

**LAPORAN UAS
STRUKTUR DATA**



**Disusun oleh:
Ummiyatun
21091397039**

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA
PROGRAM VOKASI
UNIVERSITAS NEGERI SURABAYA
2022**

#Part 1

- Kodingan

```
[*] UAS_Part1.cpp UAS_Part2.cpp
1  /*program c++ membuat undirected graph menggunakan representasi adjacency matriks*/
2  #include <bits/stdc++.h>
3  #include <iostream>
4  #define max 20
5
6  using namespace std;
7  int adj_mat [max][max];
8  /*inisialisasi matriks ke nol*/
9  void initialize (int i, int j, int v){
10     for(i = 0; i < v; i++){
11         for(j = 0; j < v; j++){
12             adj_mat[i][j] = 0;
13         }
14     }
15 }
16 /*menambahkan edge*/
17 void add_edge(int u, int v, int w)
18 {
19     adj_mat[u][v] = w;
20     adj_mat[v][u] = w;
21 }
22 /*mencetak matriks*/
23 void display_mat(int v){
24     int i,j;
25     for(i = 1; i < v; i++){
26         for(j = 1; j < v; j++){
27             cout << adj_mat[i][j]<<" ";
28         }
29         cout << endl;
30     }
31 }
32 /*input vertex*/
33 int main()
34 {
35     int v = 5;
36
37     add_edge(1,2,5);
38     add_edge(2,3,1);
39     add_edge(4,1,3);
40     add_edge(2,4,1);
41     add_edge(3,1,1);
42
43     display_mat(v);
44
45     return 0;
```

- Output

```
0 5 1 3
5 0 1 1
1 1 0 0
3 1 0 0

-----
Process exited after 0.6601 seconds with return value 0
Press any key to continue . . .
```

#Part 2

- Kodingan

```
UAS_Part1.cpp  [*] UAS_Part2.cpp
1  #include <iostream>
2  #include <conio.h>
3  #include <string.h>
4  using namespace std;
5  int main()
6  {
7      char kota_A,kota_B,kota_C,kota_D,kota_E;
8      int jumlah,panjang,hasil1,hasil2,hasil3,hasil4,hasil5,hasil6,hasil7;
9      /*input jumlah kota dalam kerajaan Britan*/
10     cout<<"Jumlah kota yang berada dalam kerajaan Britan : "<<endl;
11     cin>>jumlah;
12     /*input nama kota*/
13     cout<<"Kota 1 : ";
14     cin>>kota_A;
15     cout<<"Kota 2 : ";
16     cin>>kota_B;
17     cout<<"Kota 3 : ";
18     cin>>kota_C;
19     cout<<"Kota 4 : ";
20     cin>>kota_D;
21     cout<<"Kota 5 : ";
22     cin>>kota_E;
23     cout<<endl;
24     /*deklarasi graph*/
25     /*menampilkan setiap graph yang terjadi*/
26     cout<<"Tepinya adalah : "<<endl<<endl;
27     cout<<kota_A<<kota_B<<" ";
28     cout<<kota_A<<kota_D<<" ";
29     cout<<kota_A<<kota_E<<" ";
30     cout<<kota_B<<kota_C<<" ";
31     cout<<kota_C<<kota_E<<" ";
32     cout<<kota_C<<kota_D<<" ";
33     cout<<kota_D<<kota_E<<endl<<endl;
34     /*deklarasi edge*/
35     /*menampilkan panjang jalan yang menghubungkan vertex*/
36     cout<<"Panjang jalan antar kota : "<<endl;
37     cout<<"panjang "<<kota_A<<" ke "<<kota_B<<" : ";cin>>hasil1;
38     cout<<"panjang "<<kota_A<<" ke "<<kota_D<<" : ";cin>>hasil2;
39     cout<<"panjang "<<kota_A<<" ke "<<kota_E<<" : ";cin>>hasil3;
40     cout<<"panjang "<<kota_B<<" ke "<<kota_C<<" : ";cin>>hasil4;
41     cout<<"panjang "<<kota_C<<" ke "<<kota_E<<" : ";cin>>hasil5;
42     cout<<"panjang "<<kota_C<<" ke "<<kota_D<<" : ";cin>>hasil6;
43     cout<<"panjang "<<kota_D<<" ke "<<kota_E<<" : ";cin>>hasil7;
44     cout<<endl;
```

```

45      /*deklarasi adjacency*/
46      /*menampilkan jalan yang menghubungkan kedua simpul (x,y,z)*/
47      cout<<"semua jalan dan panjangnya yang ada di kerajaan Britan : "<<endl;
48      cout<<"("<<kota_A<<","<<kota_B<<","<<hasil1<<")";
49      cout<<"("<<kota_A<<","<<kota_D<<","<<hasil2<<")";
50      cout<<"("<<kota_A<<","<<kota_E<<","<<hasil3<<")";
51      cout<<"("<<kota_B<<","<<kota_C<<","<<hasil4<<")";
52      cout<<"("<<kota_C<<","<<kota_E<<","<<hasil5<<")";
53      cout<<"("<<kota_C<<","<<kota_D<<","<<hasil6<<")";
54      cout<<"("<<kota_D<<","<<kota_E<<","<<hasil7<<")";
55      cout<<endl<<endl;
56      /*menampilkan hasil*/
57      /*menampilkan tempat dimana pedagang berada*/
58      cout<<"pedagang sekarang berada di kota : "<<endl<<endl;
59      cout<<kota_A;
60      cout<<endl<<endl;
61      /*menampilkan kota yang diserang naga*/
62      cout<<"kota yang diserang naga adalah : "<<endl<<endl;
63      cout<<kota_C;
64      cout<<endl<<endl;
65      /*menampilkan kota yang terdapat kastil*/
66      cout<<"kota yang memiliki kastil adalah : "<<endl<<endl;
67      cout<<kota_E;
68      cout<<endl<<endl;
69      /*menampilkan jalur tercepat*/
70      cout<<"jalur yang paling cepat ditempuh oleh pedagang untuk ke kastil adalah : "<<endl<<endl;
71      cout<<kota_A<<"->"<<kota_D<<"->"<<kota_E<<endl;
72      cout<<endl<<endl;
73      /*menampilkan total edge yang harus ditempuh*/
74      cout<<"dengan jarak : "<<endl<<endl;
75      cout<<hasil2+hasil7<<endl<<endl;
76
77      getch();
78
79      return 0;
80

```

- **Output**

```

Jumlah kota yang berada dalam kerajaan Britan :
5
Kota 1 : 1
Kota 2 : 2
Kota 3 : 3
Kota 4 : 4
Kota 5 : 5

Tepinya adalah :
12,14,15,23,35,34,45

Panjang jalan antar kota :
panjang 1 ke 2 : 12
panjang 1 ke 4 : 11
panjang 1 ke 5 : 30
panjang 2 ke 3 : 14
panjang 3 ke 5 : 5
panjang 3 ke 4 : 15
panjang 4 ke 5 : 10

Semua jalan dan panjangnya yang ada di kerajaan Britan :
(1,2,12)(1,4,11)(1,5,30)(2,3,14)(3,5,5)(3,4,15)(4,5,10)

Pedagang sekarang berada di kota :
1

Kota yang diserang naga adalah :
3

Kota yang memiliki kastil adalah :
5

Jalur yang paling cepat ditempuh oleh pedagang untuk ke kastil adalah :
1->4->5

dengan jarak :
21

```