

LAPORAN UAS STRUKTUR DATA



Disusun oleh :

Affandika Febrian Putra Yunanto (21091397030)

Kelas B

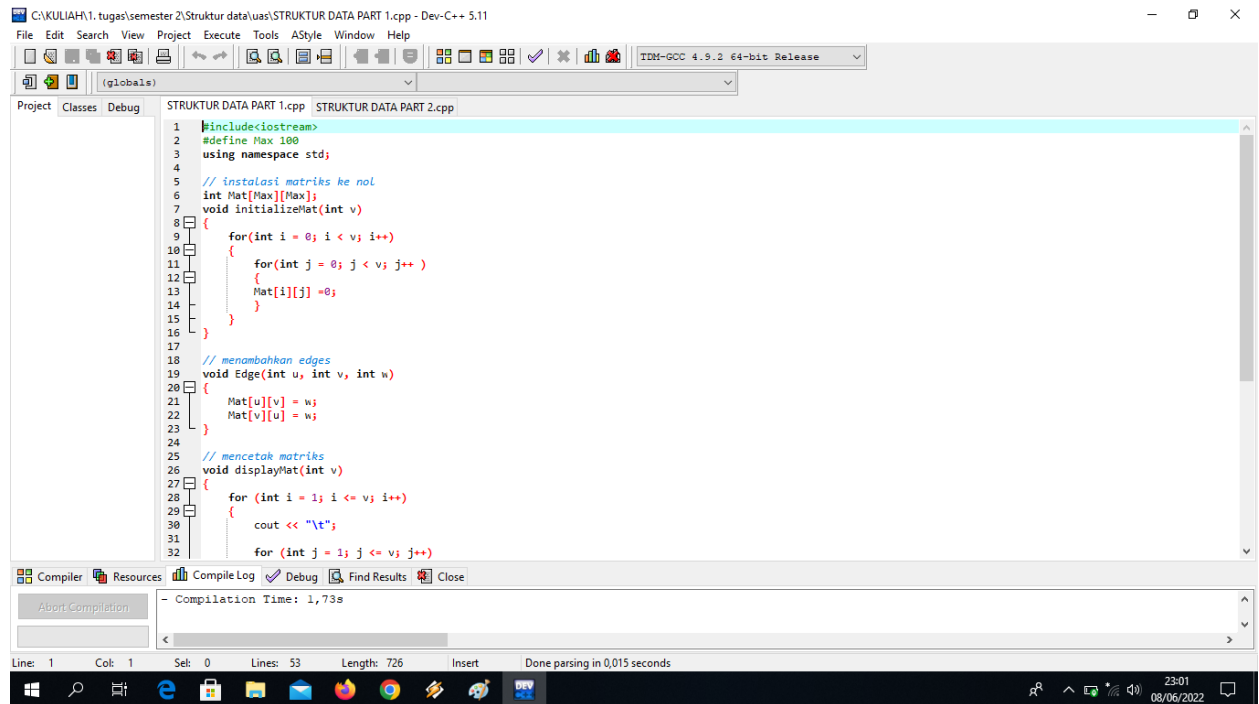
UNIVERSITAS NEGERI SURABAYA

D4 MANAJEMEN INFORMATIKA

2022

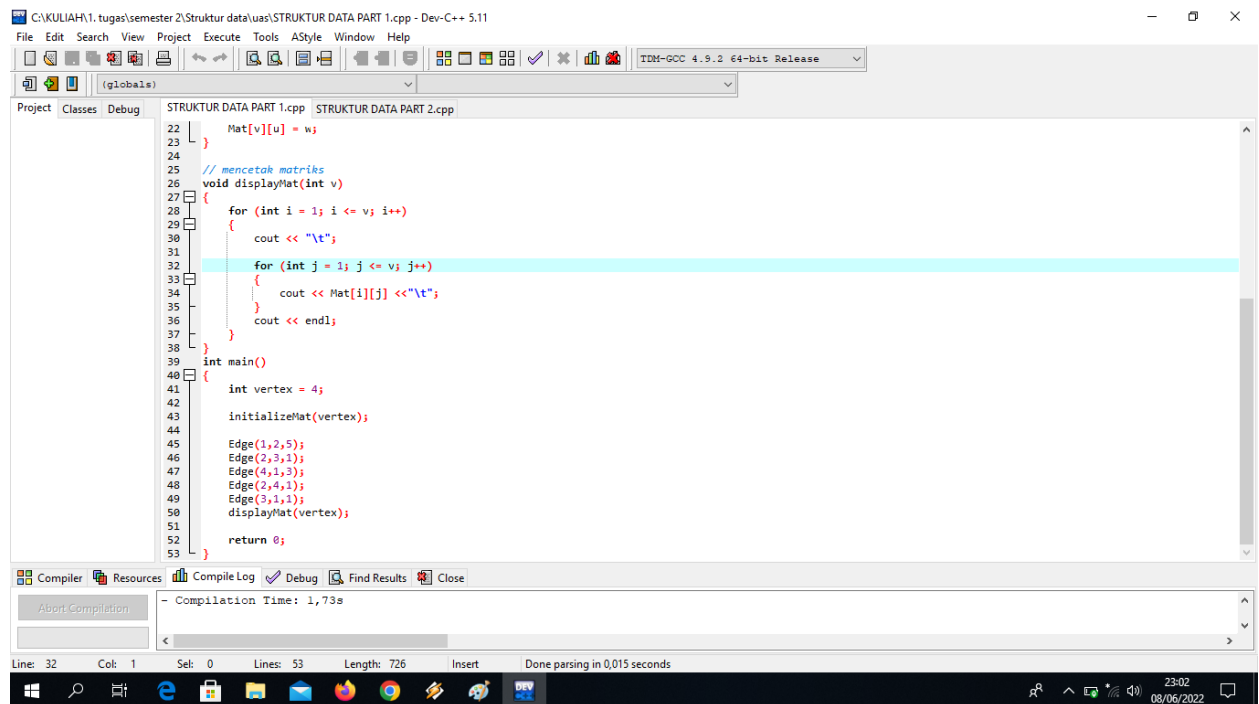
1. Membuat kodingan dan laporan sebuah algoritma yang membuat undirected graph menggunakan representasi adjacency list dengan input vertex dan edge.

Input



The screenshot shows a C++ IDE with the file 'STRUKTUR DATA PART 1.cpp'. The code defines a matrix 'Mat' of size 'Max' (100) and implements functions to initialize it to zero, add edges, and display the matrix. The 'main' function is partially visible at the bottom.

