

# **LAPORAN PRAKTIKUM**

## **MODUL I “TIPE DATA”**



**Disusun oleh:**

**Muhammad Dani Ayubi**

**2311102003**

**S1 IF-11-A**

**PROGRAM STUDI S1 INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

**PURWOKERTO**

**2024**

# **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 pemrograman yang telah ditentukan.

## **BAB II**

### **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**

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.

#### **2) 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. perbedaan antara Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private.

#### **3) 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". Pada `std::map` digunakan Self-Balancing Tree khususnya Red-Black Tree.

Perbedaan antara tipe data primitif, tipe data abstrak, dan tipe data koleksi terletak pada kompleksitas strukturnya dan operasi yang terkait. Tipe data primitif hanya memiliki fitur sederhana, sedangkan tipe data abstrak dan tipe data koleksi memiliki struktur dan operasi yang lebih kompleks. Selain itu, tipe data koleksi memungkinkan untuk menyimpan banyak data dalam satu variabel, sedangkan tipe data primitif dan tipe data abstrak hanya menyimpan satu data dalam satu variabel.

## BAB III

### GUIDED

#### 1. Guided 1

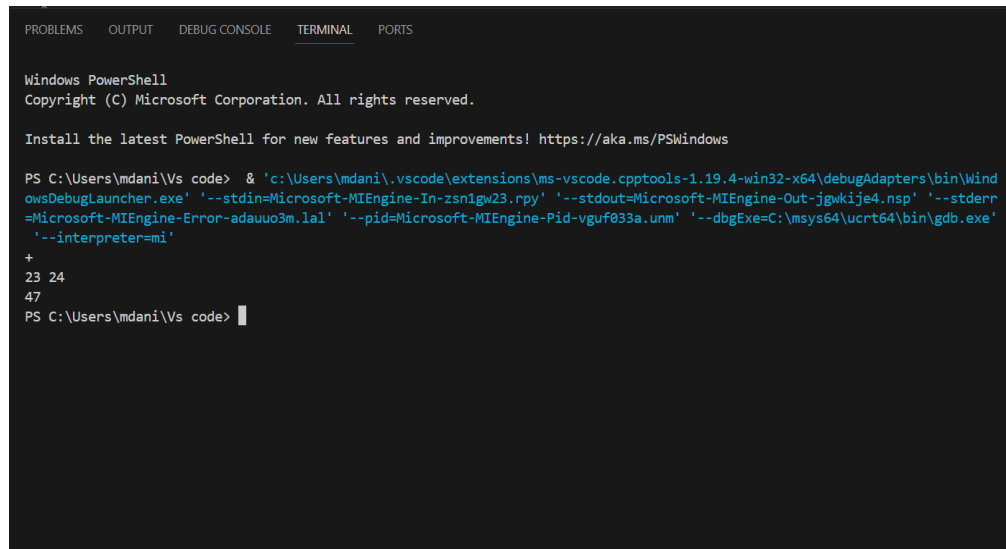
##### Source code

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

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



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-zsn1gw23.rpy' '--stdout=Microsoft-MIEngine-Out-jgwki4.nsp' '--stderr=Microsoft-MIEngine-Error-adauuo3m.lal' '--pid=Microsoft-MIEngine-Pid-vguf033a.unm' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
+
23 24
47
PS C:\Users\mdani\Vs code> |
```

## Deskripsi program

Pada program di atas, setelah pengguna memasukkan operator dan dua bilangan, program akan melakukan operasi aritmatika sesuai dengan operator yang dimasukkan oleh pengguna menggunakan konsep switch case. Jika operator yang dimasukkan oleh pengguna bukan +, -, \*, atau /, maka program akan menampilkan pesan kesalahan "Error! operator is not correct". Program kemudian akan mengembalikan nilai 0 untuk menandakan bahwa program telah berakhir dengan sukses.

## 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" ;
    mhs1.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

