

**LAPORAN PRAKTIKUM STRUKTUR  
DATA DAN ALGORITMA**

**MODUL I  
PENGENALAN CODE BLOCKS**



**Disusun Oleh :**

NAMA : Muhammad Fathiakmal Lutfi Alifia

NIM : 18102096

**Dosen**

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

**PROGRAM STUDI S1 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 kompiler 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](https://learn.microsoft.com/en-us/cpp/build/struct-vs-class) 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”. Pada

## 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

```
masukan operator (+,-,*,/)+
masukan angka : 7
9
16
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1>
```

Fathi 18102096

### Deskripsi:

Program ini menjelaskan tentang kalkulator sederhana yang meminta pengguna untuk memasukkan operator matematika (+, -, \*, /) dan dua angka. Setelah itu, program akan melakukan operasi sesuai dengan operator yang dimasukkan oleh pengguna dan menampilkan hasilnya. Program menggunakan switch case untuk menentukan operasi yang akan dilakukan berdasarkan operator yang dimasukkan pengguna.

## 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;
}
```

Screenot Output :

```
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> 18102096 Fathi
```

Deskripsi :

Program ini adalah contoh penggunaan struct dalam bahasa pemrograman C++. Struct digunakan untuk mengelompokkan beberapa variabel yang memiliki jenis data yang berbeda . Pada program terdapat definisi struct Mahasiswa yang memiliki tiga macam yaitu name, address, dan age. Setelah itu, struct Mahasiswa digunakan untuk mendeklarasikan dua variabel mhs1 dan mhs2 . Kemudian, name, address, dan age pada masing-masing variabel mhs1 dan mhs2 diinisialisasi dengan menggunakan operator . Setelah itu, nilai-nilai dari mhs1 dan mhs2 dicetak ke layar menggunakan fungsi printf untuk menampilkan informasi mahasiswa yang telah disimpan dalam struct.

### 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

```
Isi array pertama: 23
Isi array kedua: 50
Isi array ketiga: 34
Isi array keempat: 78
Isi array kelima: 90
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> 18102096 Fathi
```

### Deskripsi:

Program ini contoh penggunaan array untuk menyimpan sekumpulan data dengan tipe data yang sama dalam satu variabel. terdapat deklarasi dan inisialisasi array nilai 5

elemen yaitu 23, 50, 34, 78, dan 90. Setelah itu, nilai-nilai dari array t dicetak ke layar menggunakan cout untuk menampilkan isi dari masing-masing elemen array. Perlu diperhatikan bahwa indeks array dimulai dari 0, sehingga elemen pertama dari array nilai memiliki indeks 0 (nilai[0]), elemen kedua memiliki indeks 1 (nilai[1]), dan seterusnya.

## 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
float hitungLuasPersegi(float sisi) {
    return sisi * sisi;
}

// Fungsi untuk menampilkan hasil perhitungan
void tampilkanHasil(float luas) {
    cout << "Luas persegi adalah: " << luas << endl;
}

int main() {
    // Data primitif
    float sisiPersegi = 5.0;

    // Memanggil fungsi untuk menghitung luas persegi
    float luas = hitungLuasPersegi(sisiPersegi);

    // Memanggil fungsi untuk menampilkan hasil
```



```
tampilkanHasil(luas);  
  
return 0;  
}
```

Screenshot Output :

```
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> cd "c:\Users\Administrator\Documents\GitHub\L  
aporan-Praktikum-SDA\Modul1\" ; if ($?) { g++ unguided1.cpp -o unguided1 } ; if ($?) { .\unguided1 }  
Luas persegi adalah: 25  
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> 18102096 fathi
```

Deskripsi :

Program C++ di atas adalah contoh sederhana yang menggunakan data primitif dalam dua fungsi terpisah. Fungsi *hitungLuasPersegi* mengambil panjang sisi persegi sebagai input dan mengembalikan luas persegi berdasarkan rumus sisi dikalikan dengan sisi. Sementara itu, fungsi *tampilkanHasil* menerima luas persegi sebagai argumen dan menampilkan hasilnya ke layar. Dalam fungsi *main*, sebuah variabel *sisiPersegi* dideklarasikan untuk menyimpan panjang sisi persegi, kemudian fungsi *hitungLuasPersegi* dipanggil dengan *sisiPersegi* sebagai argumen untuk menghitung luasnya. Hasil perhitungan kemudian disimpan dalam variabel *luas*, yang kemudian digunakan sebagai argumen dalam pemanggilan fungsi *tampilkanHasil* untuk menampilkan hasilnya ke layar.

