

**LAPORAN PRAKTIKUM STRUKTUR
DATA DAN ALGORITMA**

**MODUL I
TIPE DATA**



Disusun Oleh :

NAMA : Muhamad ihsan

NIM : 2311102077

Dosen

Wahyu Andi Saputra, S.pd., M,Eng

**PROGRAM STUDI S1TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2024**

A. Dasar Teori

Tipe data adalah adalah sebuah pengklasifikasian data berdasarkan jenis data tersebut. Tipe data dibutuhkan agar kompiller dapat mengetahui bagaimana sebuah data akan digunakan. Adapun tipe data yang akan dipelajari, sebagai berikut :

1. Tipe data Primitif
2. Tipe data Abstrak
3. Tipe data Koleksi

Tipe data Primitif

Tipe data primitif adalah tipe data yang sudah ditentukan oleh sistem, tipe data primitif ini disediakan oleh banyak bahasa pemrograman, perbedaannya terletak pada jumlah bit yang dialokasikan untuk setiap bit pada tipe data primitif tergantung pada bahasa pemrograman, compiler dan sistem operasinya. Contoh tipe data primitif adalah :

- a. Int : adalah tipe data yang digunakan untuk menyimpan bilangan bulat seperti 12, 1, 4, dan sebagainya.
- b. Float : tipe data yang digunakan untuk menyimpan bilangan desimal seperti 1.5, 2.1, 3.14, dan sebagainya.
- c. Char : berfungsi untuk menyimpan data berupa sebuah huruf. Biasanya digunakan untuk simbol seperti A, B, C dan seterusnya
- d. Boolean : tipe data ini digunakan untuk menyimpan nilai boolean yang hanya memiliki dua nilai yaitu true dan false.

Tipe Data Abstrak

Tipe data abstrak atau yang biasa disebut Abstrak Data Tipe (ADT) merupakan tipe data yang dibentuk oleh programmer itu sendiri. Pada tipe data abstrak bisa berisi banyak tipe data, jadi nilainya bisa lebih dari satu dan beragam tipe data. Fitur Class adalah fitur Object Oriented Program (OPP) pada bahasa C++ yang mirip dengan fitur data structures Struct pada bahasa C. Keduanya berfungsi untuk membungkus tipe data di dalamnya sebagai anggota. menurut learn.microsoft.com perbedaan antara Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private.

Tipe Data Koleksi

Tipe data koleksi (Collection Data Type) adalah tipe data yang digunakan untuk mengelompokkan dan menyimpan beberapa nilai atau objek secara bersamaan. Tipe data koleksi memungkinkan Anda menyimpan, mengelola, dan mengakses sejumlah besar data dengan cara yang terstruktur. Ada beberapa tipe data koleksi yang umum digunakan dalam pemrograman, dan di antaranya adalah:

- a. Array : Array adalah struktur data statis yang menyimpan elemen-elemen dengan tipe data yang sama. Elemen-elemen tersebut dapat diakses dengan menggunakan indeks. Array memiliki ukuran tetap yang ditentukan saat deklarasi.
- b. Vector : Vector adalah Standard Template Library (STL) jika di dalam C/C++ memiliki bentuk `std::vector` . Umumnya, vector mirip seperti array yang memiliki kemampuan untuk menyimpan data dalam bentuk elemen-elemen yang alokasi memorinya dilakukan otomatis dan bersebelahan. Kemampuan vector bukan hanya pada jumlah elemen yang dinamis, vector pada C/C++ juga dilengkapi dengan fitur-fitur pelengkap seperti element access, iterators, capacity, modifiers
- c. Map : Map terasa mirip dengan array namun dengan index yang memungkinkan untuk berupa tipe data selain integer. Pada map, indeks tersebut diberi nama "key".

