

**LAPORAN PRAKTIKUM  
STRUKTUR DATA DAN ALGORITMA**

**MODUL I  
TIPE DATA**



**Disusun Oleh :**  
WILDAN DAFFA' HAKIM PUTRA ANTARA  
NIM : 2311102055

**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 klasifikasi nilai suatu variable yang membantu operasi komputer untuk mengenali nilai dari suatu data. Tipe data digunakan variable untuk membatasi data apa yang akan disimpan. Setiap kali variable didefinisikan pada C++, kompiler mengalokasikan memori untuk variable tersebut berdasar pada tipe data yang dipakai.

Tipe data dalam c++ utamanya dibagi menjadi 3 yaitu :

### 1. Tipe data Primitive

Tipe data ini sudah disediakan oleh sistem (predefined), ini dapat digunakan langsung oleh pengguna. Misalnya, ketika mendeklarasikan variable. Tipe data ini juga merupakan tipe data yang paling sederhana.

Contoh tipe data primitif antara lain :

#### a. Interger (int)

Ukuran dari tipe data ini bisa 2 atau 4 bytes. Tipe data ini menyimpan angka tanpa desimal. Range dari tipe data ini adalah - 2147483648 to 2147483647.

#### b. Character (char)

Ukuran dari karakter adalah 1 byte, menyimpan karakter/huruf/number/ASCII value. Range dari tipe data ini adalah - 128 to 127 or 0 to 255.

#### c. Boolean (bool)

Ukuran dari boolean adalah 1 byte, meyimpan boolean atau nilai logic seperti true atau false.

#### d. Floating Point (float)

Ukuran dari float adalah 4 bytes. Menyimpan nilai desimal dengan persisi tunggal. Tingkat presisinya adalah 7 digit desimal.

#### e. Double Floating Point (double)

Ukuran dari double adalah 8 bytes. Meyimpan nilai desimal dengan tingkat presisi dua kali dari float. Tingkat presisinya 15 digit decimal.

#### f. Void atau (tanpa nilai)

Digunakan untuk fungsi yang tidak mengembalikan nilai, biasanya digunakan untuk fungsi yang hanya menjalankan aktivitas namun nilai kembalian nya tidak berguna

### 2. Tipe data Abstrak (*User defined*)

Tipe data ini didefinisikan oleh pengguna sendiri. Biasanya, tipe data ini

berisi banyak tipe data, jadi nilainya bisa lebih dari satu dan beragam tipe data.

Contoh tipe data Abstrak antara lain :

- a. Class  
Class memiliki anggota yang berupa variable dan function yang biasa disebut attribute dan method. Class ini biasa digunakan dalam pemrograman berorientasi objek (OOP). Class sendiri merupakan blueprint dari sebuah objek yang akan dibuat nantinya.
- b. Struct  
Struct merupakan kumpulan variable yang dengan tipe data yang sama ataupun berbeda yang dikelompokkan menjadi satu.

### 3. Tipe Data Koleksi

Tipe data ini digunakan untuk mengelompokkan dan menyimpan objek secara bersamaan. Tipe data ini memungkinkan kita mengelola, menyimpan dan mengakses data secara terstruktur.

Beberapa tipe data yang sering digunakan :

- a. Array  
Merupakan struktur data yang menyimpan kumpulan elemen dengan tipe data yang sama. Elemen yang telah disimpan dapat diakses dengan menggunakan indeksnya. Array mempunyai ukuran tetap yang sudah ditentukan pada waktu deklarasi.
- b. Vector  
Vector digunakan untuk menyimpan elemen dengan tipe data yang sama. Berbeda dengan array, ukuran dari vektor dapat bertambah secara dinamis. Vector merupakan bagian C++ Standard Template Library sehingga ketika menggunakan vector kita harus menambahkan header vector file ke program kita.
- c. Map  
Map juga memiliki kemiripan dengan array, hanya saja index pada map dapat berupa selain integer. Map sendiri menyimpan pasangan data (key - value pair). Nilai key harus unik, sedangkan nilai valuenya tidak.