```

## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 23
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 8
PS C:\Users\mdani\Vs code>

```

## Deskripsi program

Dekripsi Program Program di atas adalah sebuah program dalam bahasa C yang menggunakan struktur (struct) untuk menyimpan data mahasiswa. Setiap variabel dalam struktur Mahasiswa (name, address, age) digunakan untuk menyimpan data yang terkait dengan mahasiswa tersebut. Struct Mahasiswa: merupakan struktur yang terdiri dari tiga variabel yaitu name, address, dan age yang masing-masing digunakan untuk menyimpan nama, alamat, dan umur dari mahasiswa. Variabel mhs1 dan mhs2: merupakan dua variabel dengan tipe struct Mahasiswa. Variabel ini digunakan untuk menyimpan data dari dua mahasiswa yang berbeda. Pengisian data mahasiswa: nilai dari variabel name, address, dan age pada masing masing variabel mhs1 dan mhs2 diisi dengan menggunakan operator assignment (=) yang digunakan untuk mengisi nilai pada variabel. Menampilkan data mahasiswa: nilai dari variabel mhs1 dan mhs2 kemudian ditampilkan pada layar dengan menggunakan fungsi printf(). Dalam fungsi printf(), kita menggunakan format specifier (seperti %s untuk string, dan %d untuk integer) untuk menampilkan nilai dari masing-masing variabel yang ada pada struktur Mahasiswa. Pada akhir program, nilai kembali (return) yang diberikan adalah 0, yang menunjukkan bahwa program telah selesai dieksekusi dengan sukses.

### 3. Guided Source code

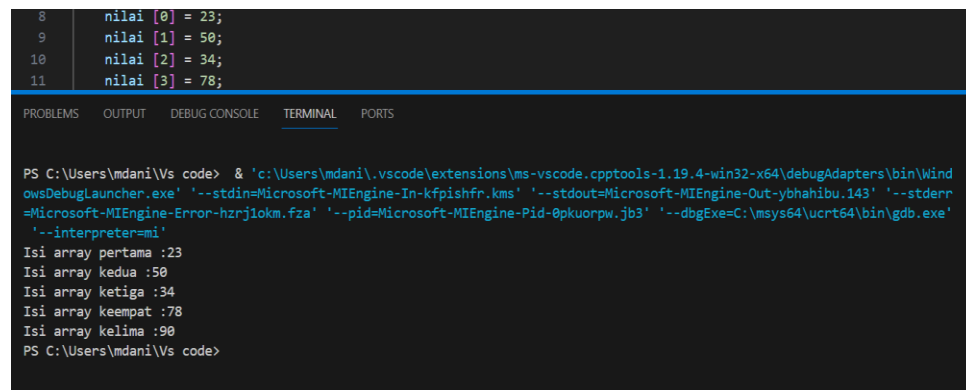
```
#include <iostream>
using namespace std;

int main ()
{

    int nilai [5] ;
    nilai [0] = 23;
    nilai [1] = 50;
    nilai [2] = 34;
    nilai [3] = 78;
    nilai [4] = 90;

    //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



```
8      nilai [0] = 23;
9      nilai [1] = 50;
10     nilai [2] = 34;
11     nilai [3] = 78;

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-kfpishfr.kms' '--stdout=Microsoft-MIEngine-Out-ybhahibu.143' '--stderr=Microsoft-MIEngine-Error-hzrjlok.m.fza' '--pid=Microsoft-MIEngine-Pid-0pkurpw.jb3' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe'
'--interpreter=mi'
Isi array pertama :23
Isi array kedua :50
Isi array ketiga :34
Isi array keempat :78
Isi array kelima :90
PS C:\Users\mdani\Vs code>
```

### Deskripsi program

Pengisian nilai pada array: nilai pada masing-masing elemen array diisi dengan menggunakan operator assignment (=) yang digunakan untuk mengisi nilai pada variabel. Menampilkan nilai array: nilai pada masing-masing elemen array kemudian ditampilkan pada layar dengan menggunakan fungsi cout<< Dalam fungsi cout<<, kita menggunakan nama array dengan index dari masing-masing elemen array untuk menampilkan nilai pada elemen tersebut. Pada akhir program, nilai kembali (return) yang diberikan adalah 0, yang menunjukkan bahwa program telah selesai dieksekusi dengan sukses.



# LATIHAN KELAS - UNGUIDED

## 1. Unguided 1

### Source code

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

#include <iostream>
using namespace std;

// Fungsi penjumlahan
int tambah(int a, int b) {
    return a + b;
}

// Fungsi pengurangan
int kurang(int a, int b) {
    return a - b;
}

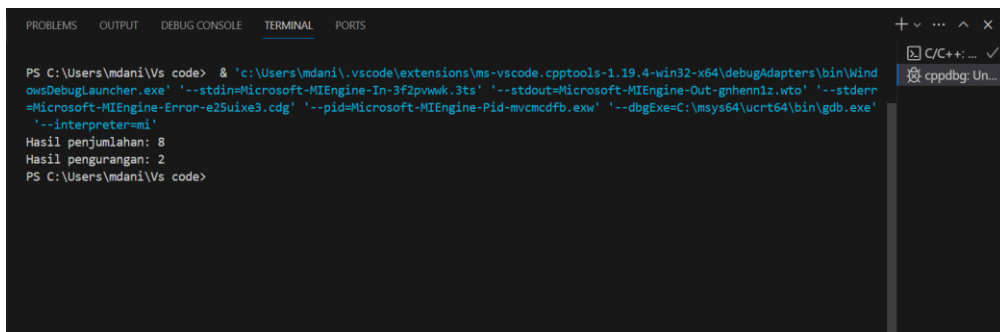
int main() {
    int nilai1 = 5;
    int nilai2 = 3;

