

**LAPORAN PRAKTIKUM  
STRUKTUR DATA DAN ALGORITMA**

**MODUL I  
TIPE DATA**



**Dosen : Wahyu Andi Saputra, S.Pd., M.Eng.**

**Disusun oleh:**

**FALAH ASBI PANGESTU (2311102045)**

**IF-11-B**

**PROGRAM STUDI S1 INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
2024**

# BAB I

## DASAR TEORI

### 1. Tipe Data

Dalam pemrograman, istilah "tipe data" selalu ada ketika berbicara tentang variabel. Ini adalah jenis nilai yang akan ditampung oleh variabel dan harus ada ketika membuat variabel agar memiliki nilai yang jelas dan spesifik ketika digunakan. Dalam C++ ini, tipe data terbagi menjadi tiga bagian utama, yaitu tipe data primitive, derived, dan user defined.

Primitive	Derived	User Defined
<ul style="list-style-type: none"><li>• Integer</li><li>• Short</li><li>• Long long integer</li><li>• Float</li><li>• Double</li><li>• Char</li><li>• String</li></ul>	<ul style="list-style-type: none"><li>• Function</li><li>• Array</li><li>• Pointer</li><li>• Reference</li></ul>	<ul style="list-style-type: none"><li>• Class</li><li>• Structure</li><li>• Union</li><li>• Enum</li><li>• Typedef</li></ul>

### 2. 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 di alokasikan untuk setiap bit pada tipe data primitif tergantung pada bahasa pemrograman, compiler dan sistem operasinya. Contoh tipe data primitif adalah:

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

### 3. Tipe Data Abstrak

Tipe data abstrak atau yang biasa disebut Abstrak Data Tipe(ADT) merupakan tipe data yang dibentuk oleh programmer. Pada tipe data abstrak bisa berisi banyak data, jadi nilainya bisa lebih dari satu dan beragam tipe data. Fitur Class adalah fitur Object Oriented Program(OOP) 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](https://learn.microsoft.com/en-us/cpp/faq/struct-vs-class) perbedaan antara Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private.

### 4. 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 elemenelemen 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”. Pada `std::map` digunakan Self-Balancing Tree khususnya Red-Black Tree.

## BAB II

### GUIDED

#### LATIHAN – GUIDED

##### 1. Guided 1

Tipe data primitif.

##### Source Code

```
#include <iostream>

using namespace std;

int main() {
    char op;
    float num1, num2;

    // Meminta pengguna untuk memasukkan operator: +, -, *, /
    cout << "Masukkan operator (+, -, *, /): ";
    cin >> op;

    // Meminta pengguna untuk memasukkan operand
    cout << "Masukkan dua angka: ";
    cin >> num1 >> num2;

    // Switch statement
    switch (op) {
        // If user enter +
        case '+':
            cout << "Hasil penjumlahan: " << num1 + num2;
            break;
            // If user enter -
        case '-':
            cout << "Hasil pengurangan: " << num1 - num2;
            break;
            // If user enter *
```