## B. Guided

### 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;
```

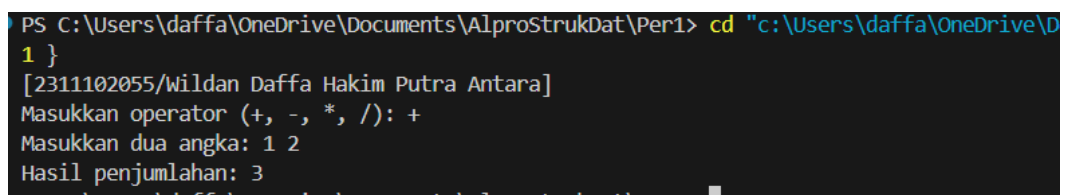
```

        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 C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1> cd "c:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1"
1 }
[2311102055/Wildan Daffa Hakim Putra Antara]
Masukkan operator (+, -, *, /): +
Masukkan dua angka: 1 2
Hasil penjumlahan: 3

```

### Deskripsi:

Program ini merupakan program kalkulator sederhana, yang menerima operator lalu meminta user menginputkan dua angka lalu program akan menampilkan hasil dari operasi yang dilakukan.

Program ini memiliki 3 variable op dengan tipe data char, num1 dan num2 dengan tipe data float. Setelah itu program menampilkan output “Masukkan operator (+, -, \*, /):” lalu inputannya disimpan pada variable op. Setelah itu program menampilkan output “Masukkan dua angka:” dan menerima inputan 2 angka yang akan disimpan pada variable num1 dan num2. Lalu lanjut pada switch case dengan arguments berupa nilai variable op. apabila +,

maka output akan ditambahkan dan ditampilkan pada terminal berlaku juga untuk operator lain seperti -, \*, /. Dan apabila tidak ada operator yang memenuhi maka akan masuk ke case default yang akan menampilkan “Error! Operator tidak benar.”

## 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;
}

```

#### Screenshots Output

```

PS C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1> cd "c:\
2 }
[2311102055/Wildan Daffa Hakim Putra Antara]
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23

```

#### Deskripsi :

Program ini mencetak data mahasiswa yang berisi nama, alamat, dan umur. Program ini menggunakan struct yang diberi nama Mahasiswa. Struct ini berisi name yang bertipe data char, address yang bertipe data char, dan age

berupa interger. Lalu, dideklarasikan variable mhs1, dan mhs2 dengan tipe data struct Mahasiswa. Lalu diinisiasi nilainya dan dicetak menggunakan printf.

### Guided 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 C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1>
3 }
[2311102055/Wildan Daffa Hakim Putra Antara]
Isi array pertama :23
Isi array kedua :50
Isi array ketiga :34
Isi array keempat :78
Isi array kelima :90

```

Deskripsi :

Program ini mencetak nilai array yang memiliki elemen sebanyak 5. Pada awal program dideklarasikan array dengan nama nilai yang memiliki ukuran 5. Lalu dimasukkan nilainya dengan menambahkan langsung kepada index nya mulai dari 0 – 4. Setelah itu, isi dari array dicetak dengan ditambah kalimat “Isi array pertama”, “Isi array kedua”, “Isi array ketiga”, “Isi array keempat”, “Isi array kelima”.

### C. Unguided/Tugas

#### 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;

float keliling_lingkaran(float jari) {
    return 3.14 * 2 * jari;
}

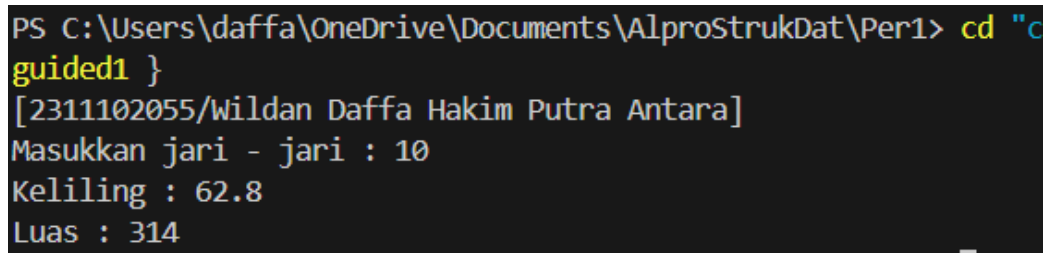
float luas_lingkaran(float jari){
    return 3.14 * jari * jari;
}

int main() {
    float r;
    cout << "Masukkan jari - jari : ";
    cin >> r;
    cout << "Keliling : " << keliling_lingkaran(r)
<< endl;

```

```
    cout << "Luas : " << luas_lingkaran(r) << endl;  
    return 0;  
}
```

#### Screenshots Output



```
PS C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1> cd "c:  
guided1 }  
[2311102055/Wildan Daffa Hakim Putra Antara]  
Masukkan jari - jari : 10  
Keliling : 62.8  
Luas : 314
```