    // Menampilkan hasil penjumlahan
    cout << "Hasil penjumlahan: " << tambah(nilai1, nilai2)
    << endl;

    // Menampilkan hasil pengurangan
    cout << "Hasil pengurangan: " << kurang(nilai1, nilai2)
    << endl;

    return 0;
}
```

### Screenshoot program



The screenshot shows a terminal window with the following output:

```
PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-3f2pwwk.3ts' '--stdout=Microsoft-MIEngine-Out-gnhenn1z.wto' '--stderr=Microsoft-MIEngine-Error-e25uixe3.cdg' '--pid=Microsoft-MIEngine-Pid-mvcmdfb.exe' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe'
'--interpreter=mi'
Hasil penjumlahan: 8
Hasil pengurangan: 2
PS C:\Users\mdani\Vs code>
```

The terminal window also shows the file explorer with the file 'C/C++: Un...' and the output 'cppdbg: Un...'.

## Deskripsi program

Fungsi pertama adalah tambah yang memiliki dua parameter a dan b dengan tipe data int. Fungsi ini mengembalikan hasil penjumlahan dari kedua parameter tersebut. Fungsi kedua adalah kurang yang juga memiliki dua parameter a dan b dengan tipe data int. Fungsi ini mengembalikan hasil pengurangan dari kedua parameter tersebut. Pada fungsi main, terdapat deklarasi variabel nilai1 dan nilai2 dengan tipe data int. Kemudian, program menampilkan hasil penjumlahan dari variabel nilai1 dan nilai2 dengan memanggil fungsi tambah dan menampilkan hasilnya dengan menggunakan perintah cout. Selanjutnya, program menampilkan hasil pengurangan dari variabel nilai1 dan nilai2 dengan memanggil fungsi kurang dan menampilkan hasilnya dengan menggunakan perintah cout. Program kemudian mengembalikan nilai 0 karena program berakhir dengan sukses.

## 2. Guided 2

### - Class Class

adalah struktur data yang memungkinkan pengguna untuk menggabungkan data dan fungsi-fungsi yang beroperasi pada data tersebut ke dalam sebuah unit yang terpisah dari program utama. Class adalah blueprint atau cetak biru (template) untuk membuat objek-objek yang serupa. Dalam class, terdapat variabel (data member) dan fungsi (method) yang memungkinkan data tersebut diakses dan dimodifikasi oleh fungsi fungsi tertentu. Source Code #include

### Source code

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

class Mahasiswa {
private:
    string nama;
    int umur;

public:
    Mahasiswa(string nm, int um) {
        nama = nm;
        umur = um;
    }

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

```
int main() {
    Mahasiswa mhs1("Muhammad Dani Ayubi", 20);
    Mahasiswa mhs2("Nandyra Dwi Azzahra", 19);

    mhs1.display();
    mhs2.display();
    return 0;
}
```

## Screenshoot program

```
PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-jvkck01f.a4o' '--stdout=Microsoft-MIEngine-Out-zsfkml14.drs' '--stderr=Microsoft-MIEngine-Error-4i02n15s.sd2' '--pid=Microsoft-MIEngine-Pid-02jebjob.fyt' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Nama: Muhammad Dani Ayubi
Umur: 20
Nama: Nandyra Dwi Azzahra
Umur: 19
PS C:\Users\mdani\Vs code> █
```

## Dekripsi Program :

Pada program di atas, terdapat class Mahasiswa yang memiliki dua data member yaitu nama dan umur, serta empat fungsi member yaitu setName(), getName(), setUmur(), dan getUmur(). Mahasiswa digunakan untuk merepresentasikan data mahasiswa yang memiliki nama dan umur. Kemudian, pada fungsi main(), objek mhs dibuat dengan menggunakan constructor default. Objek mhs kemudian diisi dengan data menggunakan fungsi setName() dan setUmur(). Data pada objek mhs dapat diambil dengan menggunakan fungsi getName() dan getUmur() yang kemudian ditampilkan di layar.

## - Struct

(structure) pada dasarnya mirip dengan class, namun struktur data ini berisi hanya variabel (data member) saja tanpa adanya fungsi (method). Struct digunakan untuk mengelompokkan beberapa variabel yang berhubungan menjadi satu kesatuan. Struct pada dasarnya merupakan tipe data gabungan atau tipe data turunan yang terdiri dari beberapa tipe data primitif.

### Source code

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

struct Mahasiswa {
    string nama;
    string jurusan;
    int angkatan;
};

int main() {
    // Deklarasi variabel mahasiswa dengan tipe struct
    Mahasiswa
    Mahasiswa mahasiswa1;

    // Mengisi data ke dalam variabel mahasiswa1
    mahasiswa1.nama = "MUHAMMAD DANI AYUBI";
    mahasiswa1.jurusan = "Informatika";
    mahasiswa1.angkatan = 2023;

    // Menampilkan data dari variabel mahasiswa1
    cout << "Data Mahasiswa 1" << endl;
    cout << "Nama: " << mahasiswa1.nama << endl;
    cout << "Jurusan: " << mahasiswa1.jurusan << endl;
    cout << "Angkatan: " << mahasiswa1.angkatan << endl;

    return 0;
}
```