B. Guided

Guided 1

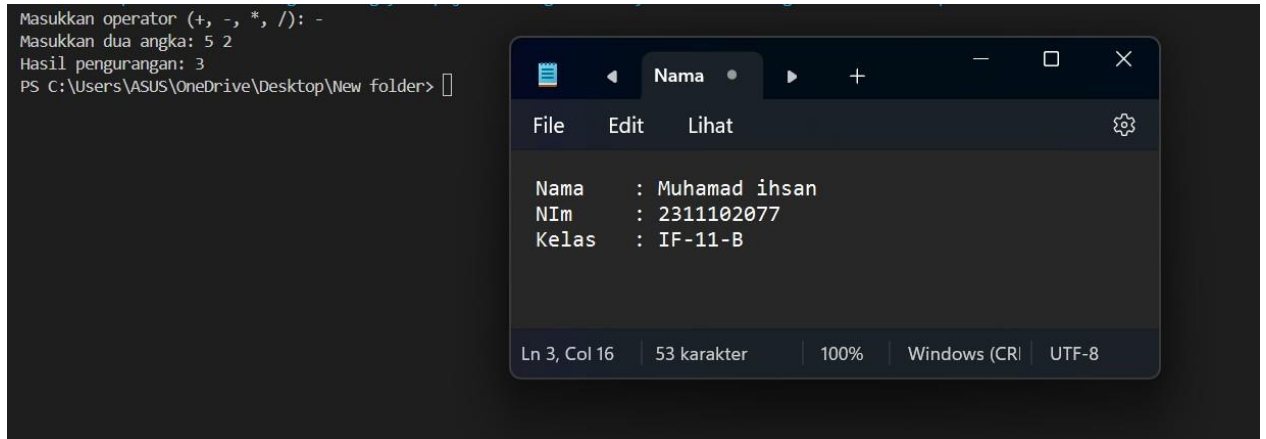
```
#include <iostream>
using namespace std;
int main ()
{
    char op;
    float num1, num2;
    cout << "masukan operator (+,-,*,/)" ;
    cin >> op;

    cout << " masukan angka : ";
    cin >> num1 >> num2;

    switch (op) {
        case '+' :
            cout << num1 + num2 ;
            break;
        case '-' :
            cout << num1 - num2;
            break;
        case '*' :
            cout << num1 * num2;
            break;
        case '/' :
            cout << num1 / num2;
            break;
        default :
            cout << "Error! operator is not correct " ;
```

```
}  
return 0;  
}
```

Screenshots Output



Deskripsi:

Kode di atas adalah program C++ yang meminta pengguna untuk memasukkan operator (+, -, *, /) dan dua angka. Berdasarkan operator yang dimasukkan, program akan melakukan operasi matematika yang sesuai dan menampilkan hasilnya. Jika operator tidak valid, program akan menampilkan pesan kesalahan.

Guided 2

```
#include <stdio.h>

// definisi struktur mahasiswa
struct Mahasiswa
{
    const char *name;
    const char *address;
    int age;
};

int main()
{
    //deklarasi
    struct Mahasiswa mhs1, mhs2;

    //input val to struct mhs1
    mhs1.name = "Dian";
    mhs1.address = "Mataram";
    mhs1.age = 22;

    //input val to struct mhs2
    mhs2.name = "Bambang";
    mhs2.address = "Surabaya";
    mhs2.age = 23;


    // mencetak isi struct mhs1
    printf("## Mahasiswa 1 ##\n");
    printf("Nama: %s\n", mhs1.name);
    printf("Alamat: %s\n", mhs1.address);
    printf("Umur: %d\n", mhs1.age);

    // mencetak isi struct mhs2
    printf("## Mahasiswa 2 ##\n");
    printf("Nama: %s\n", mhs2.name);
    printf("Alamat: %s\n", mhs2.address);
    printf("Umur: %d\n", mhs2.age);

    return 0;
}
```

Screensot Output :

```
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS C:\Users\ASUS\OneDrive\Desktop\New folder>
```



Deskripsi :

