

**LAPORAN PRAKTIKUM STRUKTUR
DATA DAN ALGORITMA**

**MODUL I
PENGENALAN CODE BLOCKS**



Disusun Oleh :

NAMA : Tegar Serli Arunzika

NIM : 2311102041

Dosen

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

**PROGRAM STUDI S1 TEKNIK 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 (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/build/reference/public-private) 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”. Pad

B. Guided (berisi screenshot source code & output program disertai penjelasannya)

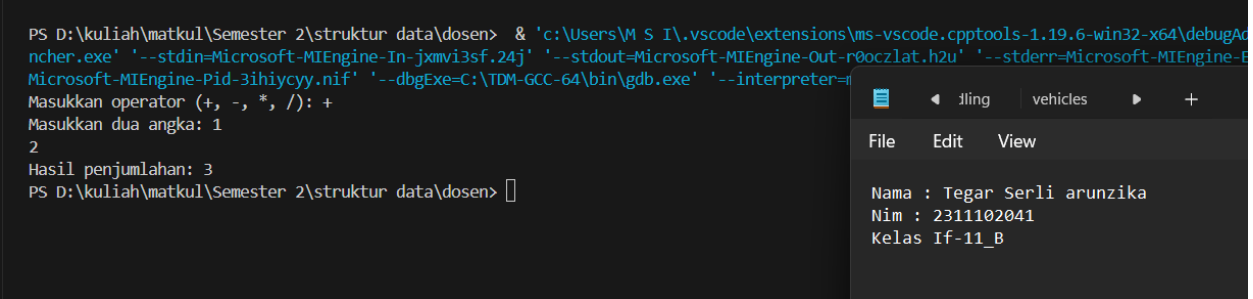
Guided 1

```
#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) {
        case '+':
            cout << "Hasil penjumlahan: " << num1 + num2;
            break;
        case '-':
            cout << "Hasil pengurangan: " << num1 - num2;
            break;
        case '*':
            cout << "Hasil perkalian: " << num1 * num2;
            break;
        case '/':
            if (num2 != 0)
                cout << "Hasil pembagian: " << num1 / num2;
            else
                cout << "Error! Tidak dapat melakukan pembagian dengan nol.";
            break;
        default:
            cout << "Error! Operator tidak benar.";
    }
    return 0;
}
```

Screenshots Output



```
PS D:\kuliah\matkul\Semester 2\struktur data\dosen> & 'c:\Users\M S I\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAd
ncher.exe' '--stdin=Microsoft-MIEngine-In-jxmvi3sf.24j' '--stdout=Microsoft-MIEngine-Out-r0oczlat.h2u' '--stderr=Microsoft-MIEngine-f
Microsoft-MIEngine-Pid-3ihicyy.nif' '--dbgExe=C:\TDM-GCC-64\bin\gdb.exe' '--interpreter=
Masukkan operator (+, -, *, /): +
Masukkan dua angka: 1
2
Hasil penjumlahan: 3
PS D:\kuliah\matkul\Semester 2\struktur data\dosen> 
```

Deskripsi:

Program ini menulis kalkulator sederhana yang meminta pengguna memasukkan operator matematika (+,-,*,/) dengan dua angka. Setelah itu, program akan melakukan operasi sesuai dengan operator ang dimasukan oleh pengguna dan menampilkan hasilnya.

Guide 2

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

return 0;
}

```

Screenshot Output

```

PS D:\kuliah\matkul\Semester 2\struktur data\dosen> & 'c:\Users\M S I\.vscode\extension
ncher.exe' '--stdin=Microsoft-MIEngine-In-bngn3lr0.qz3' '--stdout=Microsoft-MIEngine-Out
Microsoft-MIEngine-Pid-sbbqqvj0.maw' '--dbgExe=C:\msys
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS D:\kuliah\matkul\Semester 2\struktur data\dosen>

```

Deskripsi:

Program ini merupakan contoh penggunaan struktur pada bahasa pemrograman C.

Struktur digunakan untuk mengelompokkan beberapa variabel dengan tipe data berbeda.

Program ini memiliki tiga jenis definisi komposisi siswa: nama, alamat, dan usia.

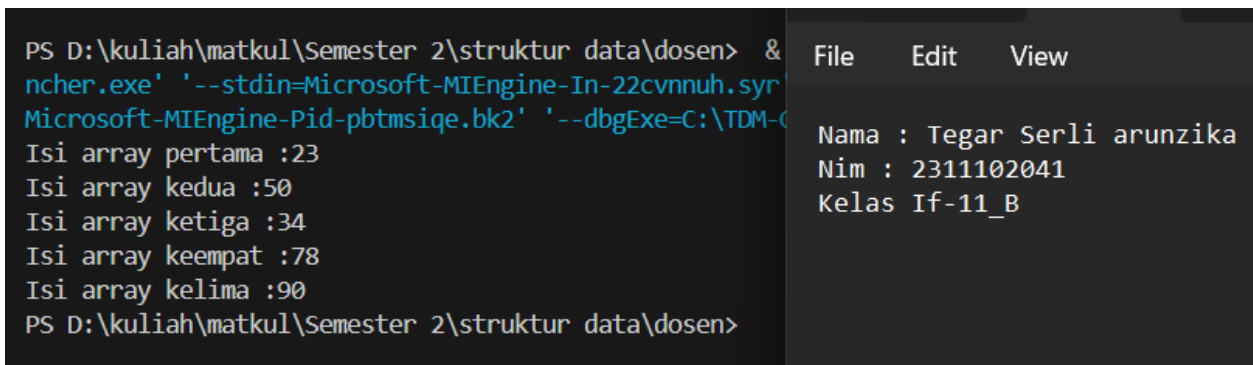
Kemudian, gunakan struktur Student untuk mendeklarasikan dua variabel, mhs1 dan mhs2. Selanjutnya nama, alamat, dan umur diinisialisasi ke variabel mhs1 dan mhs2 menggunakan operator. Fungsi printf kemudian digunakan untuk mencetak nilai mhs1 dan mhs2 ke layar, menampilkan informasi siswa yang disimpan.

Guide 3