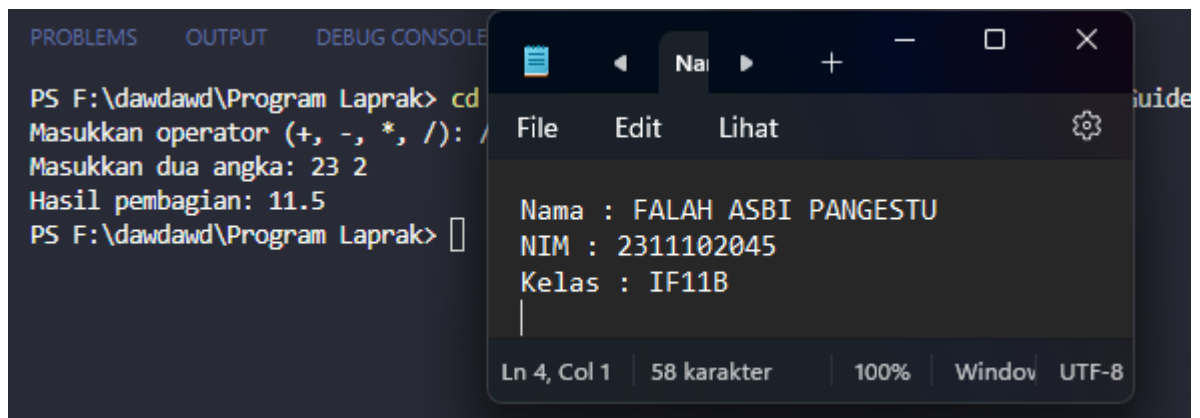
```

        case '*':
            cout << "Hasil perkalian: " << num1 * num2;
            break;
            // If user enter /
        case '/':
            if (num2 != 0)
                cout << "Hasil pembagian: " << num1 / num2;
            else
                cout << "Error! Tidak dapat melakukan pembagian
dengan nol.";
            break;
            // If the operator is other than +, -, * or /,
            // error message will display
        default:
            cout << "Error! Operator tidak benar.";
    }

    return 0;
}

```

### Screenshoot program



### Deskripsi program

Ini adalah program kalkulator yang menggunakan tipe data primitif. Dia meminta pengguna untuk memasukkan (+, -, \*, /) dan kemudian meminta dua angka yang akan dilaksanakan oleh program. Dalam kasus ini, saya memasukkan bagi (/) dan program kemudian membaginya, yaitu 11,5.

## 2. Guided 2

Tipe data abstrak.

### Source code

```
#include <stdio.h>

//Struct

struct Mahasiswa

{

    const char *name;

    const char *address;

    int age;

};

int main()

{

    // menggunakan struct

    struct Mahasiswa mhs1, mhs2;

    // mengisi nilai ke struct

    mhs1.name = "Dian";

    mhs1.address = "Mataram";

    mhs1.age = 22;

    mhs2.name = "Bambang";

    mhs2.address = "Surabaya";

    mhs2.age = 23;

    // mencetak isi struct

    printf("## Mahasiswa 1 ##\n");

    printf("Nama: %s\n", mhs1.name);

    printf("Alamat: %s\n", mhs1.address);

    printf("Umur: %d\n", mhs1.age);

    printf("## Mahasiswa 2 ##\n");

    printf("Nama: %s\n", mhs2.name);

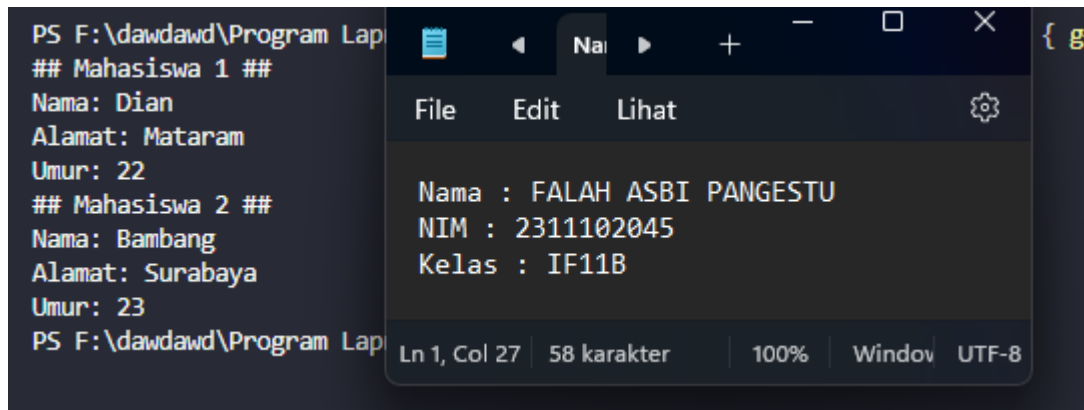
    printf("Alamat: %s\n", mhs2.address);

    printf("Umur: %d\n", mhs2.age);

    return 0;

}
```

