

LAPORAN PRAKTIKUM

MODUL I TIPE DATA



Disusun oleh:
Muhammad Rifki Fadhilah
NIM: 2311102032

Dosen Pengampu:
Wahyu Andi Saputra, S.Pd., M.Eng.

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
PURWOKERTO
2024**

BAB I

TUJUAN PRAKTIKUM

1. Memahami apa itu tipe data dan memahami apa saja macam-macam tipe data.
2. Menggunakan tipe data primitif, abstrak, dan kolektif.
3. Memahami perbedaan antara tipe data primitif, abstrak, dan kolektif.

BAB II

DASAR TEORI

Tipe data dalam pemrograman bisa dikelompokkan menjadi beberapa kategori, antara lain:

1. Tipe Data Primitif: Tipe data primitif adalah tipe data yang tersedia secara langsung dalam bahasa pemrograman dan biasanya direpresentasikan oleh tipe data bawaan dari komputer. Contoh tipe data primitif umum meliputi integer, float, double, char, dan boolean.
2. Tipe Data Abstrak: Tipe data abstrak adalah tipe data yang didefinisikan oleh pengguna atau programmer berdasarkan kebutuhan dan tidak tergantung pada representasi fisik di dalam komputer. Contoh tipe data abstrak meliputi array, list, stack, queue, dan tree.
3. Koleksi: Koleksi adalah kumpulan dari beberapa elemen yang terdiri dari satu atau lebih tipe data. Koleksi bisa terdiri dari tipe data primitif atau tipe data abstrak. Contoh koleksi meliputi array, vector, set, map, dan hash table.

Berikut adalah penjelasan singkat tentang masing-masing kategori tipe data:

1. Tipe Data Primitif: Tipe data primitif adalah tipe data dasar yang disediakan oleh bahasa pemrograman untuk merepresentasikan data dasar seperti angka, karakter, atau nilai kebenaran (true/false). Contoh tipe data primitif meliputi:
 - a. int: Untuk menyimpan bilangan bulat.
 - b. float dan double: Untuk menyimpan bilangan pecahan.
 - c. char: Untuk menyimpan karakter.
 - d. bool: Untuk menyimpan nilai kebenaran (true atau false).
2. Tipe Data Abstrak: Tipe data abstrak adalah tipe data yang didefinisikan oleh pengguna atau programmer dan biasanya tidak memiliki implementasi langsung di dalam bahasa pemrograman. Tipe data abstrak ini biasanya digunakan untuk menyembunyikan kompleksitas implementasi dari pengguna. Contoh tipe data abstrak meliputi:

- a. Array: Kumpulan elemen dengan tipe data yang sama yang disusun dalam urutan tertentu.
 - b. List: Kumpulan elemen yang disusun secara linier dan memungkinkan penambahan dan penghapusan elemen.
 - c. Stack dan Queue: Struktur data yang memungkinkan penambahan atau penghapusan elemen sesuai dengan aturan tertentu (Stack menggunakan prinsip LIFO, sedangkan Queue menggunakan prinsip FIFO).
 - d. Tree: Struktur data yang terdiri dari simpul-simpul yang terhubung dan membentuk hirarki.
3. Koleksi: Koleksi adalah tipe data yang dapat menyimpan banyak elemen sekaligus. Koleksi ini bisa terdiri dari tipe data primitif atau tipe data abstrak lainnya. Contoh koleksi meliputi:
- a. Array: Koleksi dari elemen-elemen dengan tipe data yang sama yang disusun dalam urutan tertentu.
 - b. Vector: Koleksi dinamis yang dapat menyesuaikan ukurannya saat elemen-elemen ditambahkan atau dihapus.
 - c. Set: Koleksi yang hanya menyimpan elemen-elemen unik tanpa adanya urutan tertentu.
 - d. Map atau Dictionary: Koleksi yang terdiri dari pasangan kunci-nilai yang memungkinkan pengaksesan nilai berdasarkan kunci tertentu.