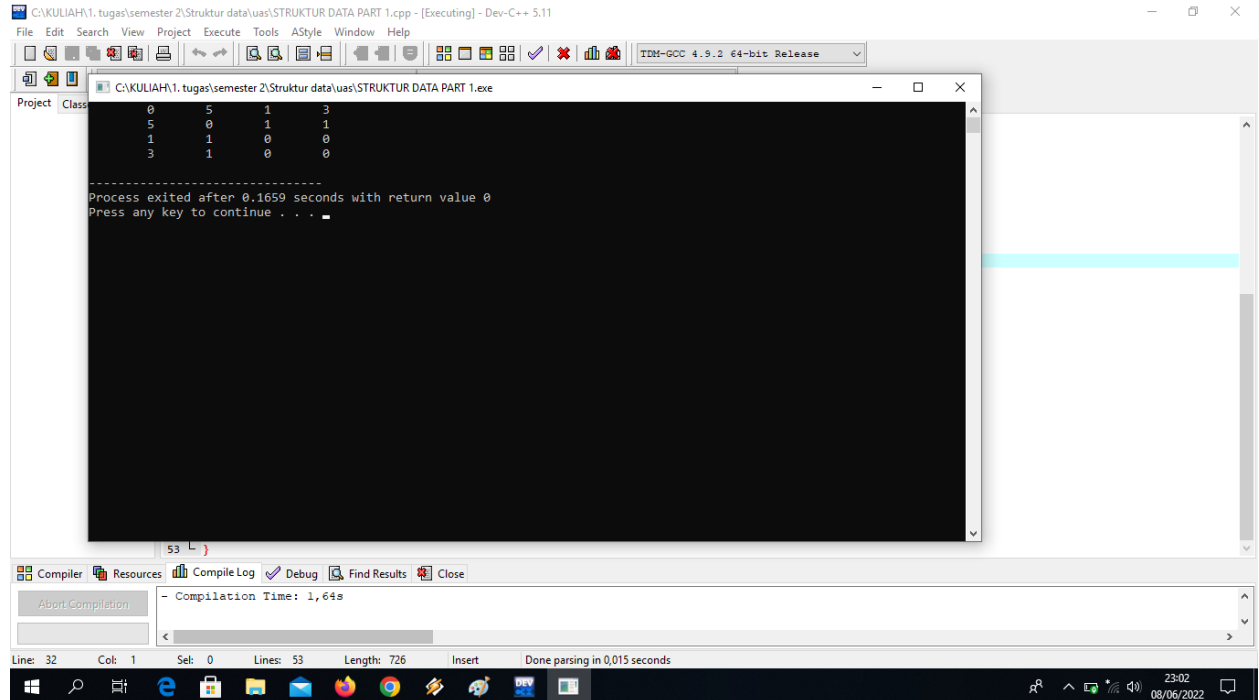
```
1 #include<iostream>
2 #define Max 100
3 using namespace std;
4
5 // instalasi matriks ke nol
6 int Mat[Max][Max];
7 void initializeMat(int v)
8 {
9     for(int i = 0; i < v; i++)
10     {
11         for(int j = 0; j < v; j++)
12         {
13             Mat[i][j] = 0;
14         }
15     }
16 }
17
18 // menambahkan edges
19 void Edge(int u, int v, int w)
20 {
21     Mat[u][v] = w;
22     Mat[v][u] = w;
23 }
24
25 // mencetak matriks
26 void displayMat(int v)
27 {
28     for (int i = 1; i <= v; i++)
29     {
30         cout << "\t";
31         for (int j = 1; j <= v; j++)
```