## Screenshoot program



## Deskripsi program

Program ini menggunakan tipe data abstrak, pada tipe data abstrak bisa berisi banyak tipe data, jadi nilainya bisa lebih dari satu dan beragam tipe data. Tipe data abstrak (ADT) adalah tipe objek yang perilakunya ditentukan oleh set nilai dan operasi. Dalam definisi ADT, hanya disebutkan apa yang akan dilakukan, tetapi tidak dijelaskan bagaimana operasi tersebut akan dilakukan. Itu tidak menentukan algoritma yang digunakan untuk menjalankan operasi atau bagaimana data akan diatur dalam memori. Memberikan perspektif yang tidak bergantung pada implementasi, itu disebut "abstrak".

### 3. Guided 3

Tipe data koleksi.

#### Source code

```
#include <iostream>

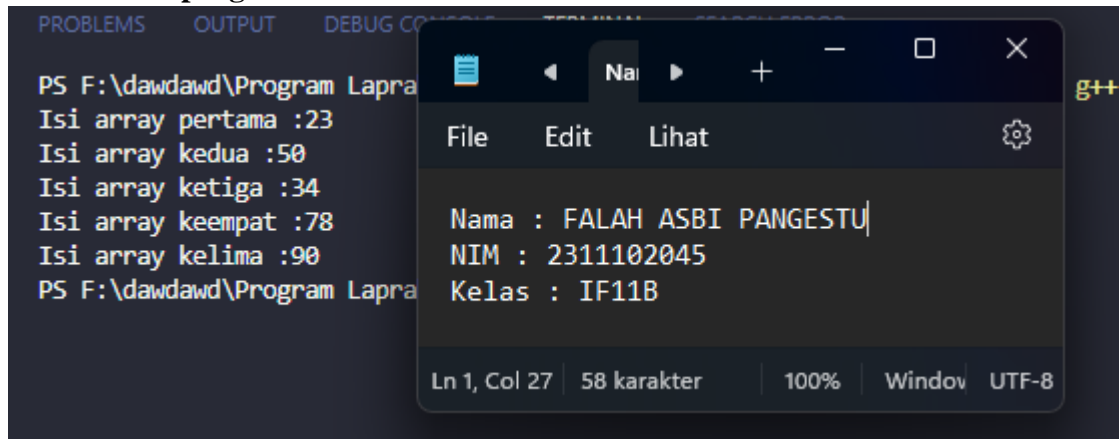
using namespace std;

int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5];
    nilai[0] = 23;
    nilai[1] = 50;
    nilai[2] = 34;
    nilai[3] = 78;
    nilai[4] = 90;

    //mencetak 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;  
}
```

### Screenshoot program



### Deskripsi program

Program ini menggunakan tipe data kolektif dengan mendeklarasikan array, dimana program tersebut memiliki 5 data yaitu dalam array pertama 23, kedua 50, ketiga 34, keempat 78, kelima 9. Selanjutnya program menampilkan array.



## BAB III

### UNGUIDED

#### TUGAS – UNGUIDED

##### 1. 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!

##### Source Code

```
#include <iostream>

using namespace std;

// Membuat fungsi luas persegi panjang
int LuasPersegiPanjang(int a, int b) {
    return 2*(a+b);
}

// Membuat fungsi luas persegi
float LuasPersegi(float a, float b) {
    return a*b;
}

int main() {
    // Menggunakan tipe data primitif int dan float
    int angka1 = 12, angka2 = 2;
    float angka3 = 4.5, angka4 = 7.5;

    // Memanggil fungsi pertama
    cout << "Menghitung Luas Persegi Panjang dan Luas Persegi" <<
endl;

    cout << "===== " <<
endl;

    // Memanggil fungsi pertama
    int HasilLuasPersegiPanjang = LuasPersegiPanjang(angka1,
angka2);
```

```

    cout << "Hasil dari luas persegi panjang " << angka1 << " dan "
    << angka2 << " : " << HasilLuasPersegiPanjang << endl;

    // Memanggil fungsi kedua
    int HasilLuasPersegi = LuasPersegi(angka3, angka4);

    cout << "Hasil dari luas persegi " << angka3 << " dan " <<
    angka4 << " : " << HasilLuasPersegi << endl;

    return 0;
}