#### Deskripsi:

Program ini merupakan program sederhana untuk menghitung keliling dan luas lingkaran. Pada program ini terdapat dua fungsi. Fungsi pertama bernama `keliling_lingkaran` mempunyai parameter bertipe data float dan memberikan kembalian float juga, fungsi ini akan mengembalikan hasil perkalian  $3.14 \times 2 \times \text{jari}$ . Fungsi kedua merupakan fungsi yang bernama `luas_lingkaran` mempunyai parameter jari yang bertipe data float dan kembalian float juga, fungsi ini akan mengembalikan nilai dari perkalian  $3.14 \times \text{jari} \times \text{jari}$ . Pada fungsi utama (main) terdapat variable `r` yang digunakan untuk menampung inputan user yaitu nilai dari jari – jari lingkaran yang nantinya akan dipakai untuk arguments ketika memanggil fungsi `keliling_lingkaran` dan `luas_lingkaran`. Setelah user menginputkan nilai jari maka program akan memanggil kedua fungsi tadi dan menampilkannya pada output disertai dengan kata “keliling : “ dan “Luas: “.

Kesimpulan : tipe data primitif merupakan tipe data yang paling sederhana dan sudah didefinisikan oleh sistem, tipe data ini memiliki jumlah bit yang dialokasikan tergantung pada bahasa pemrograman, operating system, dan compilernya. Biasanya digunakan untuk operasi aritmatika dan logika dan sebagai fondasi untuk struktur data yang lebih rumit.

#### Unguided 2 :

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

#### **Class**

Pada class terdapat attributes dan method yang sebenarnya adalah variable dan fungsi yang ada di dalam class. Class sendiri merupakan blueprint dari sebuah objek yang akan dibuat. Class biasanya digunakan untuk abstraksi data. Selain itu

class juga mendukung pewarisan. Variable pada class dapat private (hanya bisa diakses dari dalam class sehingga perlu getter dan setter) dan public (bisa diakses dari luar).

### **Struct**

Struct digunakan untuk melakukan grouping pada suatu data. Variable dari struct ini bersifat public dan dapat berisi berbagai macam tipe data.

### **Contoh dari Class :**

```
#include <iostream>

using namespace std;

class Book
{
private:
    bool sudah_dibaca = false;
public:
    int tahun_terbit;
    string penerbit;
    string judul;
    Book(string nama, string pener, int tahun){
        tahun_terbit = tahun;
        penerbit = pener;
        judul = nama;
    };
    string cek_dibaca() {
        if (sudah_dibaca) {
            return "sudah dibaca";
        } else {
```

```

        return "belum dibaca";

    }

}

void tandai_dibaca() {
    sudah_dibaca = true;
}

};

int main() {
    Book buk1("halo dika", "UD. Sinar Jendela",
2024);

    cout << "Nama Buku : " << buk1.judul << endl;
    cout << "Penerbit : " << buk1.penerbit << endl;
    cout << "Tahun : " << buk1.tahun_terbit <<
endl;
    cout << "Status : " << buk1.cek_dibaca() <<
endl;
    buk1.tandai_dibaca();
    cout << "Status : " << buk1.cek_dibaca() <<
endl;
    return 0;
}

```

Output :

```

PS C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1> cd
($?) { .\unguided2_class }
[2311102055/Wildan Daffa Hakim Putra Antara]
Nama Buku : halo dika
Penerbit : UD. Sinar Jendela
Tahun : 2024
Status : belum dibaca
Status : sudah dibaca

```

Deskripsi :

Pada program ini dibuat class bernama buku yang berisi attributes public tahun\_terbit, penerbit, judul dan private attribute yaitu sudah dibaca. Juga terdapat dua method yaitu cek\_dibaca yang mengambil nilai dari attribute private untuk mengecek apakah buku sudah dibaca dan method tandai\_dibaca untuk merubah atau menjadi setter bagi private method menjadi true. Pada fungsi main, object dibuat dengan memanggil class buku1 yang diberi arguments sesuai constructor pada class nya, lalu mencetak attributes judul, penerbit, tahun\_terbit. Lalu, memanggil method cek\_dibaca yang saat itu variable sudah\_dibaca masih bernilai false sehingga menampilkan status belum dibaca. Lalu, kita memanggil fungsi tandai\_dibaca yang akan merubah nilai sudah\_dibaca menjadi true. Sehingga saat memanggil method cek\_dibaca maka statusnya akan berubah menjadi sudah dibaca.

