

LAPORAN PRAKTIKUM

MODUL I

TIPE DATA



Disusun oleh:

Diva Octaviani

NIM: 2311102006

Dosen Pengampu:

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

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

INSTITUT TEKNOLOGI TELKOM PURWOKERTO

PURWOKERTO

2023

BAB I

TUJUAN PRAKTIKUM

1. Mahasiswa dapat mempelajari tipe data primitif, abstrak, dan kolektif.
2. Mahasiswa dapat memahami pengaplikasian pada tools yang digunakan.
3. Mahasiswa mengaplikasikan berbagai tipe data pada bahasa pemograman yang telah ditentukan.

BAB II

DASAR TEORI

Tipe data adalah adalah sebuah pengklasifikasian data berdasarkan jenis suatu data. Tipe data dibutuhkan agar kompiller dapat mengetahui bagaimana sebuah data akan digunakan. Adapun jenis tipe data diantaranya tipe data primitif, abstrak, dan koleksi.

1) Tipe Data Primitif

Tipe data primitif adalah jenis tipe data yang sudah ditentukan oleh sistem, Tipe data ini disediakan oleh banyak bahasa pemrograman, perbedaannya terletak pada jumlah bit yang dialokasikan untuk setiap bit pada tipe data primitif tergantung pada bahasa pemrograman, kompiller, dan sistem operasinya. Contoh tipe data primitif adalah integer, float, char, boolean.

2) Tipe Data Abstrak

Tipe data abstrak merupakan jenis tipe data yang dibentuk oleh programmer itu sendiri. Tipe data ini 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 struct pada bahasa C. Keduanya berfungsi untuk menampung tipe data di dalamnya sebagai anggota.

3) Tipe Data Koleksi

Tipe data koleksi adalah tipe data yang digunakan untuk mengelompokkan dan menyimpan beberapa nilai atau objek secara bersamaan. Tipe data koleksi memungkinkan untuk menyimpan, mengelola, dan mengakses sejumlah besar data dengan cara yang terstruktur. Beberapa tipe data koleksi yang umum digunakan dalam pemrograman diantaranya array, vector, map.

BAB III

GUIDED

1. Guided 1

Source code

```
#include <iostream>
using namespace std;
// Main program
int main()
{
    char op;
    float num1, num2;
    // It allows user to enter operator i.e. +, -, *, /
    cin >> op;
    // It allow user to enter the operands
    cin >> num1 >> num2;
    // Switch statement begins
    switch (op)
    {
        // If user enter +
        case '+':
            cout << num1 + num2;
            break;
        // If user enter -
        case '-':
            cout << num1 - num2;
            break;
        // If user enter *
        case '*':
            cout << num1 * num2;
            break;
        // If user enter /
        case '/':
```

```

    cout << num1 / num2;
    break;
    // If the operator is other than +, -, * or /,
    // error message will display
    default:
    cout << "Error! operator is not correct";
    } // switch statement ends
    return 0;
}

```

Screenshoot program

```

PS D:\Praktikum Struktur Data\Modul 1> 14 - 10
4

```

Deskripsi program

Program ini memungkinkan pengguna untuk memasukkan operator aritmatika penjumlahan (+), pengurangan (-), perkalian (*), atau pembagian (/). Program menggunakan pernyataan switch untuk memeriksa operator yang dimasukkan dan melakukan perhitungan yang sesuai. Jika pengguna melakukan operator selain yang didukung, pesan kesalahan akan ditampilkan.

2. Guided 2

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

```

PS D:\Praktikum Struktur Data> cd "d:\Praktikum Struktur Data\Modul 1\" ; if ($?) { g++ guided2.cpp -o guided2 } ; if ($?) { .\guided2 }
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23

```

Deskripsi program

Program dibuat untuk menyimpan dan menampilkan informasi mahasiswa. Konsep penting dalam program ini adalah penggunaan struct “Mahasiswa”. Struct ini mendefinisikan sebuah tipe data baru yang dapat menyimpan informasi terkait mahasiswa, seperti nama (berupa string), alamat (berupa string), dan umur (berupa integer). Program menggunakan fungsi printf untuk menunjukkan informasi mengenai dua mahasiswa yang berbeda.

3. Guided 3

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

```
PS D:\Praktikum Struktur Data\Modul 1> cd "d:\Praktikum Struktur Data\Modul 1\" ; 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
```

Deskripsi program

Program tersebut menerapkan penggunaan array. Program ini mendeklarasikan array nilai bertipe integer yang dapat menampung 5 elemen. Nilai-nilai elemen array diisi secara manual, kemudian program mencetak nilai-nilai elemen array satu per satu.

LATIHAN KELAS - 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;

//2311102006 Diva Octaviani

float hitungKeliling(float sisi) {
    return 4 * sisi;
}

float hitungLuas(float sisi) {
    return sisi * sisi;
}

int main() {
    float sisi;

    cout << "Masukkan panjang sisi persegi: ";
    cin >> sisi;

    cout << "Keliling Persegi: " << hitungKeliling(sisi) << endl;
    cout << "Luas Persegi: " << hitungLuas(sisi) << endl;