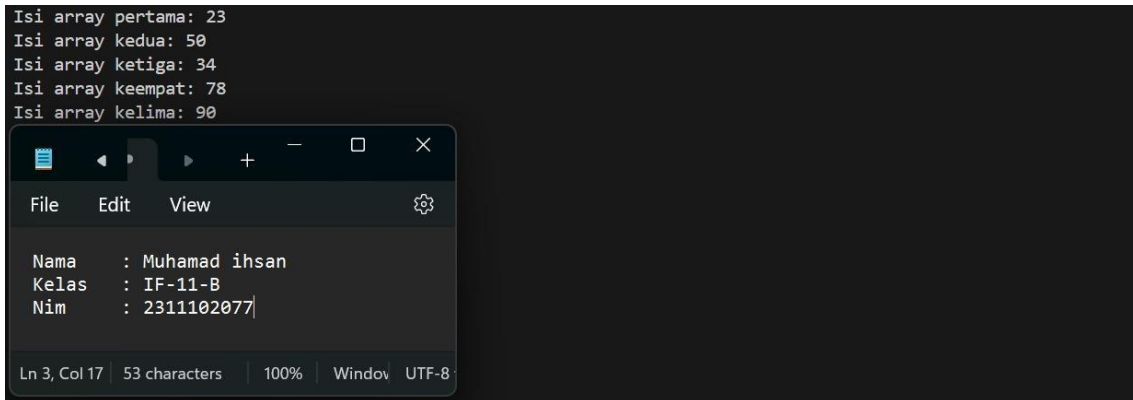
Dalam kode di atas, terdapat definisi struktur dengan nama Mahasiswa. Struktur ini memiliki tiga anggota yaitu Sebuah pointer ke karakter (string) yang menyimpan nama mahasiswa. address, Sebuah pointer ke karakter (string) yang menyimpan alamat mahasiswa. age, Sebuah bilangan bulat (integer) yang menyimpan usia mahasiswa. Deklarasi Variabel di dalam fungsi main(), dua variabel mhs1 dan mhs2 dideklarasikan dengan tipe data struct Mahasiswa. Variabel ini akan digunakan untuk menyimpan informasi tentang dua mahasiswa. Fungsi main() mengembalikan nilai 0 (nol), menandakan bahwa program berakhir dengan sukses.

Guided 3

```
#include <iostream>
using namespace std;
int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5] = {23, 50, 34, 78, 90};

    //mencetak isi array
    cout << "Isi array pertama: " << nilai [0]<< endl;
    cout << "Isi array kedua: " << nilai [1]<< endl;
    cout << "Isi array ketiga: " << nilai [2]<< endl;
    cout << "Isi array keempat: " << nilai [3]<< endl;
    cout << "Isi array kelima: " << nilai [4]<< endl;
    return 0;
}
```

Screenshots Output



The screenshot shows the output of a C++ program in a terminal window. The output displays the contents of five arrays, each with its index and value: 'Isi array pertama: 23', 'Isi array kedua: 50', 'Isi array ketiga: 34', 'Isi array keempat: 78', and 'Isi array kelima: 90'. Below the terminal window, a code editor window is visible, showing a C++ program that declares and initializes an array named 'nilai' with five elements: 23, 50, 34, 78, and 90. The editor also shows the student's name 'Muhamad ihsan', class 'IF-11-B', and NIM '2311102077'.

```
Isi array pertama: 23
Isi array kedua: 50
Isi array ketiga: 34
Isi array keempat: 78
Isi array kelima: 90
```

```
File Edit View
Nama : Muhamad ihsan
Kelas : IF-11-B
Nim : 2311102077
Ln 3, Col 17 | 53 characters | 100% | Window UTF-8
```

Deskripsi:

program di atas merupakan program C++ yang mengilustrasikan penggunaan array. Pada program ini, terdapat deklarasi dan inisialisasi sebuah array dengan nama nilai. Array ini memiliki lima elemen dengan indeks dari 0 hingga 4. Nilai-nilai elemen array diinisialisasi secara berurutan: nilai[0] diatur menjadi 23, nilai[1] diatur menjadi 50, nilai[2] diatur menjadi 34, nilai[3] diatur menjadi 78, dan nilai[4] diatur menjadi 90. Setelah inisialisasi, program mencetak isi array tersebut dengan menggunakan perintah cout. Hasil cetakan menampilkan nilai-nilai elemen array sesuai dengan indeksnya.