The screenshot shows the continuation of the C++ program. It includes the 'displayMat' function, the 'main' function which sets up the graph with 4 vertices and 5 edges, and the 'return 0;' statement. The 'displayMat' function is highlighted with a blue selection bar.

```
32         for (int j = 1; j <= v; j++)
33         {
34             cout << Mat[i][j] << "\t";
35         }
36         cout << endl;
37     }
38 }
39
40 int main()
41 {
42     int vertex = 4;
43     initializeMat(vertex);
44
45     Edge(1,2,5);
46     Edge(2,3,1);
47     Edge(4,1,3);
48     Edge(2,4,1);
49     Edge(3,1,1);
50     displayMat(vertex);
51
52     return 0;
53 }
```

Output



```
C:\KULIAH\1. tugas\semester 2\Struktur data\uas\STRUKTUR DATA PART 1.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
C:\KULIAH\1. tugas\semester 2\Struktur data\uas\STRUKTUR DATA PART 1.exe
Project Class
0 5 1 3
5 0 1 1
1 1 0 0
3 1 0 0

-----
Process exited after 0.1659 seconds with return value 0
Press any key to continue . . .
```

Compiler Resources Compile Log Debug Find Results Close

Compilation Time: 1,64s

Line: 32 Col: 1 Sel: 0 Lines: 53 Length: 726 Insert Done parsing in 0,015 seconds

23:02 08/06/2022

2. Terdapat seorang pedagang Rahmad, Rahmad setiap bulan berkeliling di kerajaan Britan untuk berdagang. Tetapi suatu hari, pedagang ini mendapat berita bahwa ada seekor naga yang sedang menyerang salah satu kota. Jadi pedagang ini bergegas menuju ke istana untuk memberitahu raja bahwa ada kota yang sedang diserang sambil menghindari kota yang diserang tersebut. Sehingga raja bisa mengirimkan pasukan untuk menyerang kota tersebut.

Buat kodingan dan laporan cara kerja kodingan tersebut. Jelaskan menggunakan algoritma apa kodingan anda berjalan (dijkstra, A*, bellman ford, dll) dan jelaskan cara kerjanya. Peta kota adalah sebuah undirected, weighted graph. Boleh menggunakan adjacency list atau menggunakan adjacency matrix.

