

**LAPORAN PRAKTIKUM  
STRUKTUR DATA**

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



**Disusun Oleh :**

NAMA : Damanik, Yohanes Geovan Ondova  
NIM : 103112400022

**Dosen**

FAHRUDIN MUKTI WIBOWO

**PROGRAM STUDI STRUKTUR DATA  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2025**

## A. Dasar Teori

- ADT adalah tipe data dan kumpulan operasi dasar (primitif) yang terkait dengannya, termasuk definisi invarian dan aksioma
- Primitif dalam ADT dikelompokkan menjadi:
  - a) Konstruktor/Kreator: Membentuk nilai tipe.
  - b) Selector: Mengakses tipe komponen.
  - c) Prosedur pengubah nilai komponen.
  - d) Tipe validator komponen: Menguji apakah dapat membentuk tipe sesuai batasan.
  - e) Destruktor: Menghancurkan nilai objek.
  - f) Baca: Antarmuka dengan perangkat I/O.
  - g) Operator Relasional: Mendefinisikan perbandingan.
  - h) Aritmatika: Operator aritmatika pada tipe tersebut.
  - i) Konversi: Dari tipe tersebut ke tipe dan sebaliknya.

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

### Guided 1

#### Mahasiswa.cpp

```
#include "mahasiswa.h"
#include <iostream>
using namespace std;

void inputMhs(mahasiswa &m)
{
    cout << "Input nama = ";
    cin >> (m).nim;
    cout << "Input nilai = ";
    cin >> (m).nilai1;
    cout << "Input nilai = ";
    cin >> (m).nilai2;
}

float rata2(mahasiswa m)
{
    return float(m.nilai1 + m.nilai2) / 2;
}
```

## Mahasiswa.h

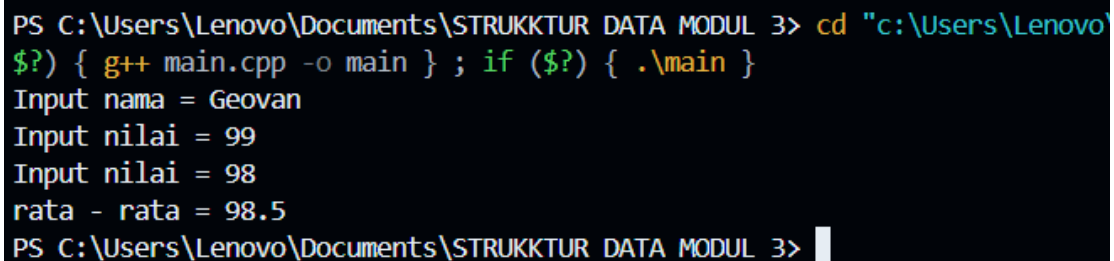
```
#ifndef MAHASISWA_H_INCLUDED
#define MAHASISWA_H_INCLUDED
struct mahasiswa
{
    char nim[10];
    int nilai1, nilai2;
};
void inputMhs(mahasiswa &m);
float rata2(mahasiswa m);
#endif
```

## Main.cpp

```
#include <iostream>
#include "mahasiswa.h"
#include "mahasiswa.cpp"
using namespace std;

int main()
{
    mahasiswa mhs;
    inputMhs(mhs);
    cout << "rata - rata = " << rata2(mhs);
    return 0;
}
```

## Output



```
PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3> cd "c:\Users\Lenovo\'
$?) { g++ main.cpp -o main } ; if ($?) { .\main }
Input nama = Geovan
Input nilai = 99
Input nilai = 98
rata - rata = 98.5
PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3> |
```

## Deskripsi:

Program ini digunakan untuk mencari rata - rata. Program akan meminta input nama dan 2 nilai dari user, lalu program akan mengeksekusinya.

