

**LAPORAN PRAKTIKUM
STRUKTUR DATA DAN ALGORITMA**

**MODUL I
TIPE DATA**



Disusun Oleh :
YASVIN SYAHGANA
NIM : 2311102065

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 atau data untuk memberitahu compiler bagaimana data tersebut akan diolah. Compiler akan mengalokasikan beberapa memori untuk variable itu bergantung pada data apa yang dideklarasikan. Pada umumnya ada 3 jenis tipe data yaitu :

1. Tipe data primitive, yaitu tipe data yang sudah didefinisikan oleh sistem dan dapat langsung digunakan. Contoh : int, char, float double.

2. Tipe data Abstrak

Tipe data ini dibentuk oleh programmer itu sendiri. Biasanya dalam tipe data ini berisi sekumpulan tipe data. Contoh dari tipe data ini adalah class, class sendiri merupakan fitur Object Oriented Programming. Selain itu terdapat juga struct yang isinya juga berbaagai tipe data.

3. Tipe Data Koleksi

Tipe data ini digunakkan untuk mngelompokkan beberapa nilai atau objek secara bersamaan.

Beberapa tipe data yang sering digunakan :

- a. Array

Array ini merupakan tipe data yang statis yang menyimpan nilai dengan tipe data yang sama. ukuran dari array diatur ketika mendekralasikannya. Elemen pada array diakses menggunakan index yang dimulai dari 0.

- b. Vector

Vector pada dasarnya mirip dengan array, hanya saja vector mengalokasikan memorinya secara dinamis dan otomatis. Vector sendiri merupakan Standard Template Library (STL) jika di dalam C/C++ memiliki bentuk `std::vector` .

- c. Map

Map juga mirip dengan array, namun pada key atau index nya bisa berupa tipe data selain interger. Pada `std::map` digunakan Self-Balancing Tree khususnya Red-Black Tree

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 '*':
```

```

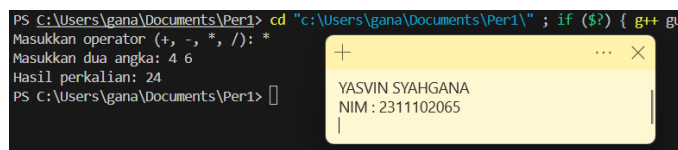
        cout << "Hasil perkalian: " << num1
* num2;

        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



Deskripsi:

Ini adalah kalkulator sederhana yang dibuat dengan c++

Program ini memiliki 3 variable yaitu op berupa character, num1 berupa float, num2 berupa float. Program akan menerima inputan operator matematika dari pengguna. Setelah itu, menerima inputan berupa angka sebanyak 2 angka lalu disimpan pada variabel num1 dan num2.

Program menggunakan switch statement untuk memilih operasi kalkulator

berdasarkan input yang dimasukkan oleh pengguna. Setelah itu, Program menampilkan hasil operasi aritmatika atau pesan operator tidak benar apabila operator tidak valid.

Guided 2

```
#include <stdio.h>

// struktur mahasiswa
struct Mahasiswa
{
    const char *name;
    const char *address;
    int age;
};

int main()
{
    struct Mahasiswa mhs1, mhs2;
    mhs1.name = "Dian";
    mhs1.address = "Mataram";
    mhs1.age = 22;

    mhs2.name = "Bambang";
    mhs2.address = "Surabaya";
    mhs2.age = 23;

    printf("## Mahasiswa 1 ##\n");
    printf("Nama: %s\n", mhs1.name);
    printf("Alamat: %s\n", mhs1.address);
```

```

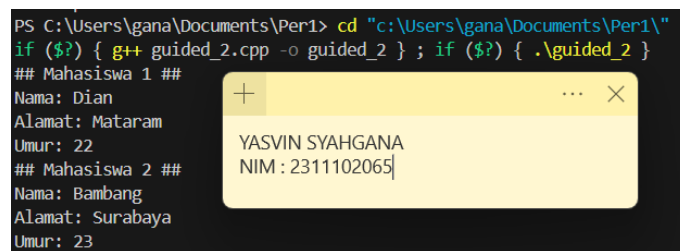
printf("Umur: %d\n", mhs1.age);

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

return 0;
}

