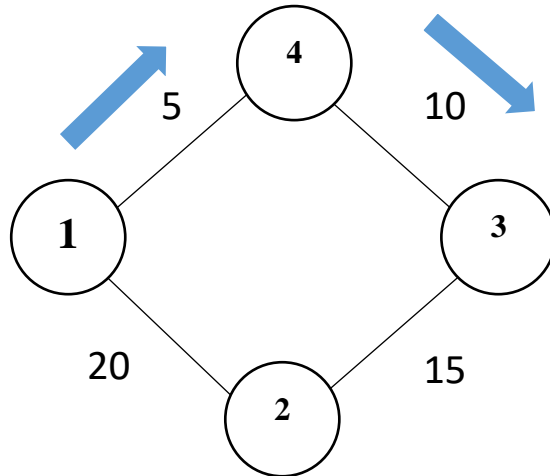
```
C:\Users\Dani PC\Documents\C++\uasstr22.exe
(1)    (2)    (3)
(1)    0      10     15
(2)    10     0      13
(3)    15     13     0

-----
Process exited after 4.028 seconds with return value 0
Press any key to continue . . .
```

Nomor 2

```
uasstr22.cpp uasstrd.cpp
1 #include <stdio.h>
2 #include <stdbool.h>
3 #define infinite 999
4
5 int n;
6 int tabel[100][100];
7
8 int dijkstra(int awal, int akhir){
9     int jarak[n];
10    bool visit[n];
11    int i, j, v;
12
13    //inisialisasi dari jarak[] = infinite
14    for(i = 1; i <= n; i++){
15        jarak[i] = infinite;
16    }
17
18    //inisialisasi dari jarak visit[] = false
19    for(i = 1; i <= n; i++){
20        visit[i] = false;
21    }
22
23    jarak[awal] = 0;
24    while(true){
25        int u = -1;
26        int jarakMin = infinite;
27        //mencari vertex yang belum dikunjungi
28        for(i = 1; i <= n; i++){
29            if((visit[i] == false) && (jarak[i] < jarakMin)){
30                u = i;
31                jarakMin = jarak[i];
32            }
33        }
34        //untuk mengakhiri perulangan while
35        if((u == -1) || (jarak[u] == infinite)){
36            break;
37        }
38        visit[u] = true;
39        //melakukan relax untuk semua neighbor u
40        for(v = 1; v <= n; v++){
41            if(tabel[u][v] != 0){
42                if(jarak[v] > jarak[u] + tabel[u][v]){
43                    jarak[v] = jarak[u] + tabel[u][v];
44                }
45            }
46        }
47    }
48    //mengembalikan jarak terdekat dari kedua vertex
49    return jarak[akhir];
50 }
51
52 int main(){
53     int awal, akhir;
54     int i, j;
55
56     printf("Masukkan Banyak Kota\t: ");
57     scanf("%d", &n);
58
59     printf("Masukkan Jarak Vertex secara Matriks :\n");
60
61     for(i = 1; i <= n; i++){
62         for(j = 1; j <= n; j++){
63             scanf("%d", &tabel[i][j]);
64         }
65     }
66
67     int jawab = 0, x;
68     //menggunakan perulangan agar bisa di tes berulang kali
69     while(jawab == 0){
70         printf("Kota Tempat Pedagang Berada\t: ");
71         scanf("%d", &awal);
72         printf("Kota Tempat Kastil\t\t: ");
73         scanf("%d", &akhir);
74
75         // fungsi dijkstra
76         printf("Jarak minimal dari kota (Pedagang) %d ke (Kastil) %d: %d\n ", awal, akhir, dijkstra(awal, akhir));
77
78         ulang :
79         printf("Apakah ingin melanjutkan program? <y/n>");
80         scanf("%s", &x);
81         if(x == 'y' || x == 'Y'){
82             jawab = 0;
83         }else if(x == 'n' || x == 'N'){
84             printf("Terima Kasih");
85             jawab = 1;
86         }else{
87             printf("Input salah masukkan y atau n");
88             goto ulang;
89         }
90     }
91 }
92
93
94
95
96
97
98
```

	(1)	(2)	(3)	(4)
(1)	0	20	0	5
(2)	20	0	15	0
(3)	0	15	0	10
(4)	5	0	10	0



```

C:\Users\Dani PC\Documents\C++\uasstrd.exe
Masukkan Banyak Kota : 4
Masukkan Jarak Vertex secara Matriks :
0 20 0 5
20 0 15 0
0 15 0 10
5 0 10 0
Kota Tempat Pedagang Berada : 1
Kota Tempat Kastil : 3
Jarak minimal dari kota (Pedagang) 1 ke (Kastil) 3: 15
Apakah ingin melanjutkan program? <y/n>
  
```