

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL III
ABSTRACT DATA TYPE (ADT)**



Disusun Oleh :

NAMA : DEVI YULIANA

NIM : 103112400151

Dosen

FAHRUDIN MUKTI WIBOWO

STRUKTUR DATA

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

A. Dasar Teori

1. Abstract Data Type (ADT) adalah konsep dalam pemrograman yang digunakan untuk mendefinisikan tipe data beserta operasi dasarnya. ADT tidak hanya berisi data, tetapi juga fungsi-fungsi (prosedur) yang dapat digunakan untuk mengelola data tersebut.
2. Tujuan utama dari ADT adalah untuk membuat program menjadi lebih terstruktur, efisien, dan mudah dikelola. Dengan ADT, data dan fungsinya dapat digabung dalam satu kesatuan sehingga pengguna tidak perlu mengetahui detail implementasinya.
3. Dalam bahasa C++, ADT biasanya diimplementasikan menggunakan struct untuk mendefinisikan tipe data, dan beberapa fungsi sebagai operasi dasarnya, seperti:
 - a) Konstruktor (Creator): membentuk atau menginisialisasi data.
 - b) Selector (Accessor): mengambil nilai dari suatu data.
 - c) Mutator (Modifier): mengubah nilai suatu data.
 - d) Destructor: menghapus atau membersihkan data dari memori.
4. Struktur program ADT umumnya dibagi menjadi beberapa file, yaitu:
 - a) File header (.h) berisi definisi tipe data dan deklarasi fungsi.
 - b) File implementasi (.cpp) berisi isi (logika) dari fungsi.
 - c) File utama (main.cpp) sebagai driver untuk menjalankan program.
5. Pada praktikum ini, penerapan ADT dilakukan dalam bentuk program yang mengelola data mahasiswa, data pelajaran, serta manipulasi array dua dimensi (2D) dan pointer, untuk memperkuat pemahaman konsep modularisasi dan abstraksi data.

B. Guided (berisi screenshot source code & output program disertai penjelasannya)

Guided 1

mahasiswa.cpp

```

mahasiswa.cpp latihan\mahasiswa.cpp\...
1  #include "mahasiswa.h"
2  #include <iostream>
3  using namespace std;
4
5
6  void inputMhs(mahasiswa &m)
7  {
8      cout << "input nama = ";
9      cin >> (m).nim;
10     cout << "input nilai 1 = ";
11     cin >> (m).nilai1;
12     cout << "input nilai 2 = ";
13     cin >> (m).nilai2;
14 }
15
16 float rata2(mahasiswa m)
17 {
18     return float(m.nilai1+m.nilai2)/2;
19 }

```

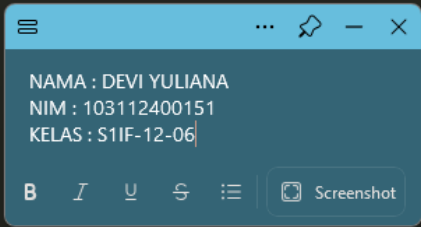
NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

B I U S :≡ Screenshot

mahasiswa.h

```
h mahasiswa.h latihan\mahasiswa.h\...

1  #ifndef MAHASISWA_H_INCLUDED
2  #define MAHASISWA_H_INCLUDED
3  struct mahasiswa
4  {
5      char nim[10];
6      int nilai1, nilai2;
7  };
8  void inputMhs(mahasiswa &m);
9  float rata2(mahasiswa m);
10 #endif
```

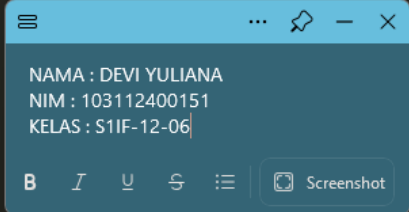


NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

main.cpp

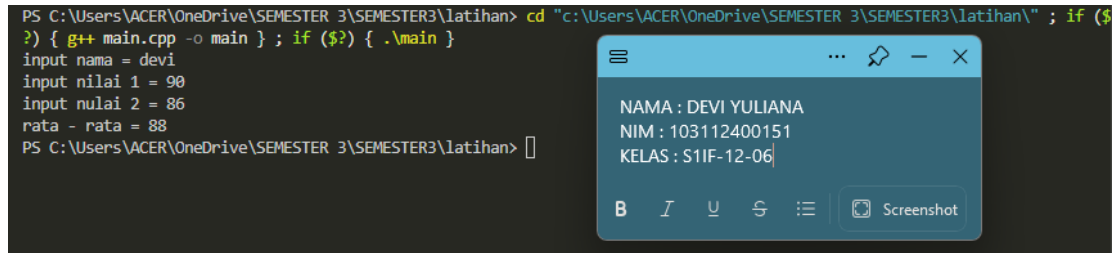
```
C++ main.cpp latihan\main.cpp\...

1  #include <iostream>
2  #include "mahasiswa.h"
3  #include "mahasiswa.cpp"
4
5  using namespace std;
6
7  int main()
8  {
9      mahasiswa mhs;
10     inputMhs(mhs);
11     cout << "rata - rata = " << rata2(mhs);
12     return 0;
13 }
```



NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

Screenshots Output



```
PS C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\latihan> cd "c:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\latihan\" ; if ($?) { g++ main.cpp -o main } ; if ($?) { .\main }
input nama = devi
input nilai 1 = 90
input nilai 2 = 86
rata - rata = 88
PS C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\latihan>
```

NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S11IF-12-06

Deskripsi :

Di dalamnya, tipe data mahasiswa dibuat dalam bentuk struct yang berisi beberapa field, yaitu nim, nilai1, dan nilai2. Program ini dibagi menjadi tiga bagian file utama: main.cpp, mahasiswa.h, dan mahasiswa.cpp. Tujuannya adalah agar struktur program lebih rapi, mudah dibaca, dan sesuai dengan prinsip modularitas dalam pemrograman.

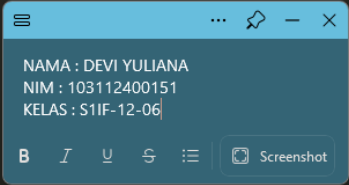
Pada file mahasiswa.h, didefinisikan struktur mahasiswa beserta deklarasi dua buah fungsi, yaitu inputMhs() dan rata2(). Fungsi inputMhs() digunakan untuk mengisi data mahasiswa, sedangkan fungsi rata2() digunakan untuk menghitung rata-rata dari dua nilai yang telah dimasukkan. File ini hanya berisi definisi tipe dan deklarasi fungsi tanpa implementasi logikanya.

Kemudian, di file mahasiswa.cpp, berisi implementasi dari fungsi-fungsi yang telah dideklarasikan sebelumnya. Pada bagian inputMhs(), program meminta pengguna untuk memasukkan NIM serta dua nilai mahasiswa. Setelah itu, fungsi rata2() akan menghitung rata-rata dari dua nilai tersebut dengan rumus $(\text{nilai1} + \text{nilai2}) / 2$. Hasilnya dikonversi ke tipe float supaya hasil pembagian bisa menampilkan angka desimal.

C. Unguided/Tugas (berisi screenshot source code & output program disertai penjelasannya)

Unguided 1

```
g++ mahasiswa.cpp LAPRAK WEEK 3 NO 1\mahasiswa.cpp\...
1  #include <iostream>
2  #include "mahasiswa.h"
3  using namespace std;
4
5  void inputMhs(Mahasiswa &m) {
6      cout << "Nama : ";
7      cin >> m.nama;
8      cout << "NIM : ";
9      cin >> m.nim;
10     cout << "Nilai UTS : ";
11     cin >> m.uts;
12     cout << "Nilai UAS : ";
13     cin >> m.uas;
14     cout << "Nilai Tugas : ";
15     cin >> m.tugas;
16     m.nilaiAkhir = hitungNilaiAkhir(m);
17 }
18
19 float hitungNilaiAkhir(Mahasiswa m) {
20     return (0.3 * m.uts) + (0.4 * m.uas) + (0.3 * m.tugas);
21 }
22
23 void tampilMhs(Mahasiswa m) {
24     cout << m.nama << "\t" << m.nim << "\t"
25          << m.uts << "\t" << m.uas << "\t"
26          << m.tugas << "\t" << m.nilaiAkhir << endl;
27 }
```



```
h mahasiswa.h LAPRAK WEEK 3 NO 1\mahasiswa.h\...
1  #ifndef MAHASISWA_H_INCLUDED
2  #define MAHASISWA_H_INCLUDED
3
4  #include <string>
5  using namespace std;
6
7  struct Mahasiswa {
8      string nama;
9      string nim;
10     float uts;
11     float uas;
12     float tugas;
13     float nilaiAkhir;
14 };
15
16 void inputMhs(Mahasiswa &m);
17 float hitungNilaiAkhir(Mahasiswa m);
18 void tampilMhs(Mahasiswa m);
19
20 #endif
```



```
main.cpp LAPRAK WEEK 3 NO 1\main.cpp\...
1 #include <iostream>
2 #include "mahasiswa.h"
3 using namespace std;
4
5 int main() {
6     const int maks = 10;
7     Mahasiswa mhs[maks];
8     int jumlah;
9
10    cout << "Masukkan jumlah mahasiswa (maks 10): ";
11    cin >> jumlah;
12
13    if (jumlah > maks) {
14        cout << "Jumlah melebihi batas!\n";
15        return 0;
16    }
17
18    for (int i = 0; i < jumlah; i++) {
19        cout << "\nData Mahasiswa ke-" << i + 1 << endl;
20        inputMhs(mhs[i]);
21    }
22
23    cout << "\n=====\\n";
24    cout << "No\\tNama\\tNIM\\tUTS\\tUAS\\tTugas\\tNilai Akhir\\n";
25    cout << "-----\\n";
26
27    for (int i = 0; i < jumlah; i++) {
28        cout << i + 1 << "\\t";
29        tampilMhs(mhs[i]);
30    }
31
32    return 0;
33 }
```