```

### Screenshoot program

```

PS F:\dawdawd\Program Laprak> cd "f:\dawdawd\Pro
Menghitung Luas Persegi Panjang dan Luas Persegi
=====
Hasil dari luas persegi panjang 12 dan 2 : 28
Hasil dari luas persegi 4.5 dan 7.5 : 33
PS F:\dawdawd\Program Laprak>

```

### Deskripsi program

Program ini untuk menghitung luas persegi dan persegi panjang dengan tipe data primitif yang menggunakan dua fungsi yaitu fungsi Persegi panjang ( $2 * (a+b)$ ) dan Persegi ( $a*b$ ).

Kesimpulan dari mempelajari tipe data primitif yaitu memungkinkan programmer untuk membuat program yang efisien, aman, mudah di pahami, serta membantu dalam mengoptimalkan memori. Dengan int, float, char, bool, void, array, pointers untuk menyimpan informasi dengan efisien dan efektif.

Integer dan floating-point digunakan untuk data numerik, character untuk representasi karakter tunggal, dan boolean untuk nilai kebenaran. Void digunakan untuk fungsi tanpa nilai kembalian, dan array memungkinkan penyimpanan sejumlah nilai dalam satu variabel. Pointer memberikan fleksibilitas dalam mengelola alamat memori, dan enumeration membantu membuat kode lebih terstruktur.

### 2. Unguided 2

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

#### Source code

```

#include <iostream>

using namespace std;

```

```

// Definition of the structure representing Warga data
struct Warga {
    string nama;
    int64_t nik;
};

// Definition of the class representing motor data
class DealerMotor {
private:
    string MerkMotor;
    string NomerPolisi;

public:
    DealerMotor(string MerkMotor, string NomerPolisi) {
        this->MerkMotor = MerkMotor;
        this->NomerPolisi = NomerPolisi;
    }

    void info() {
        cout << "Merek Motor: " << MerkMotor << endl;
        cout << "Nomer Polisi Motor: " << NomerPolisi << endl;
    }
};

int main() {
    Warga wrg;
    wrg.nama = "Falah Asbi Pangestu";
    wrg.nik = 3329022453234000;

    cout << "Nama Warga yang terdaftar: " << wrg.nama << endl;
    cout << "NIK yang terdaftar: " << wrg.nik << endl;

    // Using the DealerMotor class

```

```

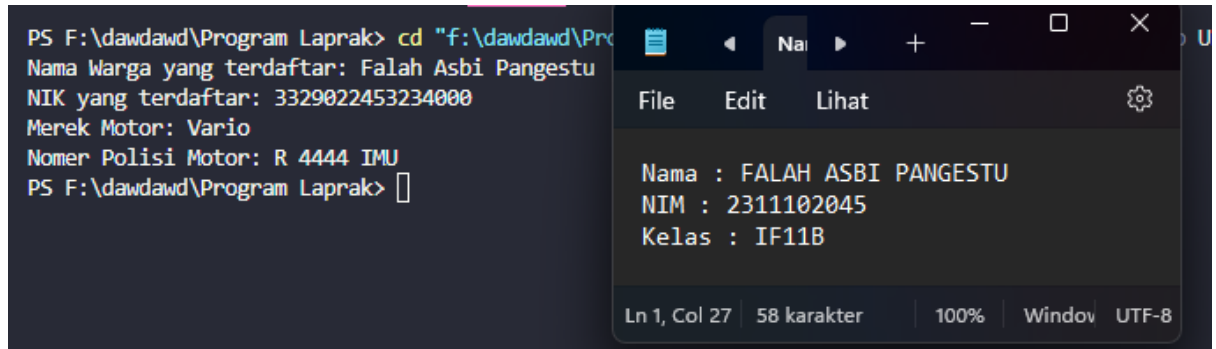
DealerMotor mtr("Vario", "R 4444 IMU");

mtr.info();

return 0;
}

```

### Screenshoot program



### Deskripsi program

Class dan struct merupakan dua prinsip mendasar dalam paradigma pemrograman berorientasi objek (OOP) yang berfungsi untuk merapikan data dan perilaku yang saling terkait ke dalam suatu kerangka yang dikenal sebagai tipe data pengguna.