Input

C:\KULIAH\1. tugas\semester 2\Struktur data\uas\STRUKTUR DATA PART 2.cpp - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

(glibc)

STRUKTUR DATA PART 1.cpp STRUKTUR DATA PART 2.cpp

```
1 #include <iostream>
2 #include <conio.h>
3 #include <string.h>
4 using namespace std;
5 int main()
6 {
7     char k1,k2,k3,k4,k5;
8     int jumlah,panjang, hasil1,hasil2,hasil3,hasil4,hasil5,hasil6,hasil7;
9
10    cout<<"* Jumlah kota yang berada di kerajaan Britan : "<< endl;
11    cin>>jumlah;
12
13    // menampilkan vertex
14    // input nama kota
15    cout<<"Kota Pertama : ";
16    cin>>k1;
17    cout<<"Kota Kedua : ";
18    cin>>k2;
19    cout<<"Kota Ketiga : ";
20    cin>>k3;
21    cout<<"Kota Keempat : ";
22    cin>>k4;
23    cout<<"Kota kelima : ";
24    cin>>k5;
25
26    cout<<endl;
27
28    // menampilkan setiap edge yang terjadi
29    cout<<"* Sisi-sisinya adalah : "<<endl<<endl;
30    cout<<k1<<k2<<",";
31    cout<<k1<<k4<<",";
32    cout<<k1<<k5<<",";
```

Compiler Resources Compile Log Debug Find Results Close

Compilation Time: 1,73s

Line: 1 Col: 1 Sel: 0 Lines: 87 Length: 2321 Insert Done parsing in 0,015 seconds

23:04 08/06/2022

C:\KULIAH\1. tugas\semester 2\Struktur data\uas\STRUKTUR DATA PART 2.cpp - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

(glibc)

STRUKTUR DATA PART 1.cpp STRUKTUR DATA PART 2.cpp

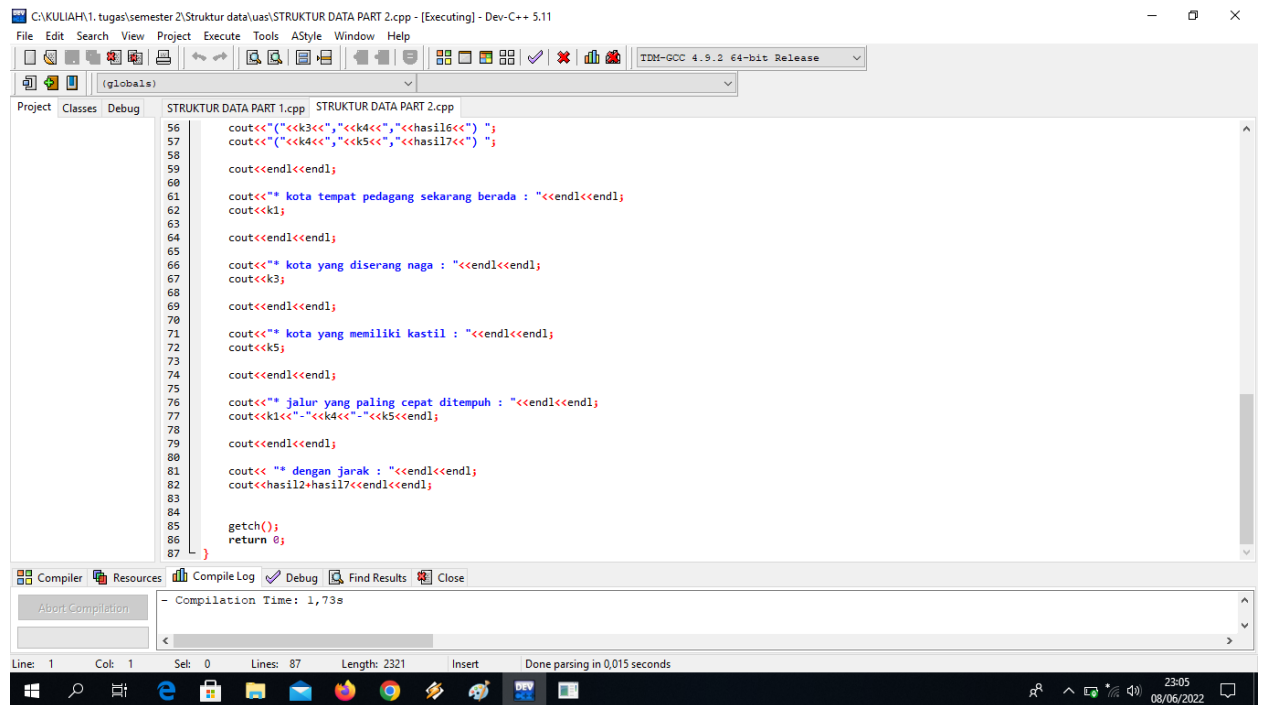
```
28 // menampilkan setiap edge yang terjadi
29 cout<<"* Sisi-sisinya adalah : "<<endl<<endl;
30 cout<<k1<<k2<<",";
31 cout<<k1<<k4<<",";
32 cout<<k1<<k5<<",";
33 cout<<k2<<k3<<",";
34 cout<<k3<<k5<<",";
35 cout<<k3<<k4<<",";
36 cout<<k4<<k5<<endl<<endl;
37
38 // menampilkan panjang jalan yang menghubungkan vertex
39 cout<<"* Panjang jalan antar kota : "<<endl;
40 cout<<"panjang "<<k1<<" ke "<<k2<<" : "; cin>> hasil1;
41 cout<<"panjang "<<k1<<" ke "<<k4<<" : "; cin>> hasil2;
42 cout<<"panjang "<<k1<<" ke "<<k5<<" : "; cin>> hasil3;
43 cout<<"panjang "<<k2<<" ke "<<k3<<" : "; cin>> hasil4;
44 cout<<"panjang "<<k3<<" ke "<<k5<<" : "; cin>> hasil5;
45 cout<<"panjang "<<k3<<" ke "<<k4<<" : "; cin>> hasil6;
46 cout<<"panjang "<<k4<<" ke "<<k5<<" : "; cin>> hasil7;
47
48 cout<<endl;
49
50 cout<<"* seluruh jalan yang ada dalam kerajaan britan dan panjang jalannya : "<< endl;
51 cout<<"("<<k1<<","<<k2<<","<<hasil1<<") ";
52 cout<<"("<<k1<<","<<k4<<","<<hasil2<<") ";
53 cout<<"("<<k1<<","<<k5<<","<<hasil3<<") ";
54 cout<<"("<<k2<<","<<k3<<","<<hasil4<<") ";
55 cout<<"("<<k3<<","<<k5<<","<<hasil5<<") ";
56 cout<<"("<<k3<<","<<k4<<","<<hasil6<<") ";
57 cout<<"("<<k4<<","<<k5<<","<<hasil7<<") ";
58
59 cout<<endl<<endl;
```