```

Screenshots Output



```

PS C:\Users\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1\" ;
if ($?) { g++ guided_2.cpp -o guided_2 } ; if ($?) { .\guided_2 }
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23

```

Deskripsi :

Ini merupakan program untuk menampilkan data mahasiswa, pada program ini menggunakan struct, untuk mengumpulkan data name, address, dan age menjadi satu pada struct Mahasiswa. Di fungsi main variabel mhs1, mhs2 dideklarasikan lalu isinya diinisiasikan dengan name, address, dan age. Setelah itu, 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\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1\" ;
if ($?) { g++ guided3.cpp -o guided3 } ; if ($?) { .\guided3 }
Isi array pertama :23
Isi array kedua :50
Isi array ketiga :34
Isi array keempat :78
Isi array kelima :90
PS C:\Users\gana\Documents\Per1>

```

YASVIN SYAHGANA
NIM : 2311102065

Deskripsi :

Ini merupakan program mencetak array yang berukuran 5. Pada fungsi main, dideklarasikan variabel nilai berupa array yang berukuran 5 dengan tipe data integer. Lalu pada program diberikan nilai dari index 0 sampai 4 dan setelah itu dicetak pada output.

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>

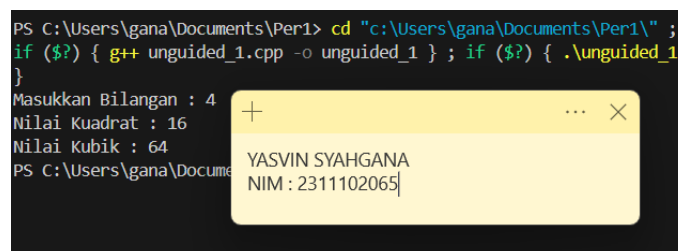
using namespace std;

double kuadrat_bil(double c) {
    double kuadrat = c * c;
    return kuadrat;
}

double kubik_bil(double c) {
    double kubik = c * c * c;
    return kubik;
}

int main() {
    double bill;
    cout << "Masukkan Bilangan : ";
    cin >> bill;
    double kuadrat = kuadrat_bil(bill);
    double kubik = kubik_bil(bill);
    cout << "Nilai Kuadrat : " << kuadrat << endl;
    cout << "Nilai Kubik : " << kubik << endl;
    return 0;
}
```

Screenshots Output



```
PS C:\Users\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1\" ;
if ($?) { g++ unguided_1.cpp -o unguided_1 } ; if ($?) { .\unguided_1
}
Masukkan Bilangan : 4
Nilai Kuadrat : 16
Nilai Kubik : 64
PS C:\Users\gana\Documents\Per1>
```

YASVIN SYAHGANA
NIM : 2311102065

Deskripsi:

Ini merupakan sebuah program yang digunakan untuk menghitung nilai kuadrat dan kubik dari sebuah bilangan yang diinputkan pengguna.

Pertama, dideklarasikan dua fungsi, yaitu `kuadrat_bil` dan `kubik_bil`, yang masing-masing mengembalikan nilai kuadrat dan kubik dari bilangan yang diterima dengan tipe data `double`.

Kemudian, program meminta pengguna untuk memasukkan sebuah bilangan. Bilangan yang telah dimasukkan, disimpan dalam variabel `bill`. Selanjutnya, program memanggil fungsi `kuadrat_bil` dan `kubik_bil` dengan menggunakan bilangan yang dimasukkan oleh pengguna sebagai argumen. Hasil perhitungan kuadrat dan kubik disimpan dalam variabel `kuadrat` dan `kubik` secara berturut-turut. Setelah itu, hasil yang telah disimpan ditampilkan pada output. Dengan ditambah kata “nilai kuadrat : “ untuk `kuadrat_bil` dan “nilai kubik : “ untuk fungsi `kubik_bil`.