**Struct :** Di dalam anggota anggota struct memiliki aksesibilitas default 'public' artinya anggota-anggota dapat diakses langsung dari luar struct, Struct tidak mendukung inheritance. Tidak dapat diwariskan dari struct atau kelas lain.

**Class :** Anggota-anggota class memiliki aksesibilitas default 'private'. Untuk mengakses anggota dari luar class, perlu menggunakan metode akses seperti getter dan setter. Class juga mendukung inheritance yang dapat diwariskan dari class lain atau struct lain.

### 3. Unguided 3

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

#### Source code

```

#include <iostream>

#include <map>

using namespace std;

int main() {

    // Mendeklarasikan fungsi array

    map<int, string> NamaKeluarga;
}

```

```

NamaKeluarga[2311102045] = "Falah";

NamaKeluarga[4544] = "Asbi";

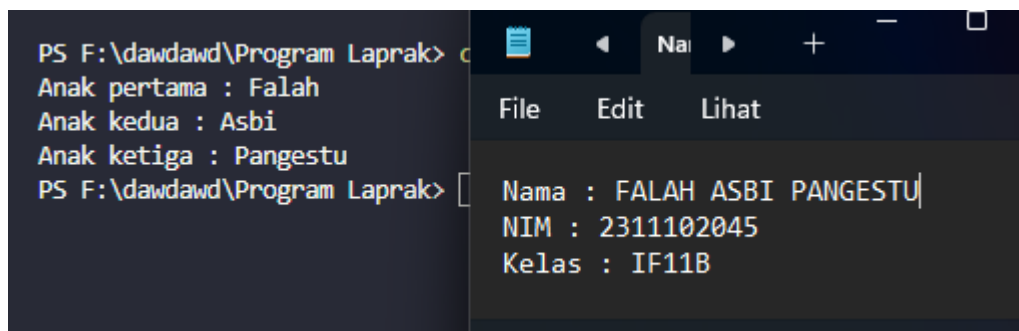
NamaKeluarga[4567] = "Pangestu";


// Mencetak array
cout << "Anak pertama : " << NamaKeluarga[2311102045] << endl
      << "Anak kedua : " << NamaKeluarga[4544] << endl
      << "Anak ketiga : " << NamaKeluarga[4567] << endl;


return 0;
}

```

### Screenshoot program



### Deskripsi program

Program ini menggunakan fungsi map yang sebagaimana indeksinya adalah objek sembarang.

Map adalah seperti array yang indeksinya adalah objek sembarang, bukan integer. Pada map, objek yang digunakan sebagai “indeks” disebut kunci (key). Objek yang ditunjuk oleh indeks tersebut disebut nilai (value). Satu kunci hanya boleh menunjuk pada satu nilai, akan tetapi satu nilai bisa ditunjuk oleh beberapa kunci.

## **BAB IV**

### **KESIMPULAN**

Dalam konteks pengembangan perangkat lunak, saya memiliki pemahaman yang mendalam tentang bagaimana data diorganisir, disimpan, dan dimanipulasi. Memahami jenis data: Integer adalah untuk bilangan bulat, Float adalah untuk bilangan desimal, Char adalah untuk data huruf, dan Boolean adalah untuk menyimpan dua nilai, yaitu benar dan salah.

## **DAFTAR PUSTAKA**

Putra, Muhammad Taufik D., et al. *BELAJAR DASAR PEMROGRAMAN DENGAN C++*.  
Edited by Damayanti, Evi, CV WIDINA MEDIA UTAMA, 2022.

Asisten Praktikum, “Modul 1 Tipe Data”, Learning Management System, 2024