C. Unguided

Unguided 1

Buatlah program menggunakan tipe data primitif minimal dua fungsi dan bebas. Menampilkan program, jelaskan program tersebut dan ambil kesimpulan dari materi tipe data primitif!

```
#include <iostream>
using namespace std;
// Fungsi untuk menghitung luas persegi panjang
float hitungLuasPersegiPanjang(float panjang, float lebar) {
    return panjang * lebar;
}
// Fungsi untuk menampilkan hasil perhitungan luas persegi panjang
void tampilkanLuasPersegiPanjang(float panjang, float lebar) {
    float luas = hitungLuasPersegiPanjang(panjang, lebar);
    cout << "Luas persegi panjang dengan panjang " << panjang << " dan
lebar " << lebar << " adalah " << luas << endl;
}
int main() {
    float panjang, lebar;
    cout << "Masukkan panjang persegi panjang: ";
    cin >> panjang;
    cout << "Masukkan lebar persegi panjang: ";
```

```

    cin >> lebar;
    tampilkanLuasPersegiPanjang(panjang, lebar);
    return 0;
}

```

Screenshot Output :

The screenshot shows the output of a C++ program in a terminal window. The output text is:

Masukkan panjang persegi panjang: 17

Masukkan lebar persegi panjang: 5

Luas persegi panjang dengan panjang 17 dan lebar 5 adalah 85

Below the terminal output is a code editor window. The editor has a menu bar with 'File', 'Edit', 'View', and a settings icon. The main text area contains student information:

Nama : Muhamad ihsan

Kelas : IF-11-B

Nim : 2311102077

At the bottom of the editor, a status bar shows 'Ln 3, Col 17', '53 characters', '100%', 'Window', and 'UTF-8'.

Deskripsi :

Kode ini menghitung dan menampilkan luas persegi panjang berdasarkan panjang dan lebar yang dimasukkan oleh pengguna. Dalam kode ini, terdapat dua fungsi mengalikan panjang dan lebar. Tampilkan LuasPersegi Panjang(float panjang, float lebar), Menampilkan hasil perhitungan luas persegi panjang. Pengguna diminta memasukkan nilai panjang dan lebar persegi panjang melalui input. Kode kemudian menghitung luas dengan memanggil fungsi hitungLuasPersegiPanjang. Hasil perhitungan ditampilkan menggunakan fungsi tampilkanLuasPersegiPanjang. Setelah input diberikan, program menghitung dan menampilkan luas persegi Panjang.

Unguided 2 :

Jelaskan fungsi dari class dan struct secara detail dan berikan contoh programnya ?

Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private.

Contoh:

```

#include <iostream>
using namespace std;
// Definisi struct untuk merepresentasikan data mahasiswa
struct Mahasiswa {
    string nama;
    int umur;
}

```



```

};
// Definisi class untuk merepresentasikan sebuah mobil
class Mobil {
private:
    string merek;
    int tahunProduksi;
public:
    // Konstruktor untuk membuat objek Mobil
    Mobil(string merek, int tahunProduksi) {
        this->merek = merek;
        this->tahunProduksi = tahunProduksi;
    }
    // Metode untuk menampilkan informasi mobil
    void tampilkanInfo() {
        cout << "Merek mobil: " << merek << endl;
        cout << "Tahun produksi: " << tahunProduksi << endl;
    }
};

int main() {
    // Menggunakan struct Mahasiswa
    Mahasiswa mhs;
    mhs.nama = "YANTO";
    mhs.umur = 20;
    cout << "Nama mahasiswa: " << mhs.nama << endl;
    cout << "Umur mahasiswa: " << mhs.umur << endl;
    // Menggunakan class Mobil
    Mobil mobil("civic turbo", 2020);
    mobil.tampilkanInfo();

    return 0;
}