BAB III

GUIDED

1. Guided 1

Source code

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

main() {
    char op;
    float num1, num2;
    cin >> op;
    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;
}
```

Screenshoot program



```
guide1.cpp

PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum> cd 'd:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> & .\guide1.exe

+
1 2
3
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>
```

Deskripsi program

Program diatas adalah program menghitung 2 bilangan. Pertama, program mendeklarasikan variabel **op**, **num1**, dan **num2**. Selanjutnya, pengguna diminta memasukkan salah satu operator matematika yaitu: +, -, * atau /. Jika operator yang dimasukkan adalah +(penjumlahan), maka **num1 + num2**. Jika operator yang dimasukkan adalah -(pengurangan), maka **num1 – num2**. Jika operator yang dimasukkan adalah *(perkalian), maka **num1 * num2**. Jika operator yang dimasukkan adalah /(pembagian), maka **num1 / num2**. Jika pengguna memasukkan operator selain 4 operator tersebut maka akan error. Setelah memasukkan operator, tekan enter. Lalu, inputkan bilangan ke-1 dan ke-2. Jika sudah, tekan enter. Dan program akan memunculkan hasilnya.

2. Guided 2

Source code

```
#include <stdio.h>

struct Mahasiswa{
    const char *name;
    const char *address;
```

```

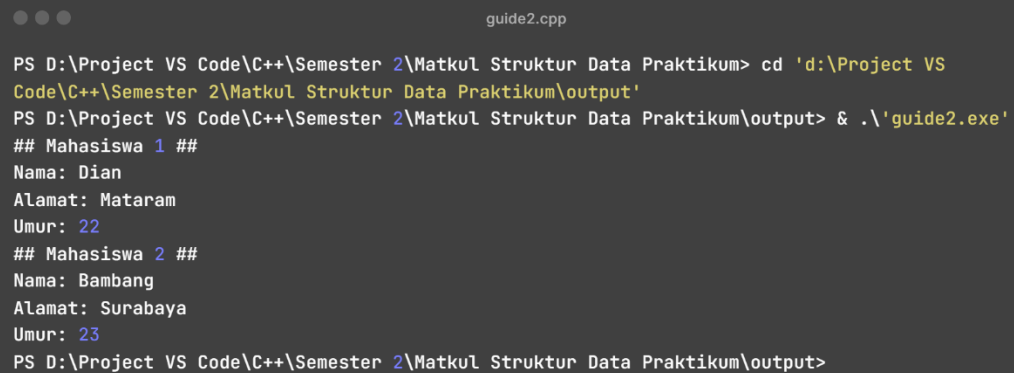
    int age;
};

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);
}

```

Screenshoot program



```

guide2.cpp
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum> cd 'd:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> & .\guide2.exe
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>

```

Deskripsi program

Program diatas adalah program menampilkan data mahasiswa. Pertama, program membuat struct bernama **Mahasiswa** yang berisi **name,address**, dan **age** yang merupakan pointer ke tipe data string. Lalu deklarasikan **mhs1** dan **mhs2** dengan tipe **Mahasiswa**. Kemudian isikan nilai dari masing masing anggota struct. Terakhir, menggunakan fungsi printf untuk menampilkan data mahasiswa, **%s** untuk menampilkan string yang disimpan dalam pointer sedangkan **%d** digunakan untuk menampilkan nilai integer.

3. Guided 3

Source code

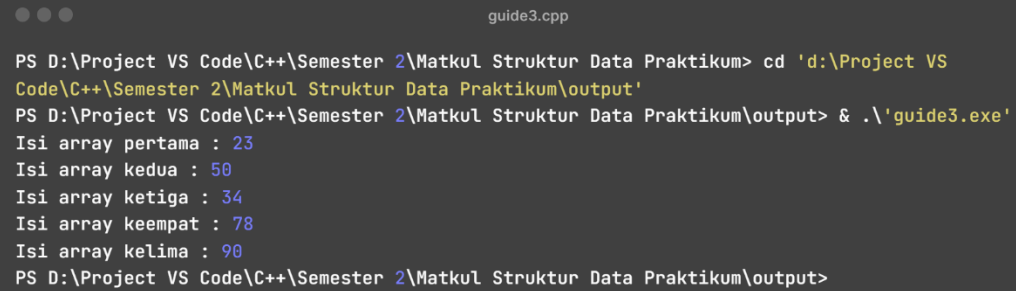
```
#include <iostream>
using namespace std;
int main()
{
    int nilai[5];
    nilai[0] = 23;
    nilai[1] = 50;
    nilai[2] = 34;
    nilai[3] = 78;
    nilai[4] = 90;