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

// Definisi struct untuk merepresentasikan titik di bidang 2D
struct Titik {
    int x;
    int y;
};

// Definisi class untuk merepresentasikan sebuah lingkaran
class Lingkaran {
private:
    Titik pusat;
    double radius;

public:
    // Constructor untuk inisialisasi objek lingkaran
    Lingkaran(Titik pusat, double radius) {
        this->pusat = pusat;
        this->radius = radius;
    }

    // Metode untuk menghitung luas lingkaran
    double hitungLuas() {
        return 3.14 * radius * radius;
    }

    // Metode untuk menampilkan informasi lingkaran
    void tampilkanInfo() {
        cout << "Lingkaran dengan pusat di (" << pusat.x << ", " <<
pusat.y << ") ";
        cout << "dan radius " << radius << endl;
    }
};

int main() {
    // Membuat objek titik dan lingkaran
    Titik titikPusat = {3, 4};
```

```

    Lingkaran lingkaran(titikPusat, 5);

    // Memanggil metode untuk menampilkan informasi lingkaran
    lingkaran.tampilkanInfo();

    // Memanggil metode untuk menghitung dan menampilkan luas
    lingkaran
    cout << "Luas lingkaran: " << lingkaran.hitungLuas() << endl;

    return 0;
}

```

Screenshot Output :

```

Lingkaran dengan pusat di (3, 4) dan radius 5
Luas lingkaran: 78.5
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> 18102095 Fathi

```

Deskripsi

:

Dalam contoh di atas, struct Titik digunakan untuk merepresentasikan titik di bidang 2D dengan dua variabel x dan y, sedangkan class Lingkaran digunakan untuk merepresentasikan sebuah lingkaran dengan properti pusat (yang merupakan sebuah titik) dan radius. Class Lingkaran memiliki metode hitungLuas() untuk menghitung luas lingkaran dan tampilkanInfo() untuk menampilkan informasi tentang lingkaran tersebut. Dalam fungsi main(), objek lingkaran dibuat dengan menggunakan objek titik sebagai pusatnya, dan kemudian metode-metode dari class Lingkaran dipanggil untuk menampilkan informasi dan menghitung luas lingkaran.

### Unguided 3 :

3

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

Di C++, **std::map** adalah struktur data dari Standard Template Library (STL) yang menyimpan data dalam bentuk pasangan kunci-nilai. Setiap elemen dalam **std::map** terdiri dari sebuah kunci dan nilai yang terkait. Perbedaan utama antara array dan **std::map** adalah sebagai berikut:

#### 1. Array:

- Array adalah kumpulan elemen data yang disimpan secara berurutan dalam memori.

- Elemen-elemen dalam array diakses menggunakan indeks numerik yang dimulai dari 0.
- Ukuran array harus ditentukan pada saat pendeklarasiannya, dan ukuran tersebut tetap selama program berjalan.

## 2. **std::map**:

- **std::map** adalah struktur data yang menyimpan elemen-elemennya dalam bentuk pasangan kunci-nilai, di mana kunci bersifat unik.
- Elemen-elemen dalam **std::map** tidak disimpan secara berurutan dalam memori, melainkan diurutkan berdasarkan kunci.
- Elemen-elemen dalam **std::map** dapat diakses menggunakan kunci yang terkait dengannya.
- Ukuran **std::map** dapat berubah secara dinamis saat elemen ditambahkan atau dihapus.

Code Program menggunakan fungsi Map

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

int main() {
    // Deklarasi dan inisialisasi std::map dengan pasangan nama dan usia
    map<string, int> data;
    data["John"] = 25;
    data["Alice"] = 30;
    data["Bob"] = 28;

    // Menampilkan usia seseorang berdasarkan nama
    cout << "Usia John: " << data["John"] << " tahun" << endl;
    cout << "Usia Alice: " << data["Alice"] << " tahun" << endl;
    cout << "Usia Bob: " << data["Bob"] << " tahun" << endl;

    return 0;
}
```

### Screenshot Output :

```
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> cd "c:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Usia John: 25 tahun
Usia Alice: 30 tahun
Usia Bob: 28 tahun
PS C:\Users\Administrator\Documents\GitHub\Laporan-Praktikum-SDA\Modul1> 18102096 Fathi
```

### Deskripsi:

Dalam program di atas, **std::map** digunakan untuk menyimpan pasangan nama dan usia. Kemudian, usia seseorang dapat diakses menggunakan kunci (nama mereka). Perhatikan bahwa elemen-elemen dalam **std::map** tidak disimpan dalam urutan tertentu.

### D. Kesimpulan

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. Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private Dan Array dengan Map itu berbeda karena Array memiliki tipe data yang sama sedangkan Map tipe data selain integer.

### E. Referensi

[https://books.google.co.id/books?hl=id&lr=&id=49ZbEAAQBAJ&oi=fnd&pg=PA2&dq=pemrograman+c%2B%2B+tipe+data&ots=4sUAwZJRLx&sig=4wsaU3w8SCQtvZO9va\\_XwXvY9Pw&redir\\_esc=y#v=onepage&q=pemrograman%20c%2B%2B%20tipe%20data&f=false](https://books.google.co.id/books?hl=id&lr=&id=49ZbEAAQBAJ&oi=fnd&pg=PA2&dq=pemrograman+c%2B%2B+tipe+data&ots=4sUAwZJRLx&sig=4wsaU3w8SCQtvZO9va_XwXvY9Pw&redir_esc=y#v=onepage&q=pemrograman%20c%2B%2B%20tipe%20data&f=false) (Pemrograman C++ AD Samala, BR Fajri , F Ranuarja (2021)).