```

Screenshot Output :

The screenshot shows the output of a C++ program in a terminal window. The output displays the name 'YANTO', age '20', car brand 'civic turbo', and production year '2020'. Below the terminal, a code editor window is visible, showing a menu bar with 'File', 'Edit', and 'View'. The editor contains a form with fields for 'Nama' (Muhamad ihsan), 'Kelas' (IF-11-B), and 'Nim' (2311102077). The status bar at the bottom indicates 'Ln 3, Col 17', '53 characters', '100%', 'Window', and 'UTF-8'.

```

Nama mahasiswa: YANTO
Umur mahasiswa: 20
Merek mobil: civic turbo
Tahun produksi: 2020

```

File Edit View

Nama : Muhamad ihsan
Kelas : IF-11-B
Nim : 2311102077

Ln 3, Col 17 | 53 characters | 100% | Window | UTF-8

Deskripsi:

Kode ini menunjukkan cara menggunakan struktur dan kelas dalam bahasa pemrograman C++ dengan 2 anggota yaitu nama dan umur. dan menggunakan class Mobil untuk merepresentasikan mobil dengan dua atribut yaitu merek dan tahunProduksi. Dalam class Mobil, terdapat Metode showInfo() digunakan untuk menampilkan informasi tentang mobil.

Unguided

3

Buat dan jelaskan program menggunakan fungsi map dan jelaskan perbedaan dari array dengan map?

Array adalah struktur data statis yang menyimpan elemen-elemen dengan tipe data yang sama. Elemen-elemen tersebut dapat diakses dengan menggunakan indeks. Array memiliki ukuran tetap yang ditentukan saat deklarasi. Sedangkan

Map terasa mirip dengan array namun dengan index yang memungkinkan untuk berupa tipe data selain integer. Pada map, indeks tersebut diberi nama “key”. Pada std::map digunakan Self-Balancing Tree khususnya Red-Black Tree.

Code Program menggunakan fungsi Map

```
#include <iostream>
#include <map>
using namespace std;
int main() {
    // Membuat objek map dengan key bertipe string dan value bertipe int
    map<string, int> umur;
    // Menambahkan elemen ke dalam map
    umur["Agus"] = 19;
    umur["yanto"] = 18;
    umur["chery"] = 20;
    // Mengakses nilai dari map menggunakan key
    cout << "Umur Agus: " << umur["Agus"] << endl;
    cout << "Umur yanto: " << umur["yanto"] << endl;
    cout << "Umur chery: " << umur["chery"] << endl;

    return 0;
}
```

[illegible]

Kode ini membuat sebuah map dengan key bertipe string dan value bertipe int. Kemudian, elemen-elemen (nama dan umur) ditambahkan ke dalam map. Nilai dari map diakses menggunakan key. fungsi map dari library map untuk membuat sebuah map yang memiliki key bertipe string dan value bertipe int.

Tipe data primitif adalah tipe data yang ditentukan sistem. Tipe data primitif ini disediakan oleh banyak bahasa pemrograman, perbedaannya terletak pada jumlah bit yang dialokasikan untuk setiap bit dalam tipe data primitif, tergantung pada bahasa pemrograman, compiler, dan sistem operasi. Struct dan Class adalah akses default di mana Struct bersifat publik dan Kelas bersifat pribadi. Dan Array dan Map berbeda karena Array mempunyai tipe data yang sama sedangkan Map mempunyai tipe data yang berbeda dengan integer.

<https://ejournal.itelkom-pwt.ac.id/index.php/infotel/article/view/103> (arief hendra saptadi,2012)