### Screenshoot program

```
PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-45xc4wro.owo' '--stdout=Microsoft-MIEngine-Out-jthbb5kv.h2u' '--stderr=Microsoft-MIEngine-Error-npfxlmom.ccw' '--pid=Microsoft-MIEngine-Pid-gzp0or40.2du' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe'
'--interpreter=mi'
Data Mahasiswa 1
Nama: MUHAMMAD DANI AYUBI
Jurusan: Informatika
Angkatan: 2023
PS C:\Users\mdani\Vs code>
```

## Deskripsi program

Program di atas merupakan contoh sederhana penggunaan struct untuk menyimpan informasi mahasiswa. Terdapat tiga variabel pada struct Mahasiswa yaitu nama, jurusan, dan angkatan.

### 3. Unguided 3 Source code

```
#include <iostream>
#include <map>
using namespace std;
int main()
{
    map<string, int> myMap;

    myMap["Dani"] = 18;
    myMap["Agus"] = 20;
    myMap["Alim"] = 19;

    for (auto it = myMap.begin(); it != myMap.end(); it++) {
        cout << it->first << " : " << it->second << endl;
    }

    cout << "Umur Dani: " << myMap["Dani"] << endl;
    return 0;
}
```

## Screenshoot program

```
PS C:\Users\mdani\Vs code> & 'c:\Users\mdani\.vscode\extensions\ms-vscode.cpptools-1.19.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-0tzf5dul.d0r' '--stdout=Microsoft-MIEngine-Out-33tgmIys.1zn' '--stderr=Microsoft-MIEngine-Error-afocrspk.z4x' '--pid=Microsoft-MIEngine-Pid-r3rh5r41.nki' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Agus : 20
Alim : 19
Dani : 18
Umur Dani: 18
PS C:\Users\mdani\Vs code>
```

## Deskripsi program

Program di atas menggunakan fungsi map pada C++. Fungsi ini digunakan untuk membuat sebuah asosiasi antara suatu nilai (yang disebut key) dengan nilai lainnya (yang disebut value). Dalam contoh di atas, kita membuat sebuah objek myMap yang menggunakan string sebagai kunci (key) dan int sebagai nilai (value).

### **Perbedaan antara array dan map**

terletak pada cara pengaksesan datanya. Dalam array, data diakses melalui indeks yang dimulai dari 0, sedangkan dalam map, data diakses melalui kunci yang ditentukan saat membuat objek map. Selain itu, map juga memiliki kelebihan dalam hal efisiensi karena ukuran map dapat disesuaikan secara dinamis, sedangkan array memiliki ukuran yang tetap dan harus ditentukan sebelum penggunaannya. Namun, kelemahan dari map adalah pengaksesan datanya membutuhkan waktu yang lebih lama dibandingkan dengan array karena harus melakukan pencarian kunci terlebih dahulu.

## **BAB IV**

### **Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa:

1. Tipe data primitif adalah tipe data bawaan dari bahasa pemrograman yang memiliki nilai dasar atau default, seperti int, float, dan char.
2. Tipe data abstrak adalah tipe data yang tidak memiliki nilai dasar atau default, dan perlu didefinisikan secara eksplisit, seperti class dan struct.
3. Tipe data koleksi adalah tipe data yang dapat menyimpan beberapa nilai dalam satu variabel, seperti array dan map.
4. Array adalah tipe data koleksi yang menyimpan nilai dengan tipe data yang sama, dan memiliki indeks untuk mengakses setiap elemen.
5. Map adalah tipe data koleksi yang menyimpan pasangan key-value, dan dapat diakses dengan menggunakan key.
6. Penggunaan tipe data primitif, tipe data abstrak, dan tipe data koleksi akan sangat mempengaruhi struktur program dan efisiensi penggunaan memori pada program.
7. Pemilihan tipe data yang tepat juga akan memudahkan dalam pengolahan data pada program, dan menghindari terjadinya error atau bug pada program.