**Contoh dari Struct :**

```
#include <iostream>

using namespace std;

struct Baju
{
    int stok;
    string merk;
    char quality;
};

int main() {

    Baju bajuA;

    bajuA.merk = "Nevada";
```

```

    bajuA.stok = 999;

    bajuA.quality = 'A';

    cout << "Merk : " << bajuA.merk << endl;
    cout << "Stok : " << bajuA.stok << endl;
    cout << "Quality : " << bajuA.quality << endl;
    return 0;
}

```

Output :

```

PS C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1>
if ($?) { .\unguided2_struct }
[2311102055/Wildan Daffa Hakim Putra Antara]
Merk : Nevada
Stok : 999
Quality : A

```

Deskripsi :

Pada program ini dibuat struct bernama Baju yang didalamnya terdiri beberapa variable seperti stok (interger), merk (string), dan quality (character). Lalu pada fungsi main dibuat variable bernama bajuA dengan tipe data struct Baju lalu merk, stok, dan quality diberikan nilai sesuai tipe datanya setelah itu nilai tersebut ditampilkan ke output.

Unguided 3

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

Contoh dari map :

```

#include <iostream>

#include <map>

using namespace std;

```

```
int main()
{
    map<int, string> bulan = {
        {1, "Januari"},
        {2, "Februari"},
        {3, "Maret"},
        {4, "April"},
        {5, "Mei"},
        {6, "Juni"},
        {7, "Juli"},
        {8, "Agustus"},
        {9, "September"},
        {10, "Oktober"},
        {11, "November"},
        {12, "Desember"},
    };

    int bulan_num;

    cout << "Masukkan nomor bulan : ";
    cin >> bulan_num;

    if (bulan_num < 1 || bulan_num > 12) {
        cout << "Tidak ada bulan ke-" << bulan_num
        << endl;

        return 1;
    }

    cout << "Bulan ke - " << bulan_num << " adalah : "
    << bulan[bulan_num] << endl;

    return 0;
}
```

Output :

```
PS C:\Users\daffa\OneDrive\Documents\AlproStrukDat\Per1> cd "
guided3 }
[2311102055/Wildan Daffa Hakim Putra Antara]
Masukkan nomor bulan : 3
Bulan ke - 3 adalah : Maret
```

Deskripsi :

Program ini menampilkan nama bulan sesuai dengan urutan angkanya, pada program ini menggunakan variable bulan dengan tipe data map dimana key nya merupakan tipe data interger dan value nya bertipe data string yang digunakan untuk menampung urutan angka dan nama bulan. Lalu, juga ada variable bulan\_num yang bertipe data interger yang digunakan untuk menampung inputan user. Setelah itu user diminta menginput nomor dari bulan, setelah itu akan dicek apakah nomor tersebut kurang dari 1 atau lebih dari 12. Jika benar, maka akan ditampilkan tidak ada bulan dengan nomor itu. Sedangkan, jika tidak maka akan tampil “bulan ke –“ sesuai nomornya” disertai dengan nama bulannya.

Perbedaan dari array dan map terletak pada key dan banyaknya elemen yang ada didalamnya. Pada array terdapat index sebagai pengenal dari value yang berupa angka. Selain itu, banyaknya elemen harus didefinisikan terlebih dahulu. Sedangkan, pada map menggunakan key yang tidak harus berupa interger namun bisa menggunakan tipe data lain. Tentu, dari segi kecepatan akan lebih cepat array karena dia berupa index, tidak seperti map yang harus mencari key terlebih dahulu. Selain itu, banyaknya elemen pada map dapat ditambah atau dihapus secara dinamis sehingga tidak perlu mendefinisikan jumlahnya terlebih dahulu yang menyebabkan secara efisiensi memori lebih efisien map daripada array.

#### D. Kesimpulan

Tipe data digunakan untuk mengklasifikasi nilai serta memberi tahu kompiller bagaimana data akan digunakan. Tipe data ini dibagi 3 menjadi tipe data primitive yaitu tipe data yang paling sederhana dan sudah didefinisikan oleh sistem. Tipe data Abstrak adalah tipe data yang dibuat oleh user itu sendiri, dan tipe data koleksi yang dipakai untuk mengelompokkan beberapa nilai atau objek.

#### E. Referensi

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

PURBASARI, Wika, et al. ALGORITMA PEMROGRAMAN. 2024.



SUCI WULANDARI. 10 Jenis Tipe Data Pemrograman Populer  
2023. *Dibimbing.id*. Online. 2023.