

**LAPORAN PRAKTIKUM
STRUKTUR DATA DAN ALGORITMA**

**MODUL I
TIPE DATA**



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

Disusun oleh:

MUHAMMAD AULIA MUZZAKI NUGRAHA (2311102051)

IF-11-B

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

BAB I

DASAR TEORI

1. Tipe Data

Berbicara mengenai pemrograman, tidak lepas dengan yang namanya Tipe Data. Tipe data merupakan sebuah jenis nilai atau berupa nilai apa yang akan ditampung oleh sebuah variabel. Tipe data ini wajib ada ketika akan membuat sebuah variabel agar variabel tersebut memiliki nilai yang spesifik dan jelas ketika digunakan nantinya. Untuk format penulisan tipe data pada C++ adalah (Hidayat et al., n.d.):

Tipe_data [spasi] nama_variabel;

Atau

Tipe_data [spasi] nama_variabel = value;

atau

**tipe_data [spasi] nama_variabel, nama_variabel2,,
nama_variabel_N;**

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

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 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 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". 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 *
```

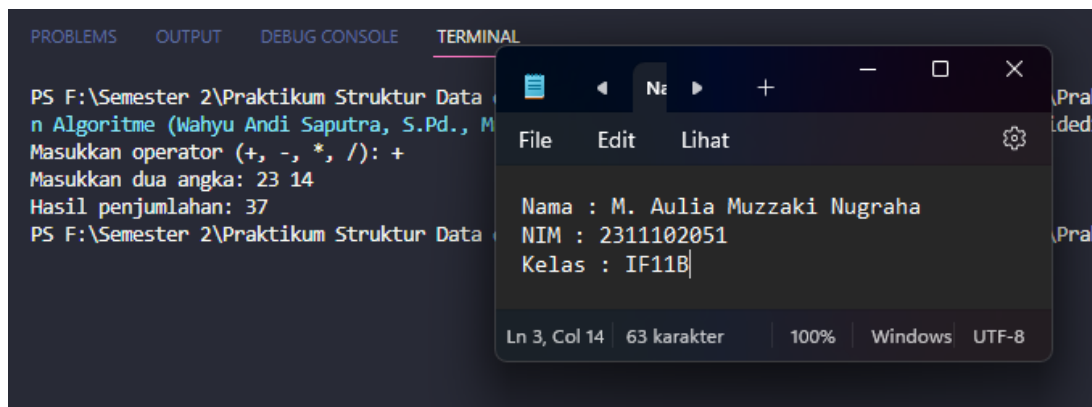
```

        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

Program ini semacam kalkulator menggunakan tipe data primitif, Program akan meminta user untuk memasukan (+, -, *, /) dan selanjutnya program akan meminta dua angka yang akan di laksanakan oleh si program. Contohnya saya memasukan plus (+) dan selanjutnya memasukan dua angka yaitu 23 dan 14 dan program akan menjumlahkannya yaitu 37.

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);

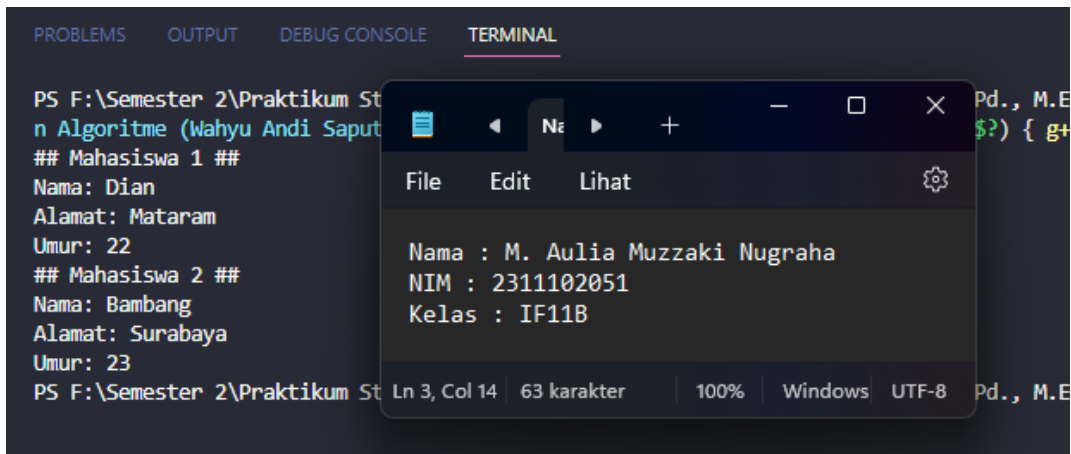
    printf("Alamat: %s\n", mhs2.address);

    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. Menurut [geeksforgeeks.org](https://www.geeksforgeeks.org/) Tipe Data Abstrak (ADT) adalah tipe (atau kelas) untuk objek yang perilakunya ditentukan oleh sekumpulan nilai dan serangkaian operasi. Definisi ADT hanya menyebutkan operasi apa yang akan dilakukan tetapi tidak menyebutkan bagaimana operasi tersebut akan dilaksanakan. Itu tidak menentukan bagaimana data akan diatur dalam memori dan algoritma apa yang digunakan untuk mengimplementasikan operasi. Disebut “abstrak” karena memberikan pandangan yang tidak bergantung pada implementasi.