Screenshots Output

```
PS C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\LAPRAK WEEK 3 NO 1> cd "C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\LAPRAK WEEK 3 NO 1" ; if ($?) { g++ main.cpp -o main } ; if ($?) { .\main }
Masukkan jumlah mahasiswa (maks 10): 5

Data Mahasiswa ke-1
Nama : devi
NIM : 101
Nilai UTS : 98
Nilai UAS : 96
Nilai Tugas : 88

Data Mahasiswa ke-2
Nama : lia
NIM : 102
Nilai UTS : 89
Nilai UAS : 97
Nilai Tugas : 87

Data Mahasiswa ke-3
Nama : ana
NIM : 103
Nilai UTS : 86
Nilai UAS : 85
Nilai Tugas : 83

Data Mahasiswa ke-4
Nama : lala
NIM : 104
Nilai UTS : 87
Nilai UAS : 83

Data Mahasiswa ke-5
Nama : ani
NIM : 105
Nilai UTS : 82
Nilai UAS : 88
Nilai Tugas : 90

=====\\n
No\\tNama\\tNIM\\tUTS\\tUAS\\tTugas\\tNilai Akhir
-----\\n
1\\tdevi\\t101\\t98\\t96\\t88\\t84.2
2\\tlia\\t102\\t89\\t97\\t87\\t91.6
3\\tana\\t103\\t86\\t85\\t83\\t84.7
4\\tlala\\t104\\t87\\t83\\t89\\t86
5\\tani\\t105\\t82\\t88\\t90\\t86.8
```

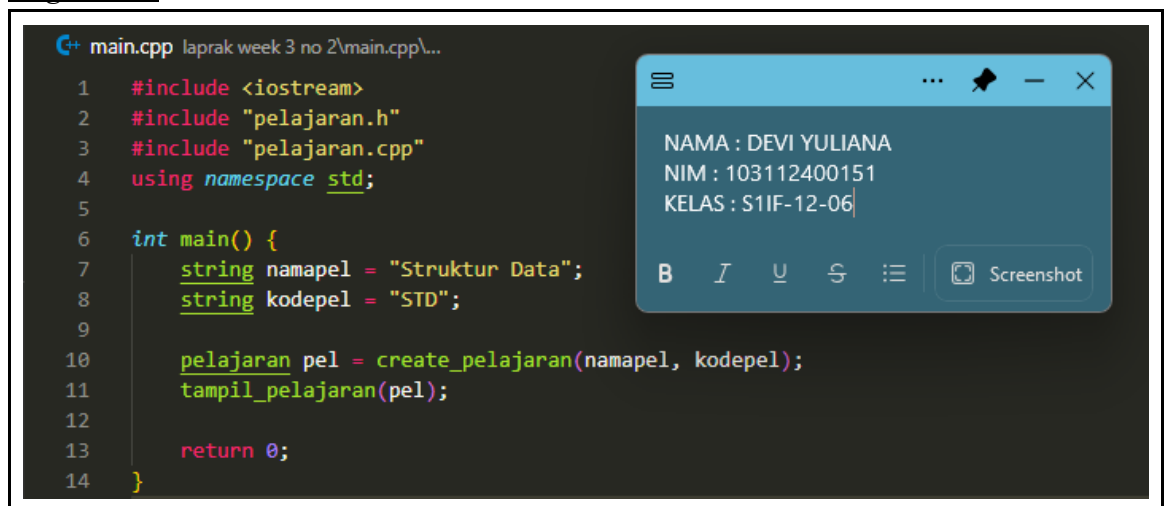
Deskripsi:

Di program ini, kodenya dibagi jadi tiga file biar lebih rapi dan gampang dipahami. Ada mahasiswa.h buat nyimpen struktur data dan deklarasi fungsi, mahasiswa.cpp buat isi atau logika dari fungsi-fungsinya, dan main.cpp yang jadi program utamanya.

Di file mahasiswa.cpp, ada beberapa fungsi penting. Fungsi inputMhs() dipakai buat masukin data mahasiswa seperti nama, NIM, nilai UTS, UAS, dan nilai tugas. Setelah itu, fungsi hitungNilaiAkhir() ngitung nilai akhirnya pakai rumus $(0.3 * \text{UTS}) + (0.4 * \text{UAS}) + (0.3 * \text{Tugas})$. Hasilnya disimpan di variabel nilaiAkhir. Terus ada juga fungsi tampilMhs() yang gunanya buat nampilin semua data mahasiswa ke layar dalam bentuk tabel biar rapi.