    return 0;
}
```

Screenshoot program

```
PS D:\Praktikum Struktur Data\Modul 1> cd "d:\Praktikum Struktur Data\Modul 1\" ; if ($?) { g++ unguided1.cpp -o unguided1 } ; if ($?) { .\unguided1 }
Masukkan panjang sisi persegi: 5
Keliling Persegi: 20
Luas Persegi: 25
```

Deskripsi program

Program ini menunjukkan contoh penggunaan tipe data primitif menggunakan fungsi hitungKeliling dan hitungLuas, yang menerima panjang sisi persegi sebagai parameter dan mengembalikan hasil perhitungan keliling dan luas persegi masing-masing. Fungsi hitungKeliling menghitung keliling persegi dengan rumus 4 kali panjang sisi, sedangkan fungsi hitungLuas menghitung luas persegi dengan rumus panjang sisi kali panjang sisi.

2. Unguided 2

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

Source code

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

//2311102006 Diva Octaviani

class Orang {
public:
    string nama;
    int umur;

    void info() {
        cout << "Nama: " << nama << endl;
        cout << "Umur: " << umur << endl;
    }
};
```

```

struct Alamat {
    string jalan;
    string kota;
    int kodePos;
};

int main() {
    Orang p1;
    p1.nama = "Diva";
    p1.umur = 18;
    p1.info();

    Alamat alamat;
    alamat.jalan = "Jl. Sijeruk";
    alamat.kota = "Pemalang";
    alamat.kodePos = 52363;

    cout << "Alamat: " << alamat.jalan << ", " << alamat.kota <<
    ", " << alamat.kodePos << endl;

    return 0;
}

```

Screenshoot program

```

PS D:\Praktikum Struktur Data\Modul 1> cd "d:\Praktikum Str
uktur Data\Modul 1\" ; if ($?) { g++ unguided2.cpp -o ungui
ded2 } ; if ($?) { .\unguided2 }
Nama: Diva
Umur: 18
Alamat: Jl. Sijeruk, Pemalang, 52363

```

Deskripsi program

Class dan struct dalam C++ berfungsi untuk mendefinisikan tipe data baru yang terdiri dari kumpulan variabel (anggota data) dan fungsi (anggota fungsi). Class dan struct memungkinkan kita untuk mengelompokkan data dan fungsi yang terkait, sehingga program menjadi lebih terstruktur dan mudah dipahami.

3. Unguided 3

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

Source code

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

//2311102006 Diva Octaviani

int main() {
    string buah[] = {"Apel", "Pisang", "Jeruk", "Anggur"};

    map<std::string, int> jumlahBuah;
    jumlahBuah["Apel"] = 10;
    jumlahBuah["Pisang"] = 5;
    jumlahBuah["Jeruk"] = 8;
    jumlahBuah["Anggur"] = 3;

    cout << "Jumlah Apel yang tersedia: " << jumlahBuah["Apel"]
    << endl;
    cout << "Jumlah Pisang yang tersedia: " <<
    jumlahBuah["Pisang"] << endl;
    cout << "Jumlah Jeruk yang tersedia: " << jumlahBuah["Jeruk"]
    << endl;
    cout << "Jumlah Anggur yang tersedia: " <<
    jumlahBuah["Anggur"] << endl;

    return 0;
}
```

Screenshoot program

```
PS D:\Praktikum Struktur Data\Modul 1> cd "d:\Praktikum Struktur Data\Modul 1\" ; if ($?) { g++ unguided3.cpp -o unguided3 } ; if ($?) { .\unguided3 }
Jumlah Apel yang tersedia: 10
Jumlah Pisang yang tersedia: 5
Jumlah Jeruk yang tersedia: 8
Jumlah Anggur yang tersedia: 3
```

Deskripsi program

Program ini mendeklarasikan sebuah array string bernama buah yang berisi nama-nama buah. Kemudian, program mendeklarasikan sebuah map jumlahBuah yang memetakan nama buah ke jumlah buah yang tersedia. Selanjutnya, program mengisi map jumlahBuah dengan nama-nama buah dari array buah dan jumlahnya. Secara ringkas, program ini menunjukkan cara menggunakan struktur map untuk menyimpan data dan cara mengakses data dalam map.

Perbedaan array dan map:

1) Array

- Struktur data sederhana yang bisa menyimpan elemen-elemen dengan tipe data yang sama.
- Elemen diakses menggunakan indeks, yang merupakan bilangan bulat non-negatif.
- Penyimpanan elemen dalam array bersifat sekuensial dan ukuran array biasanya tetap setelah deklarasi.

2) Map

- Container asosiatif yang menyimpan pasangan key-value.
- Setiap key harus unik dan digunakan untuk mengakses value yang terkait dengannya.
- Map tidak menyimpan elemen dalam urutan tertentu dan ukurannya bisa berubah secara dinamis.

BAB IV

KESIMPULAN

Dalam praktikum ini, kita telah mempelajari bagaimana mengidentifikasi dan menggunakan berbagai tipe data. Tipe data dalam pemrograman terbagi menjadi primitif, abstrak, dan koleksi. Tipe data primitif seperti integer dan string, abstrak seperti struct dan class, dan koleksi seperti array dan map. Class dan struct mendefinisikan tipe data baru yang terdiri dari kumpulan variabel yang memungkinkan kita untuk mengelompokkan data dan fungsi terkait, sehingga program menjadi lebih terstruktur dan mudah dipahami. Array dan map digunakan untuk menyimpan data. Array memiliki urutan dan diakses dengan indeks, sedangkan map menggunakan pasangan key-value dan diakses dengan key. Pemahaman yang baik tentang konsep-konsep ini sangat penting untuk merancang dan mengimplementasikan solusi pemrograman yang efisien dan efektif.

DAFTAR PUSTAKA

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

Karumanchi, N. (2016). Data Structures and algorithms made easy: Concepts, problems, Interview Questions. CareerMonk Publications.

TylerMSFT. (n.d.). Collections (C++/CX). diakses dari <https://learn.microsoft.com/en-us/cpp/cppcx/collections-c-cx?view=msvc-170>.