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

Unguided 1

```
#include <iostream>
using namespace std;

struct Mahasiswa {
    string nama;
    string nim;
    float uts;
    float uas;
    float tugas;
    float nilaiAkhir;
};

float hitungNilaiAkhir(float uts, float uas, float tugas) {
    return 0.3 * uts + 0.4 * uas + 0.3 * tugas;
}

int main() {
    Mahasiswa mhs[10];
    int n;

    cout << "Masukkan jumlah mahasiswa (maks 10): ";
    cin >> n;

    for(int i = 0; i < n; i++){
        cout << "\nData Mahasiswa ke-" << i+1 << endl;
        cout << "Nama   : ";
        cin >> mhs[i].nama;
        cout << "NIM    : ";
        cin >> mhs[i].nim;
        cout << "Nilai UTS : ";
        cin >> mhs[i].uts;
        cout << "Nilai UAS : ";
        cin >> mhs[i].uas;
        cout << "Nilai Tugas: ";
        cin >> mhs[i].tugas;

        mhs[i].nilaiAkhir = hitungNilaiAkhir(mhs[i].uts, mhs[i].uas, mhs[i].tugas);
    }

    cout << "\n=== Data Nilai Mahasiswa ===" << endl;
    for(int i = 0; i < n; i++){
        cout << "\nMahasiswa ke-" << i+1 << endl;
        cout << "Nama       : " << mhs[i].nama << endl;
        cout << "NIM        : " << mhs[i].nim << endl;
        cout << "Nilai Akhir : " << mhs[i].nilaiAkhir << endl;
    }
}
```

```
    return 0;
}
```

## Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3\Unguided1> cd "c:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3\Unguided1\" ; if ($?) { g++ unguided1.cpp -o unguided1 } ; if ($?) { .\unguided1 }
Masukkan jumlah mahasiswa (maks 10): 3

Data Mahasiswa ke-1
Nama      : Geovan
NIM       : 99
Nilai UTS : 99
Nilai UAS : 99
Nilai Tugas: 99

Data Mahasiswa ke-2
Nama      : Yosep
NIM       : 56
Nilai UTS : 67
Nilai UAS : 77
Nilai Tugas: 66

Data Mahasiswa ke-3
Nama      : Maul
NIM       : 88
Nilai UTS : 78
Nilai UAS : 68
Nilai Tugas: 77

=== Data Nilai Mahasiswa ===

Mahasiswa ke-1
Nama      : Geovan
NIM       : 99
Nilai Akhir : 99

Mahasiswa ke-2
Nama      : Yosep
NIM       : 56
Nilai Akhir : 70.7

Mahasiswa ke-3
Nama      : Maul
NIM       : 88
Nilai Akhir : 73.7
```

## Deskripsi:

Program ini merupakan sistem input dan pengolahan data nilai mahasiswa yang menggunakan struktur data (struct) untuk merepresentasikan setiap mahasiswa dengan atribut nama, NIM, nilai UTS, UAS, tugas, dan nilai akhir. Program meminta pengguna memasukkan jumlah mahasiswa (maksimal 10), kemudian menginput data masing-masing mahasiswa beserta nilainya, dan menghitung nilai akhir menggunakan fungsi `hitungNilaiAkhir()` dengan bobot 30% UTS, 40% UAS, dan 30% tugas. Setelah semua data diinput, program menampilkan seluruh data mahasiswa beserta nilai akhir yang telah dihitung. Program ini mendemonstrasikan penggunaan struct untuk mengelompokkan data terkait, fungsi dengan parameter, serta array of struct untuk mengelola kumpulan data mahasiswa secara terstruktur..

## Unguided 2

### Pelajaran.cpp

```
#include "pelajaran.h"
#include <iostream>
using namespace std;

pelajaran create_pelajaran(string nama, string kode) {
    pelajaran p;
    p.namaMapel = nama;
    p.kodeMapel = kode;
    return p;
}

void tampil_pelajaran(pelajaran p) {
    cout << "nama pelajaran : " << p.namaMapel << endl;
    cout << "nilai : " << p.kodeMapel << endl;
}
```

### Pelajaran.h

```
#ifndef PELAJARAN_H_INCLUDED
#define PELAJARAN_H_INCLUDED

#include <string>
using namespace std;

struct pelajaran {
    string namaMapel;
    string kodeMapel;
};

pelajaran create_pelajaran(string nama, string kode);
void tampil_pelajaran(pelajaran p);