```
#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;
}
```

Screenshots Output



```
PS D:\kuliah\matkul\Semester 2\struktur data\dosen> & ncher.exe' '--stdin=Microsoft-MIEngine-In-22cvnuh.syr' Microsoft-MIEngine-Pid-pbtmsiqe.bk2' '--dbgExe=C:\TDM-C Nama : Tegar Serli arunzika Nim : 2311102041 Kelas If-11_B Isi array pertama :23 Isi array kedua :50 Isi array ketiga :34 Isi array keempat :78 Isi array kelima :90 PS D:\kuliah\matkul\Semester 2\struktur data\dosen>
```

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

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

// Fungsi untuk menghitung luas persegi
double hitungLuasPersegi(double sisi) {
    return sisi * sisi;
}

// Fungsi untuk menghitung keliling lingkaran
double hitungKelilingLingkaran(double jariJari) {
    const double PI = 3.14159;
    return 2 * PI * jariJari;
}

int main() {
    double sisiPersegi = 5.0;
    double jariJariLingkaran = 3.0;

    cout << "Luas persegi dengan sisi " << sisiPersegi << " adalah " <<
    hitungLuasPersegi(sisiPersegi) << endl;
    cout << "Keliling lingkaran dengan jari-jari " << jariJariLingkaran
    << " adalah " << hitungKelilingLingkaran(jariJariLingkaran) << endl;

    return 0;
}
```

Screenshots Output

```
Install the latest PowerShell for new features and improvements
PS D:\kuliaah\matkul\Semester 2\struktur data\dosen> & 'nch
ncher.exe' '--stdin=Microsoft-MIEngine-In-qdf5eiua.t0l' 'Micro
Microsoft-MIEngine-Pid-3esgofyu.r2b' '--dbgExe=C:\TDM-GC
Luas persegi dengan sisi 5 adalah 25
Keliling lingkaran dengan jari-jari 3 adalah 18.8495
PS D:\kuliaah\matkul\Semester 2\struktur data\dosen>
```

```
Nama : Tegar Serli arunzika
Nim : 2311102041
Kelas If-11_B|
```

Deskripsi:

Program di atas memiliki dua fungsi, yaitu menghitung luas persegi dan keliling lingkaran berdasarkan jari-jari dan sisi yang di berikan.

Uguided 2

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

// Struct untuk merepresentasikan spearpart mobil
struct Spearpart {
    string nama;
    string jenis;
    double harga;
};

// Class untuk merepresentasikan mobil
class Mobil {
public:
    string merek;
    string model;
    int tahun;
    Spearpart mesin;
    Spearpart roda;

    // Metode untuk menampilkan informasi mobil beserta spearpartnya
    void tampilkanInfo() {
        cout
        <<"=====\\n
        ";
        cout <<"NAMA :          Tegar Serli Arunzika \\n";
        cout <<"Nim :             2311102041 \\n";
        cout <<"kelas:             IF-11-B \\n";
        cout<<"=====
        =====\\n";
        cout << "Mobil " << merek << " " << model << " tahun " << tahun
        << endl;
        cout << "Mesin: " << mesin.nama << ", Jenis: " << mesin.jenis <<
        ", Harga: $" << mesin.harga << endl;
        cout << "Roda: " << roda.nama << ", Jenis: " << roda.jenis << ",
        Harga: $" << roda.harga << endl;
    }
};
```