Bagian main.cpp jadi pusat jalannya program. Di situ, pengguna bakal diminta buat nentuin berapa jumlah mahasiswa yang mau dimasukin (maksimal 10 orang). Setelah data semuanya diinput, program bakal nampilin hasilnya berupa daftar mahasiswa lengkap dengan nilai-nilainya dan nilai akhir yang udah dihitung otomatis.

Unguided 2



The screenshot shows a C++ IDE with a dark theme. The main.cpp file is open, displaying the following code:

```
1 #include <iostream>
2 #include "pelajaran.h"
3 #include "pelajaran.cpp"
4 using namespace std;
5
6 int main() {
7     string namapel = "Struktur Data";
8     string kodepel = "STD";
9
10    pelajaran pel = create_pelajaran(namapel, kodepel);
11    tampil_pelajaran(pel);
12
13    return 0;
14 }
```

On the right side, there is a terminal window with a light blue header. It displays the output of the program:

```
NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06
```

Below the output, there are formatting icons (B, I, U, S, L) and a 'Screenshot' button.


```
pelajaran.cpp laprak week 3 no 2\pelajaran.cpp\...
1  #include <iostream>
2  #include "pelajaran.h"
3  using namespace std;
4
5  pelajaran create_pelajaran(string namapel, string kodepel) {
6      pelajaran p;
7      p.namaMapel = namapel;
8      p.kodeMapel = kodepel;
9      return p;
10 }
11
12 void tampil_pelajaran(pelajaran pel) {
13     cout << "nama pelajaran : " << pel.namaMapel << endl;
14     cout << "nilai : " << pel.kodeMapel << endl;
15 }
```

NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

B I U S ≡ Screenshot

```
h pelajaran.h laprak week 3 no 2\pelajaran.h\...
1  #ifndef PELAJARAN_H_INCLUDED
2  #define PELAJARAN_H_INCLUDED
3
4  #include <string>
5  using namespace std;
6
7  struct pelajaran {
8      string namaMapel;
9      string kodeMapel;
10 };
11
12 pelajaran create_pelajaran(string namapel, string kodepel);
13 void tampil_pelajaran(pelajaran pel);
14
15 #endif
```

NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

Screenshots Output :

```
PS C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\LAPRAK WEEK 3 NO 2> cd "c:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\LAPRAK WEEK 3 NO 2\" ; if ($?) { g++ main.cpp -o main } ; if ($?) { .\main }
nama pelajaran : Struktur Data
nilai : STD
PS C:\Users\ACER\OneDrive\SEMESTER 3\SEMESTER3\LAPRAK WEEK 3 NO 2> |
```

NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

Deskripsi:

Unguided 3

```
main.cpp LAPRAK WEEK 3 NO 3\main.cpp\...
1  #include <iostream>
2  using namespace std;
3
4  void tampil(int arr[3][3]) {
5      for (int i = 0; i < 3; i++, cout << endl)
6          for (int j = 0; j < 3; j++)
7              cout << arr[i][j] << " ";
8  }
9
10 void tukarArray(int a[3][3], int b[3][3], int x, int y) {
11     int t = a[x][y];
12     a[x][y] = b[x][y];
13     b[x][y] = t;
14 }
15
16 void tukarPointer(int *p1, int *p2) {
17     int t = *p1;
18     *p1 = *p2;
19     *p2 = t;
20 }
21
22 int main() {
23     int A[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
24     int B[3][3] = {{9,8,7},{6,5,4},{3,2,1}};
25     int *ptr1, *ptr2, x = 10, y = 20;
26
27     cout << "Array A sebelum ditukar:\n"; tampil(A);
28     cout << "\n\nArray B sebelum ditukar:\n"; tampil(B);
29
30     tukarArray(A, B, 1, 1);
31
32     cout << "\n\nSetelah pertukaran elemen (1,1):\n";
33     cout << "Array A:\n"; tampil(A);
34     cout << "\n\nArray B:\n"; tampil(B);
35
36     ptr1 = &x; ptr2 = &y;
37     cout << "\n\nNilai sebelum ditukar: x=" << x << ", y=" << y;
38     tukarPointer(ptr1, ptr2);
39     cout << "\n\nNilai setelah ditukar: x=" << x << ", y=" << y << endl;
40
41     return 0;
42 }
```

NAMA : DEVI YULIANA
NIM : 103112400151
KELAS : S1IF-12-06

Screenshots Output

E. Referensi

Anita Sindar, R. M. S. (2019). *Struktur Data Dan Algoritma Dengan C++ (Vol. 1)*. CV. AA. RIZKY.