

LAPORAN PRAKTIKUM
Modul 3
Abstrak Data Type



Disusun Oleh :
Fauzan Rofif Ardiyanto/2211104036
SE0602

Asisten Praktikum : Aldi
Putra
Andini Nur Hidayah

Dosen Pengampu :
Wahyu Andi Saputra

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK FAKULTAS
INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO 2024

Tujuan

1. Memahami konsep Abstract Data Typee (ADT) dan penggunaannya dalam pemrograman.

Landasan Teori

Abstact Daya Type (ADT)

ADT adalah TYPE dan sekumpulan PRIMITIF (operasi dasar) terhadap TYPE tersebut. Selain itu, dalam sebuah ADT yang lengkap, disertakan pula definisi invarian dari TYPE dan aksioma yang berlaku. ADT merupakan definisi STATIK.

Definisi type dari sebuah ADT dapat mengandung sebuah definisi ADT lain. Misalnya :

3. ADT waktu yang terdiri dari ADT JAM dan ADT DATE
4. Garis terdiri dari duah buah ADT POINT

Guided

1. Input :

```

#include <iostream>
#include <conio.h>
#include <stdlib.h>

using namespace std;

struct mahasiswa{
    char nim[10];
    int nilai1, nilai2;
};

void inputMhs(mahasiswa &m);
float rata2(mahasiswa m);

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

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

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

```

Output :

```
NOVO\Documents\ITTP\TUGAS SEMESTER 5 ITTP\Praktikum STD\STD_F  
auzanRofifArdiyanto_2211104036\03_STD\Guided\output'  
PS C:\Users\LENOVO\Documents\ITTP\TUGAS SEMESTER 5 ITTP\Prakt  
ikum STD\STD_FauzanRofifArdiyanto_2211104036\03_STD\Guided\ou  
tput> & .\'guided1.exe'  
input nim= 2211104036  
Input nilai= 100  
Input nilai= 80  
rata-rata = 90
```

Unguided

1. Buatlah program yang dapat menyimpan data mahasiswa (max 10) kedalam sebuah array dengan field nama, nim uts,uas,tugas, dan nilai akhir. Nilai akhir diperoleh dari FUNGSI dengan rumus $0.3*uts+0.4*uas+0.3*tugas$
Input:

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

const int MAX_STUDENTS = 10;

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

// Function to calculate nilai_akhir
float hitungNilaiAkhir(float uts, float uas, float tugas) {
    return 0.3 * uts + 0.4 * uas + 0.3 * tugas;
}

int main() {
    Mahasiswa mahasiswa[MAX_STUDENTS];
    int jumlah;

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

    if (jumlah > MAX_STUDENTS) {
        cout << "Jumlah mahasiswa tidak boleh lebih dari " << MAX_STUDENTS;
        return 1;
    }

    // Input data for each student
    for (int i = 0; i < jumlah; i++) {
        cout << "\nMasukkan data untuk mahasiswa ke-" << i + 1 << ": ";
        cout << "Nama: ";
        cin.ignore(); // Clear input buffer
        getline(cin, mahasiswa[i].nama);
        cout << "NIM: ";
        getline(cin, mahasiswa[i].nim);
        cout << "Nilai UTS: ";
        cin >> mahasiswa[i].uts;
        cout << "Nilai UAS: ";
```

```

        for (int i = 0; i < jumlah; i++) {
            cout << "Nilai UAS: ";
            cin >> mahasiswa[i].uas;
            cout << "Nilai Tugas: ";
            cin >> mahasiswa[i].tugas;

            // Calculate nilai_akhir
            mahasiswa[i].nilai_akhir = hitungNilaiAkhir(mahasiswa[i].

        }

        // Output data
        cout << "\nData Mahasiswa:\n";
        for (int i = 0; i < jumlah; i++) {
            cout << "\nMahasiswa ke-" << i + 1 << endl;
            cout << "Nama      : " << mahasiswa[i].nama << endl;
            cout << "NIM       : " << mahasiswa[i].nim << endl;
            cout << "Nilai UTS  : " << mahasiswa[i].uts << endl;
            cout << "Nilai UAS  : " << mahasiswa[i].uas << endl;
            cout << "Nilai Tugas: " << mahasiswa[i].tugas << endl;
            cout << "Nilai Akhir: " << mahasiswa[i].nilai_akhir << en

        }

        return 0;
    }

```

Output :

Masukkan jumlah mahasiswa (maksimum 10): 2

Masukkan data untuk mahasiswa ke-1

Nama: Fauzan Rofif

NIM: 2211104036

Nilai UTS: 100

Nilai UAS: 100

Nilai Tugas: 90

Masukkan data untuk mahasiswa ke-2

Nama: Ardiyanto

NIM: 90

Nilai UTS: 90

Nilai UAS: 100

Nilai Tugas: 100

Data Mahasiswa:

Mahasiswa ke-1

Nama : Fauzan Rofif

NIM : 2211104036

Nilai UTS : 100

Nilai UAS : 100

Nilai Tugas: 90

Nilai Akhir: 97

Mahasiswa ke-2

Nama : Ardiyanto

NIM : 90

Nilai UTS : 90

Nilai UAS : 100

Nilai Tugas: 100

Nilai Akhir: 97

2. Buatlah ADT pelajaran sebagai berikut di dalam file "pelajaran.h":