    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



```
guide3.cpp
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum> cd 'd:\Project VS
Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> & .\'guide3.exe'
Isi array pertama : 23
Isi array kedua : 50
Isi array ketiga : 34
Isi array keempat : 78
Isi array kelima : 90
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>
```

Deskripsi program

Program diatas adalah program menampilkan isi array pertama sampai kelima. Pertama deklarasikan nilai sebagai array yang berjumlah 5. Isikan array nilai tersebut dengan angka. Tampilkan isi nilai tersebut dengan cout.

LATIHAN KELAS - UNGUIDED

1. Unguided 1

Source code

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

int tambah(int x, int y){
    return x + y;
}

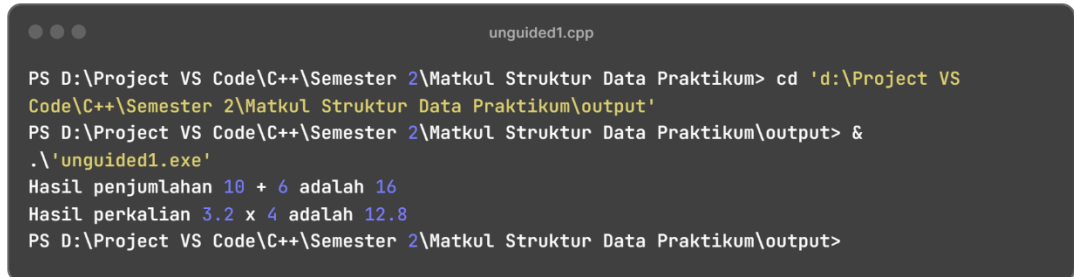
float kali(float m, float n){
    return m * n;
}

int main(){
    int x= 10;
    int y= 6;
    float m= 3.2;
    float n= 4;

    cout<<"Hasil penjumlahan 10 + 6 adalah "<<tambah(x,y)<<endl;
    cout<<"Hasil perkalian 3.2 x 4 adalah "<<kali(m,n);

}
```

Screenshoot program



```
unguided1.cpp
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum> cd 'd:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> &
.\'unguided1.exe'
Hasil penjumlahan 10 + 6 adalah 16
Hasil perkalian 3.2 x 4 adalah 12.8
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>
```

Deskripsi program

Program diatas adalah program menghitung 2 bilangan dalam operasi penjumlahan dan perkalian. Dimulai dari membuat fungsi **tambah** menerima 2 parameter yaitu **x** dan **y**, dan mengembalikan hasil penjumlahan kedua parameter tersebut. Begitu juga, fungsi **kali** menerima 2 parameter yaitu **m** dan **n** dan mengembalikan perkalian kedua parameter tersebut. Setelah itu, deklarasikan variabel **x,y,m,n** dan isikan variabel tersebut dengan angka. Lalu tampilkan hasil penjumlahan dan perkalian menggunakan cout yang diikuti pemanggilan fungsi **tambah** dan **kali** dengan parameter yang sesuai.

Kesimpulan

Tipe data primitif adalah tipe data yang sudah ditentukan oleh sistem, dan jumlah bit yang dialokasikan untuk setiap tipe data primitif tergantung pada bahasa pemrograman, compiler, dan sistem operasinya. Contoh tipe data primitif adalah int, float, char, boolean, dan lain-lain.

2. Unguided 2

Source code

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

// Contoh Struct
struct Mobil{
    string nama,merk, jenis;
    int harga;
};

// Contoh Class
class Pegawai{
public:
    string nama;
    int nip;
};

int main(){

    //Penggunaan Struct
    Mobil toyota_rush, xpander_cross;

    toyota_rush.nama = "Toyota Rush";
    toyota_rush.merk = "Toyota";
    toyota_rush.jenis = "SUV";
    toyota_rush.harga = 282700000;

    xpander_cross.nama = "Xpander Cross";
    xpander_cross.merk = "Xpander";
    xpander_cross.jenis = "Crossover";
    xpander_cross.harga = 316750000;

    cout<<"Nama Mobil : "<< toyota_rush.nama<<endl
```

```

        <<"Merk : "<< toyota_rush.merk<<endl
        <<"Jenis : "<< toyota_rush.jenis<<endl
        <<"Harga : Rp."<< toyota_rush.harga<<endl;

    cout<<endl;

    cout<<"Nama Mobil : "<< xpander_cross.nama<<endl
        <<"Merk : "<< xpander_cross.merk<<endl
        <<"Jenis : "<< xpander_cross.jenis<<endl
        <<"Harga : Rp."<< xpander_cross.harga<<endl;

    cout<<endl;

    // Penggunaan Class
    Pegawai ahmad, sujatmiko;

    ahmad.nama = "Ahmad Basikal";
    ahmad.nip = 2005031;

    sujatmiko.nama = "Sujatmiko Arafuru";
    sujatmiko.nip = 2000103;

    cout<<"Nama Pegawai : "<<ahmad.nama<<endl
        <<"NIP : "<<ahmad.nip<<endl;

    cout<<endl;

    cout<<"Nama Pegawai : "<<sujatmiko.nama<<endl
        <<"NIP : "<<sujatmiko.nip<<endl;
}