Compiler Resources Compile Log Debug Find Results Close

Compilation Time: 1,73s

Line: 1 Col: 1 Sel: 0 Lines: 87 Length: 2321 Insert Done parsing in 0,015 seconds

23:04 08/06/2022



The screenshot shows a C++ IDE with the following components:

- File Explorer:** Displays the project structure with files `STRUKTUR DATA PART 1.cpp` and `STRUKTUR DATA PART 2.cpp`.
- Code Editor:** Contains the following C++ code:

```
56 cout<<" "<<k3<<" "<<k4<<" "<<hasil6<<" ";
57 cout<<" "<<k4<<" "<<k5<<" "<<hasil7<<" ";
58
59 cout<<endl<<endl;
60
61 cout<<"* kota tempat pedagang sekarang berada : "<<endl<<endl;
62 cout<<k1;
63
64 cout<<endl<<endl;
65
66 cout<<"* kota yang diserang naga : "<<endl<<endl;
67 cout<<k3;
68
69 cout<<endl<<endl;
70
71 cout<<"* kota yang memiliki kastil : "<<endl<<endl;
72 cout<<k5;
73
74 cout<<endl<<endl;
75
76 cout<<"* jalur yang paling cepat ditempuh : "<<endl<<endl;
77 cout<<k1<<"-"<<k4<<"-"<<k5<<endl;
78
79 cout<<endl<<endl;
80
81 cout<<"* dengan jarak : "<<endl<<endl;
82 cout<<hasil2+hasil7<<endl<<endl;
83
84
85 getch();
86 return 0;
87 }
```
- Compiler Output:** Shows the compilation time as 1.73s.
- Status Bar:** Indicates the current line (1), column (1), and other file statistics.

Output