```
type pelajaran <
    namaMapel : string
    kodeMapel : string
>
fungsi create_pelajaran( namapel : string, kodepel : string ) →
    pelajaran
prosedur tampil_pelajaran( pel : pelajaran )
```

Buatlah implementasi ADT pelajaran pada file "pelajaran.cpp"

Cobalah hasil implementasi ADT pada file "main.cpp"

```
using namespace std;
int main() {
    string namapel = "Struktur Data";
    string kodepel = "STD";
    pelajaran pel = create_pelajaran(namapel, kodepel);
    tampil_pelajaran(pel);

    return 0;
}
```

Contoh output:

```
nama pelajaran : Struktur Data
nilai : STD
```

Input :

1. Main.cpp


```
// main.cpp
#include <iostream>
#include "pelajaran.h"

using namespace std;

int main() {
    string namaPel = "Struktur Data";
    string kodePel = "STD";

    pelajaran pel = create_pelajaran(namaPel, kodePel);
    tampil_pelajaran(pel);

    return 0;
}
```

2. Pelajaran.cpp

```
// pelajaran.cpp
#include "pelajaran.h"
#include <iostream>

// Function to create a pelajaran
pelajaran create_pelajaran(string namaMapel, string kodeMapel) {
    pelajaran pel;
    pel.namaMapel = namaMapel;
    pel.kodeMapel = kodeMapel;
    return pel;
}

// Procedure to display pelajaran
void tampil_pelajaran(pelajaran pel) {
    cout << "Nama Mata Pelajaran: " << pel.namaMapel << endl;
    cout << "Kode Mata Pelajaran: " << pel.kodeMapel << endl;
}
```

3. Pelajaran.h

```

// pelajaran.h
#ifndef PELAJARAN_H
#define PELAJARAN_H

#include <string>
using namespace std;

struct pelajaran {
    string namaMapel;
    string kodeMapel;
};

// Function to create a pelajaran
pelajaran create_pelajaran(string namaMapel, string kodeMapel);

// Procedure to display pelajaran
void tampil_pelajaran(pelajaran pel);

#endif

```

Output:

```

nama pelajaran : Struktur Data
nilai : STD

```

3. Buatlah program dengan ketentuan :

- 2 buah *array 2D integer* berukuran 3x3 dan 2 buah *pointer integer*
- fungsi/prosedur yang menampilkan isi sebuah *array integer 2D*
- fungsi/prosedur yang akan menukarkan isi dari 2 *array integer 2D* pada posisi tertentu

```

#include <iostream>
using namespace std;

const int SIZE = 3; // Size of the 2D arrays

// Function to display a 3x3 2D array
void displayArray(int arr[SIZE][SIZE]) {
    for (int i = 0; i < SIZE; i++) {
        for (int j = 0; j < SIZE; j++) {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
    cout << endl;
}

// Function to swap elements of two arrays at a specific position
void swapElements(int arr1[SIZE][SIZE], int arr2[SIZE][SIZE], int
// Check if the specified position is within bounds
    if (row >= 0 && row < SIZE && col >= 0 && col < SIZE) {
        // Use pointers to swap elements
        int *ptr1 = &arr1[row][col];
        int *ptr2 = &arr2[row][col];

        // Swap the values at the specified position
        int temp = *ptr1;
        *ptr1 = *ptr2;
        *ptr2 = temp;
    } else {
        cout << "Position out of bounds!" << endl;
    }
}

int main() {
    // Initialize two 3x3 2D arrays
    int array1[SIZE][SIZE] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
    int array2[SIZE][SIZE] = { {9, 8, 7}, {6, 5, 4}, {3, 2, 1} };

    // Display initial arrays
    cout << "Array 1:" << endl;
    displayArray(array1);

    cout << "Array 2:" << endl;
    displayArray(array2);
}

```

```
// Swap elements at a specific position (for example, position)
int row = 1;
int col = 1;
cout << "Swapping elements at position (" << row << ", " << col << ")\n";
swapElements(array1, array2, row, col);

// Display arrays after swapping
cout << "Array 1 after swapping:" << endl;
displayArray(array1);

cout << "Array 2 after swapping:" << endl;
displayArray(array2);

return 0;
}
```

Output :

```
PS C:\Users\LENOVO\Documents\ITTP\TUGAS SEMESTER 5 ITTP\Praktikum STD\STD_FauzanRofifArdiyanto_2211104036\03_STD\Unguided\output\output> & .\'ungudided3.exe'
```

```
Array 1:
```

```
1 2 3
4 5 6
7 8 9
```

```
Array 2:
```

```
9 8 7
6 5 4
3 2 1
```

```
Swapping elements at position (1, 1)...
```

```
Array 1 after swapping:
```

```
1 2 3
4 5 6
7 8 9
```

```
Array 2 after swapping:
```

```
9 8 7
6 5 4
3 2 1
```