#endif
```

## Main.cpp

```
#include <iostream>
#include "pelajaran.h"
#include "pelajaran.cpp"
using namespace std;

int main() {
    string namapel = "Struktur Data";
    string kodepel = "STD";

    pelajaran pel = create_pelajaran(namapel, kodepel);
    tampil_pelajaran(pel);

    return 0;
}
```

## Output

```
PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3> cd "c:\Users\Lenovo
$?) { g++ mainn.cpp -o mainn } ; if ($?) { .\mainn }
nama pelajaran : Struktur Data
nilai : STD
PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3> █
```

## Deskripsi:

Program ini merupakan implementasi sederhana dalam bahasa C++ untuk mengelola data pelajaran menggunakan struktur data. Program terdiri dari tiga file utama: file header (pelajaran.h) yang mendefinisikan struktur pelajaran berisi nama dan kode mata pelajaran beserta deklarasi fungsinya, file implementasi (pelajaran.cpp) yang berisi definisi fungsi `create_pelajaran` untuk membuat objek pelajaran dan `tampil_pelajaran` untuk menampilkan datanya, serta file main (mainn.cpp) yang menjadi driver program dengan membuat objek pelajaran "Struktur Data" dengan kode "STD" kemudian menampilkan informasinya.

### Unguided 3

```
#include <iostream>
using namespace std;

void tampilArray(int arr[3][3]) {
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
}

void tukarPosisi(int arr1[3][3], int arr2[3][3], int baris, int kolom) {
    int temp;
    temp = arr1[baris][kolom];
    arr1[baris][kolom] = arr2[baris][kolom];
    arr2[baris][kolom] = temp;
}

void tukarPointer(int *p1, int *p2) {
    int temp;
    temp = *p1;
    *p1 = *p2;
    *p2 = temp;
}

int main() {
    int A[3][3] = { {1,2,3}, {4,5,6}, {7,8,9} };
    int B[3][3] = { {9,8,7}, {6,5,4}, {3,2,1} };
    int *ptr1, *ptr2;
    int x = 10, y = 20;

    cout << "Array A:" << endl;
    tampilArray(A);
    cout << "\nArray B:" << endl;
    tampilArray(B);

    cout << "\nMenukar posisi elemen (baris=1, kolom=2)" << endl;
    tukarPosisi(A, B, 1, 2); // posisi baris ke-1 kolom ke-2

    cout << "\nArray A setelah ditukar:" << endl;
    tampilArray(A);
    cout << "\nArray B setelah ditukar:" << endl;
    tampilArray(B);

    // pointer
    ptr1 = &x;
    ptr2 = &y;
```



```

    cout << "\nSebelum tukar pointer:" << endl;
    cout << "x = " << x << ", y = " << y << endl;

    tukarPointer(ptr1, ptr2);

    cout << "Setelah tukar pointer:" << endl;
    cout << "x = " << x << ", y = " << y << endl;

    return 0;
}

```

## Output

```

PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3\Unguided1> cd "c:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3\Unguided3\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Array A:
1 2 3
4 5 6
7 8 9

Array B:
9 8 7
6 5 4
3 2 1

Menukar posisi elemen (baris=1, kolom=2)

Array A setelah ditukar:
1 2 3
4 5 4
7 8 9

Array B setelah ditukar:
9 8 7
6 5 6
3 2 1

Sebelum tukar pointer:
x = 10, y = 20
Setelah tukar pointer:
x = 20, y = 10
PS C:\Users\Lenovo\Documents\STRUKTUR DATA MODUL 3\Unguided3>

```

## Deskripsi:

Program ini mendemonstrasikan dua teknik pertukaran data dalam C++ dengan menggunakan array dua dimensi dan pointer, di mana fungsi tukarPosisi menukar elemen

tertentu antara dua array berdasarkan indeks baris dan kolom yang ditentukan, sementara fungsi `tukarPointer` memanfaatkan pointer untuk menukar nilai dua variabel secara langsung melalui alamat memorinya, menunjukkan bagaimana manipulasi data dapat dilakukan baik melalui operasi array maupun akses memori langsung dengan pointer.

#### D. Kesimpulan

#### E. Referensi