Kesimpulan : tipe data primitif digunakan untuk data dasar seperti angka, karakter, dan boolean. Penggunaan tipe data primitif mendukung operasi dasar, struktur kontrol alur program, dan pembentukan struktur data yang lebih kompleks. Pemilihan tipe data yang tepat saat mendeklarasikan variabel menjadi kunci untuk alokasi memori yang efisien dan pemrosesan data yang akurat.

Unguided 2 : Jelaskan fungsi dari `class` dan `struct` secara detail dan berikan contoh programnya

`Class` dan `struct` masuk pada abstract tipe data yang mana isinya ditentukan oleh usernya sendiri. Keduanya digunakan untuk membuat/mendeskripsikan suatu objek.

`Class`: pada `class` terdapat data (variabel) dan `method` (fungsi). Default access pada `class` sendiri adalah `private`, artinya jika objek anggota dari `class` itu sendiri ingin diakses dari luar `class`, maka harus menggunakan `public`. `Class` dapat menyimpan data dan fungsi.

Contoh `class`

```
#include <iostream>

using namespace std;

class Kucing
{
```

```

private:
    int makan = 0;
public:
    string nama;
    int umur;
    int cek_makan () {
        return makan;
    }
};

int main() {
    Kucing miaw2;
    miaw2.nama = "Sicila";
    miaw2.umur = 1;
    cout << "Nama Kucing : " << miaw2.nama <<
endl;
    cout << "Umur Kucing : " << miaw2.umur <<
endl;
    cout << "Kucing Makan sebanyak " <<
miaw2.cek_makan() << " kali" << endl;
    return 0;
}

```

Output :

```

PS C:\Users\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1\" ;
if ($?) { g++ unguided_2_ptclass.cpp -o unguided_2_ptclass } ; if ($?) {
.\unguided_2_ptclass }
Nama Kucing : Sicila
Umur Kucing : 1
Kucing Makan sebanyak 0 kali
PS C:\Users\gana\Documents\Per1>

```

+
 YASVIN SYAHGANA
 NIM : 2311102065

Deskripsi :

Program C++ di atas mendefinisikan sebuah kelas bernama Kucing, Kelas ini memiliki anggota privat makan (menunjukkan jumlah kali kucing tersebut makan) dan anggota publik nama dan umur. Selain itu, terdapat fungsi cek_makan yang mengembalikan nilai dari makan. Di dalam fungsi main, sebuah instance dari kelas Kucing yang diberi nama miaw2 dibuat. Nama dan usia kucing tersebut diatur menjadi “Sicilia” dan 1, dan kemudian informasi tentang kucing, termasuk namanya, umurnya, dan jumlah kali makannya, ditampilkan ke output.

Struct: Struct adalah tipe data yang digunakan untuk menyimpan beberapa data dengan default public. Dengan kata lain, struct melakukan pengelompokan dari beberapa data yang memiliki tipe data berbeda/sama. Struct bermanfaat untuk mengelompokkan sejumlah data dengan tipe yang berlainan.

Contoh Struct :

```

#include <iostream>

using namespace std;

struct DataDiri
{
    string nama;
    int umur;
    string status;
    string hobi;
};

```

```

int main() {
    DataDiri ghana;

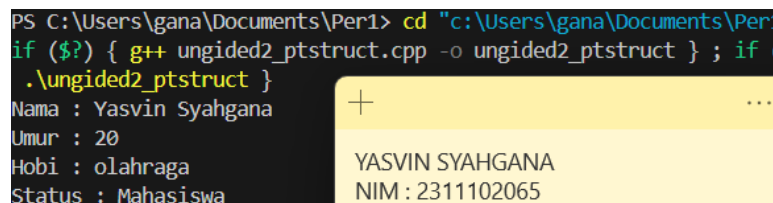
    ghana.nama = "Yasvin Syahgana";
    ghana.umur = 20;
    ghana.hobi = "olahraga";
    ghana.status = "Mahasiswa";

    cout << "Nama : " << ghana.nama << endl;
    cout << "Umur : " << ghana.umur << endl;
    cout << "Hobi : " << ghana.hobi << endl;
    cout << "Status : " << ghana.status << endl;

    return 0;
}