```

int main() {
    // Membuat objek dari class Mobil dan mengisi atribut-atributnya
    Mobil mobil;
    mobil.merek = "Nissan";
    mobil.model = "R35";
    mobil.tahun = 2020;
    mobil.mesin.nama = "RB26";
    mobil.mesin.jenis = "V6";
    mobil.mesin.harga = 5500.0;
    mobil.roda.nama = "Alloy Wheels";
    mobil.roda.jenis = "TE-37 18-inch";
    mobil.roda.harga = 800.0;

    // Menampilkan informasi mobil dan spearpartnya
    mobil.tampilkanInfo();

    return 0;
}

```

Screenshots Output

```

ncher.exe --stdin=Microsoft-MIEngine-1n-mnxdsdx5.qbr --stdout=Microsoft
Microsoft-MIEngine-Pid-4ywccj0c.tff' '--dbgExe=C:\TDM-GCC-64\bin\gdb.exe'
=====
NAMA :          Tegar Serli Arunzika
Nim  :          2311102041
kelas:          IF-11-B
=====
Mobil Nissan R35 tahun 2020
Mesin: RB26, Jenis: V6, Harga: $5500
Roda: Alloy Wheels, Jenis: TE-37 18-inch, Harga: $800
PS D:\kuliah\matkul\Semester 2\struktur data\dosen>

```

Deskripsi

Dalam program di atas, terdapat struct Spearpart yang merepresentasikan spearpart mobil dengan atribut nama, jenis, dan harga. Selanjutnya, ada class Mobil yang memiliki atribut merek, model, tahun, serta dua objek dari struct Spearpart yaitu mesin dan roda.

Program tersebut membuat objek dari class Mobil dan mengisi atribut-atributnya serta atribut-atribut spearpart mobilnya. Kemudian, program menampilkan informasi tentang mobil beserta spearpartnya menggunakan metode tampilkanInfo.

Unguided 3

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

// Struct untuk merepresentasikan informasi motor
struct Motor {
    string merek;
    string model;
    int tahun;
};

int main() {
    // Membuat map dengan key string (nama teman) dan value Motor
    (informasi motor)
    map<string, Motor> temanMotor;

    // Menambahkan data teman-teman dan informasi motor mereka ke dalam
    map
    temanMotor["Tegar"] = {"Honda", "Vario", 2019};
    temanMotor["Faiq"] = {"Yamaha", "Aerox", 2016};
    temanMotor["Burhan"] = {"Kawasaki", "zx-25RR", 2023};

    // Menampilkan informasi motor teman-teman
    for (const auto& pair : temanMotor) {
        cout << pair.first << " memiliki motor " << pair.second.merek
        << " " << pair.second.model << " tahun " << pair.second.tahun << endl;
    }

    return 0;
}
```

Screenshots Output

```
PS D:\kuliah\matkul\Semester 2\struktur data\dosen> & 'c:\U
ncher.exe' '--stdin=Microsoft-MIEngine-In-1qdbp0pf.c2v' '--s
Microsoft-MIEngine-Pid-dumue0wi.l4j' '--dbgExe=C:\TDM-GCC-64
Burhan memiliki motor Kawasaki zx-25RR tahun 2023
Faiq memiliki motor Yamaha Aerox tahun 2016
Tegar memiliki motor Honda Vario tahun 2019
PS D:\kuliah\matkul\semester 2\struktur data\dosen>
```

```
Nama : Tegar Serli arunzika
Nim : 2311102041
Kelas If-11_B|
```

Deskripsi

Dalam program di atas, kami menggunakan map dengan key berupa string (nama teman) dan value berupa struct Motor yang berisi informasi mengenai motor (merek, model, tahun). Kemudian, kami menambahkan beberapa entri ke dalam map tersebut yang merepresentasikan teman-teman dan informasi motor mereka

Perbedaan utama antara array dan map adalah:

Array hanya menyimpan elemen dengan tipe data yang sama dan diindeks berurutan, sementara map dapat menyimpan pasangan nilai dan key dengan tipe data yang berbeda dan tidak harus diindeks berurutan. Di dalam array, indeks harus berupa bilangan bulat bukan negatif, tetapi dalam map, key dapat berupa tipe data apa pun yang dapat diurutkan

D. Kesimpulan

Tipe data dalam pemrograman sangat penting untuk mengorganisir dan mengelola data dalam sebuah program. Tipe data primitif seperti int, float, char, dan boolean digunakan untuk menyimpan nilai sederhana dan dapat menjalankan operasi dasar, sedangkan tipe data abstrak dan koleksi memungkinkan pengelompokan data yang lebih kompleks dan terstruktur. Implementasi struktur data seperti array, struct, dan map memiliki karakteristik yang berbeda dan digunakan sesuai dengan kebutuhan program

E. Referensi

Sianipar, R.H. (2012). Pemrograman C++: Dasar Pemrograman Berorientasi Objek (Edisi 1). Penerbit ANDI

Tim Penulis. (2022). Algoritma dan Struktur Data. Penerbit Widina Bhakti Persada Bandung