3. Guided 3

Tipe data koleksi.

Source code

```
#include <iostream>

using namespace std;

int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5];

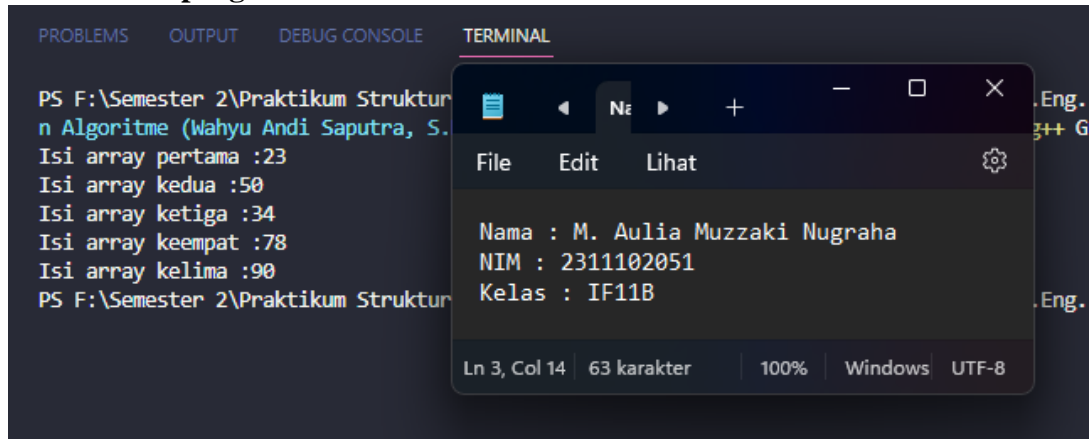
    nilai[0] = 23;
    nilai[1] = 50;
    nilai[2] = 34;
    nilai[3] = 78;
    nilai[4] = 90;

    //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 tersebut.

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;

// Fungsi pertama untuk menjumlahkan dua angka
int pengurangan(int a, int b) {
    return a - b;
}

// Fungsi kedua untuk mengalikan dua angka
float bagi(float a, float b) {
    return a / b;
}

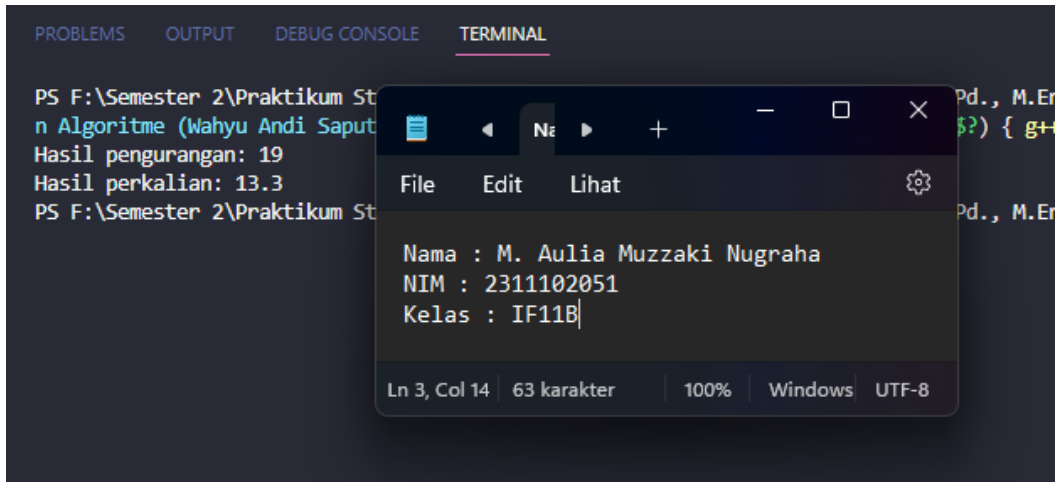
int main() {
    // Menggunakan tipe data primitif int dan float
    int angka1 = 2024, angka2 = 2005;
    float angka3 = 19.95, angka4 = 1.5;

    // Memanggil fungsi pertama
    int hasilPengurangan = pengurangan(angka1, angka2);
    cout << "Hasil pengurangan: " << hasilPengurangan << endl;

    // Memanggil fungsi kedua
    float hasilPembagi = bagi(angka3, angka4);
    cout << "Hasil perkalian: " << hasilPembagi << endl;
```

```
    return 0;
}
```

Screenshoot program



Deskripsi program

Program ini semacam kalkulator dengan tipe data primitif yang menggunakan dua fungsi yaitu fungsi Pengurangan dan pembagian. Contohnya dalam fungsi pengurangan yaitu $2024 - 2005 = 19$ dan dalam fungsi perkalian $19.95 / 1,5 = 13,35$.