```

Output :



```

PS C:\Users\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1"
if ($?) { g++ ungided2_ptstruct.cpp -o ungided2_ptstruct } ; if (
.\ungided2_ptstruct }
Nama : Yasvin Syahgana
Umur : 20
Hobi : olahraga
Status : Mahasiswa

```

YASVIN SYAHGANA
NIM: 2311102065

Deskripsi :

Program di atas merupakan struktur dengan bernama DataDiri yang didalamnya menyimpan informasi data diri seseorang. Dalam fungsi main, variabel ghana dibuat dengan tipe data struktur DataDiri. Nilai-nilai atribut dari variabel ghana diisi dengan informasi seperti nama, umur, hobi, dan status. Setelah itu, program menampilkan informasi tersebut ke output.

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() {
    map <string, int> pakaian;
    pakaian["gamis"] = 1000;
    pakaian["kemeja"] = 200;
    pakaian["koko"] = 900;
    string pakai;
    cout << "Masukkan Nama Pakaian : ";
    cin >> pakai;
    int harga = pakaian[pakai];
    if (harga == 0) {
        cout << "Pakaian tidak ditemukan " <<
endl;
        return 0;
    }
    cout << "Harganya adalah : " <<
pakaian[pakai] << endl;
    return 0;
}

```

Output :

```

PS C:\Users\gana\Documents\Per1> cd "c:\Users\gana\Documents\Per1\" ;
if ($?) { g++ unguided_3.cpp -o unguided_3 } ; if ($?) { .\unguided_3
}
Masukkan Nama Pakaian : gamis
Harganya adalah : 1000
PS C:\Users\gana\Documents\Per1>

```

+

YASVIN SYAHGANA
NIM : 2311102065

Deskripsi :

Program di atas menggunakan map untuk menyimpan informasi harga pakaian berdasarkan key nya yaitu nama pakaian. Pengguna diminta untuk memasukkan nama pakaian, dan program akan memanggil key sesuai nama pakaian dan akan diambil nilainya. Jika pakaian ditemukan, program akan menampilkan harganya. jika tidak, program akan memberikan pesan bahwa pakaian tidak ditemukan.

Perbedaan :

Array dan map adalah dua struktur data yang digunakan dalam pemrograman untuk menyimpan dan mengelola data. Array adalah kumpulan elemen yang disusun secara linear dengan menggunakan indeks angka. Elemen-elemen ini ditempatkan secara berurutan di dalam memori dan diakses dengan indeks dimulai dari 0. Array memiliki ukuran tetap yang ditentukan saat deklarasi dan tidak dapat diubah selama program berjalan.

Berbeda dengan map, merupakan struktur data asosiatif yang menyimpan elemen dalam pasangan kunci-nilai. Setiap elemen diidentifikasi oleh kunci dan kunci (key) ini dapat bertipe data selain interger, dan nilai yang terkait dapat diakses dengan menggunakan kunci tersebut. Map juga memungkinkan penambahan atau penghapusan elemen, sehingga ukurannya dapat berubah selama program berjalan.

D. Kesimpulan

Tipe data merupakan pengklasifikasian data berdasarkan nilainya, dengan tujuan memberi tahu compiler bagaimana data tersebut akan diolah. tipe data sendiri dibagi menjadi primitif, abstract, dan kolektif yang masing masing punya kegunaan masing – masing. Pada primitif tipe data ini digunakan untuk nilai yang paling sederhana seperti char, interger, double. Pada abstract kita dapat menggabungkan beberapa tipe data dan variable menjadi satu. Sedangkan, pada kolektif kita dapat mengumpulkan beberapa nilai dengan kesamaan tipe data.

E. Referensi

Parewa Labs. (2023). C++ Data Types. <https://www.programiz.com/cpp-programming/data-types>

FIRLIANA, R., & Kasih, P. (2018). Algoritma dan Pemrograman C++.