```

Screenshoot program

```
unguided2.cpp

PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> cd 'd:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> &
.\'unguided2.exe'
Nama Mobil : Toyota Rush
Merk : Toyota
Jenis : SUV
Harga : Rp.282700000

Nama Mobil : Xpander Cross
Merk : Xpander
Jenis : Crossover
Harga : Rp.316750000

Nama Pegawai : Ahmad Basikal
NIP : 2005031

Nama Pegawai : Sujatmiko Arafuru
NIP : 2000103
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>
```

Deskripsi program

Program diatas adalah program menampilkan nama mobil dan deskripsi dan nama pegawai beserta nipnya. Dimulai dari mendeklarasikan struct **Mobil** yang berisi **nama**,**merk**, dan **jenis**. Selanjutnya di class **Pegawai** diatur menjadi public dan berisi **nama** dan **nip**. Lalu buat objek **toyota_rush** dan **xpander_cross** berdasarkan struct **Mobil** dan isi sesuai atributnya. Dan, buat objek **ahmad** dan **sujatmiko** berdasarkan class **Pegawai** dan isi sesuai atributnya. Tampilkan informasi mobil dan pegawai menggunakan cout.

Fungsi Struct dan Class

Struct adalah tipe data yang digunakan untuk menggabungkan beberapa tipe data menjadi satu kesatuan. Di dalam struct, atribut-atribut dapat didefinisikan merepresentasikan data. Biasanya digunakan untuk mengelompokkan data terkait yang tidak memerlukan operasi atau metode

tambahan. Contohnya adalah struct Mahasiswa(nama,NIM,jenis_kelamin, IPK)

Class adalah tipe data yang dapat digunakan untuk membuat objek-objek yang memiliki variabel dan fungsi yang terkait. Class memungkinkan untuk mengelompokkan data dan perilaku dalam satu kesatuan yang lebih besar. Di dalam class terdapat 2 kata kunci untuk mengatur aksesibilitas dari anggota-anggota class yaitu private dan public. Private hanya dapat diakses dari dalam class sedangkan public bisa diakses dari luar class. Contohnya adalah Class Pegawai(nama,jabatan,gaji, dan NIP).