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 merek;
        string nopol;

    public:
        DealerMotor(string merek, string nopol) {
            this->merek = merek;
            this->nopol = nopol;
        }

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

    int main() {
        Warga wrg;
        wrg.nama = "Aulia Muzzaki";
        wrg.nik = 33290224567000;

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

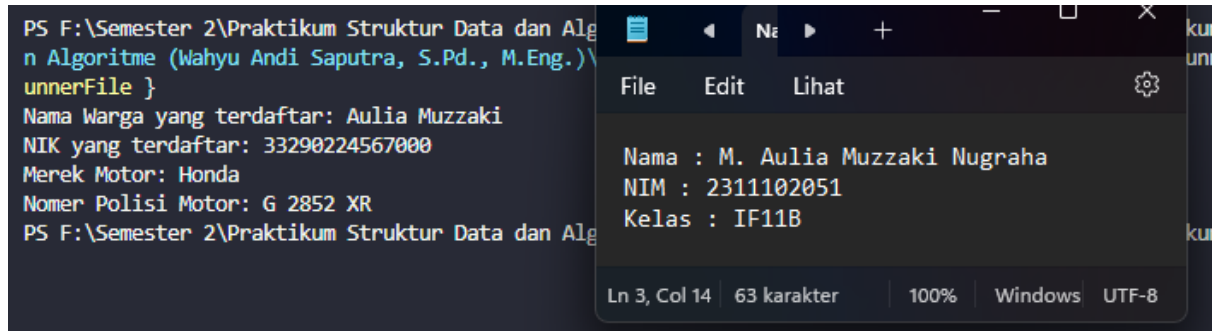
        // Using the DealerMotor class
        DealerMotor mtr("Honda", "G 2852 XR");
        mtr.info();

        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 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[0] = "Lia";
    NamaKeluarga[1] = "Iqbal";
    NamaKeluarga[2] = "Akbar";
    NamaKeluarga[3] = "Intan";
    NamaKeluarga[4] = "Zaki";
```

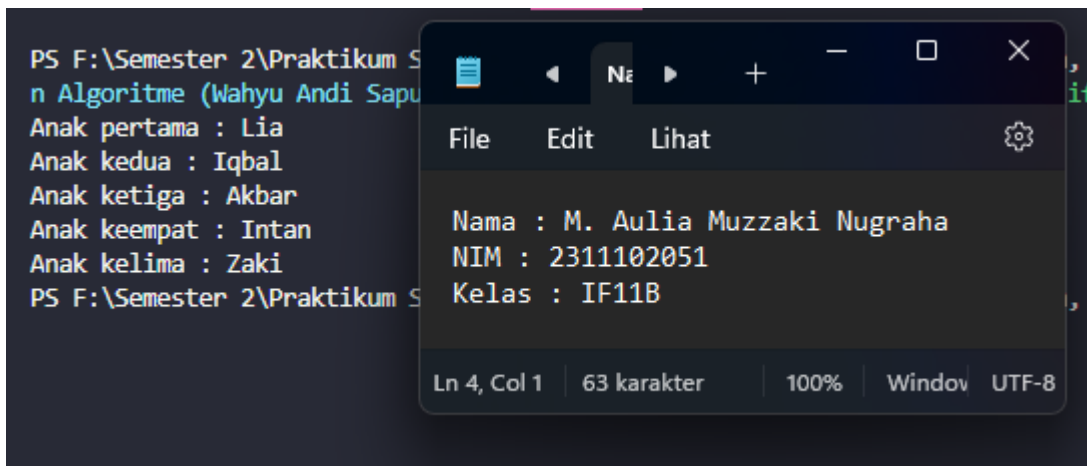
```

// Mencetak array
cout << "Anak pertama : " << NamaKeluarga[0] << endl
    << "Anak kedua : " << NamaKeluarga[1] << endl
    << "Anak ketiga : " << NamaKeluarga[2] << endl
    << "Anak keempat : " << NamaKeluarga[3] << endl
    << "Anak kelima : " << NamaKeluarga[4] << endl;

return 0;
}

```

Screenshoot program



Deskripsi program

Program ini menjelaskan nama panggilan keluarga dengan menggunakan fungsi map.

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.

Perbedaan Map dengan Array

Array	Map
<ul style="list-style-type: none"> • Elemen-elemen array diakses menggunakan indeks numerik yang dimulai dari 0. • Ukuran array biasanya tetap, dan elemen-elemen dapat diakses dan dimodifikasi secara langsung. 	<ul style="list-style-type: none"> • Map menggunakan pasangan kunci-nilai dan menyimpan elemen-elemen secara terurut berdasarkan kunci. • Kunci dalam map harus unik, dan nilai dapat diakses menggunakan kunci.

BAB IV

KESIMPULAN

Dalam Struktur Data memahami yang mendalam tentang bagaimana data diorganisir, disimpan, dan dimanipulasi dalam konteks pengembangan perangkat lunak. Memahami Tipe Data Integer yang digunakan untuk Bilangan bulat, Float yang digunakan untuk Bilangan Desimal, Char yang digunakan untuk menyimpan data sebuah huruf, dan Boolean digunakan untuk menyimpan dua nilai yaitu true dan false.

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