3. Unguided 3

Source code

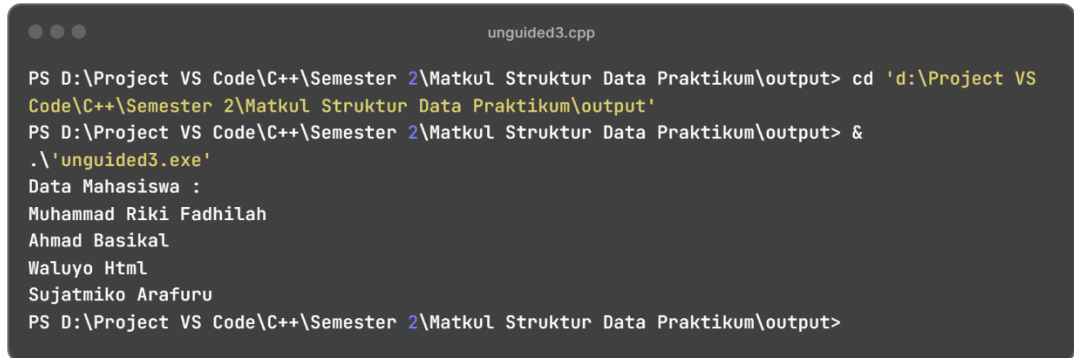
```
#include <iostream>
#include <map>
using namespace std;

int main() {
    map<int, string> mahasiswa;

    mahasiswa[2311102032] = "Muhammad Riki Fadhillah";
    mahasiswa[2311102100] = "Ahmad Basikal";
    mahasiswa[2111104054] = "Waluyo Html";
    mahasiswa[2311102011] = "Sujatmiko Arafuru";

    cout<< "Data Mahasiswa : " << endl
    << mahasiswa[2311102032] << endl
    << mahasiswa[2311102100] << endl
    << mahasiswa[2111104054] << endl
    << mahasiswa[2311102011] << endl;}
```

Screenshoot program



```
unguided3.cpp
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> cd 'd:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output'
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output> &
.\'unguided3.exe'
Data Mahasiswa :
Muhammad Riki Fadhilah
Ahmad Basikal
Waluyo Html
Sujatmiko Arafuru
PS D:\Project VS Code\C++\Semester 2\Matkul Struktur Data Praktikum\output>
```

Deskripsi program

Program diatas adalah program memunculkan data berdasarkan nomor induk mahasiswa. Pertama, impor pustaka **map**. Kemudian, deklarasikan objek **mahasiswa** sebagai map yang memiliki kunci bertipe data int dan nilai bertipe string. Selanjutnya masukkan data mahasiswa ke dalam map mahasiswa. Kunci disini adalah **nomor induk mahasiswa** yang menggunakan operator [], dan nilai adalah **nama mahasiswa**. Lalu, tampilkan data mahasiswa yang tersimpan dalam map mahasiswa.

Perbedaan Array dan Map

Dimulai dari array, array digunakan untuk menyimpan sekumpulan data dengan tipe data yang sama. Indeks array dimulai dari 0, dan untuk mengakses elemennya, menggunakan [] dengan indeks sebagai parameter. Sedangkan map, map digunakan untuk menyimpan pasangan kunci-nilai. Kunci dan nilai pada map dapat berupa tipe data apapun. Untuk mengaksesnya nilai pada map, menggunakan [] dengan kunci sebagai parameter.

BAB IV

KESIMPULAN

Dalam pemrograman, tipe data dapat dikelompokkan menjadi tipe data primitif, tipe data abstrak, dan koleksi. Tipe data primitif adalah tipe data dasar yang disediakan oleh bahasa pemrograman, seperti integer, float, char, dan bool. Tipe data abstrak adalah tipe data yang didefinisikan oleh pengguna atau programmer, seperti array, list, stack, queue, dan tree. Sedangkan koleksi adalah kumpulan dari beberapa elemen yang terdiri dari satu atau lebih tipe data, seperti array, vector, set, map, dan hash table.

Praktik pengkodean yang menggabungkan konsep tersebut dapat dilakukan dengan membuat program sederhana yang menggunakan map, struct, class, dan tipe data primitif. Sebagai contoh, kita dapat membuat program untuk mengelola data mahasiswa. Kita dapat menggunakan struct untuk merepresentasikan data mahasiswa (seperti nama, nim, dan jurusan), class untuk mengelola data mahasiswa (seperti menambah, menghapus, atau mencari mahasiswa), dan map untuk menyimpan data mahasiswa berdasarkan nim sebagai kunci. Dengan kombinasi konsep ini, kita dapat membuat program yang efisien dan mudah diorganisir.