

# **LAPORAN PRAKTIKUM**

## **MODUL I TIPE DATA**



**Disusun oleh:**  
**Ilhan Sahal Mansiz**  
**NIM: 2311102029**

**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 pemrograman yang telah ditentukan.

## **BAB II**

### **DASAR TEORI**

Tipe data adalah adalah sebuah pengklasifikasian data berdasarkan jenis data tersebut. Tipe data dibutuhkan agar kompiler dapat mengetahui bagaimana sebuah data akan digunakan. Adapun tipe data yang akan dipelajari, sebagai berikut :

#### **1. Tipe Data Primitif:**

Tipe data primitif merupakan tipe data yang merupakan bagian terkecil dari tipe data dalam bahasa pemrograman. Mereka biasanya diimplementasikan langsung oleh bahasa pemrograman dan digunakan untuk menyimpan nilai sederhana seperti angka atau karakter. Contoh tipe data primitif termasuk integer, floating-point numbers, karakter, dan boolean. Tipe data primitif biasanya memiliki ukuran yang tetap dan operasi yang sederhana.

#### **2. Tipe Data Abstrak:**

Tipe data abstrak (ADT - Abstract Data Type) adalah sebuah tipe data yang didefinisikan oleh pengguna atau bahasa pemrograman untuk menyimpan koleksi nilai dan operasi-operasi yang dapat dilakukan terhadap koleksi tersebut. Ini merupakan representasi konseptual dari sebuah tipe data dan sering kali dibangun di atas tipe data primitif. Contoh tipe data abstrak termasuk tipe data seperti array, string, dan queue. Tipe data abstrak memungkinkan pengguna untuk menyembunyikan detail implementasi dan fokus pada penggunaan fungsionalitas yang diberikan.

#### **3. Tipe Data Koleksi:**

Tipe data koleksi adalah tipe data yang digunakan untuk menyimpan kumpulan data dalam satu struktur. Mereka digunakan untuk menyimpan, mengelola, dan mengakses banyak data dalam satu waktu. Contoh tipe data koleksi termasuk array, list, set, map (dictionary), dan tuple. Tipe data koleksi ini biasanya memiliki operasi-operasi tertentu yang memungkinkan pengguna untuk memanipulasi elemen-elemen yang disimpan di dalamnya.

## BAB III

### GUIDED

#### 1. Guided 1

##### Source code

```
#include <iostream>
using namespace std;
// main program
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

```
1  #include <iostream>
2  using namespace std;
3  // main program
4  main()
5  {
6      char op;
7      float num1, num2;
8      cin >> op;
9      cin >> num1 >> num2;
10     switch (op)
11     {
12         case '+':
13             cout << num1 + num2;
14             break;
15         case '-':
16             cout << num1 - num2;
17             break;
18         case '*':
19             cout << num1 * num2;
20             break;
21         case '/':
22             cout << num1 / num2;
23             break;
24         default:
25             cout << "Error! operator is not correct";
26     }
27     return 0;
28 }
```

PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE   PORTS

```
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd "c:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1"
ur_data_guided1 } ; if ($?) { .\struktur_data_guided1 }
+
5 10
15
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> 
```

**Deskripsi program**

Program tersebut merupakan sebuah program sederhana dalam bahasa C++ yang melakukan operasi aritmatika dasar (penjumlahan, pengurangan, perkalian, dan pembagian) antara dua bilangan. Program ini menerima input operator dan dua bilangan dari pengguna, kemudian melakukan operasi sesuai dengan operator yang dimasukkan. Jika operator tidak valid, program akan menampilkan pesan kesalahan. Program tersebut menggunakan struktur kontrol switch-case untuk menentukan operasi yang akan dilakukan berdasarkan operator yang dimasukkan oleh pengguna.

## 2. Guided 2

### Surce Code

```
#include <stdio.h>

// struct
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;
}
```

## Screenshot Program

```
1  #include <stdio.h>
2
3  // struct
4  struct Mahasiswa
5  {
6      const char *name;
7      const char *address;
8      int age;
9  };
10
11 int main()
12 {
13     struct Mahasiswa mhs1, mhs2;
14     mhs1.name = "Dian";
15     mhs1.address = "Mataram";
16     mhs1.age = 22;
17     mhs2.name = "Bambang";
18     mhs2.address = "Surabaya";
19     mhs2.age = 23;
20
21
22     printf("## Mahasiswa 1 ##\n");
23     printf("Nama: %s\n", mhs1.name);
24     printf("Alamat: %s\n", mhs1.address);
25     printf("Umur: %d\n", mhs1.age);
26     printf("## Mahasiswa 2 ##\n");
27     printf("Nama: %s\n", mhs2.name);
28     printf("Alamat: %s\n", mhs2.address);
29     printf("Umur: %d\n", mhs2.age);
30     return 0;
31 }
32
```



```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> 
```

### Deskripsi Program

Program tersebut merupakan sebuah program dalam bahasa C yang menggunakan struktur struct untuk merepresentasikan data mahasiswa. Setiap objek Mahasiswa memiliki tiga atribut: nama, alamat, dan umur. Pada fungsi main(), dua objek mahasiswa (mhs1 dan mhs2) dibuat dan diinisialisasi dengan nilai-nilai yang sesuai. Setelah itu, program mencetak informasi masing-masing mahasiswa ke layar menggunakan fungsi printf(). Setelah menyelesaikan pencetakan, program mengembalikan nilai 0 yang menandakan bahwa program telah berakhir dengan sukses.

### 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;
}
```

## Screenshot Program

```
struktur_data_guided3.cpp > main()
1   #include <iostream>
2   using namespace std;
3   int main()
4   {
5       int nilai[5];
6       nilai[0] = 23;
7       nilai[1] = 50;
8       nilai[2] = 34;
9       nilai[3] = 78;
10      nilai[4] = 90;
11
12      cout << "Isi array pertama :" << nilai[0] << endl;
13      cout << "Isi array kedua :" << nilai[1] << endl;
14      cout << "Isi array ketiga :" << nilai[2] << endl;
15      cout << "Isi array keempat :" << nilai[3] << endl;
16      cout << "Isi array kelima :" << nilai[4] << endl;
17      return 0;
18  }
```

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd "c:\
ur_data_guided3 } ; if ($?) { .\struktur_data_guided3 }
Isi array pertama :23
Isi array kedua :50
Isi array ketiga :34
Isi array keempat :78
Isi array kelima :90
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1>
```

## **Deskripsi Program**

Program di atas adalah contoh penggunaan array dalam bahasa C++. Array nilai dideklarasikan sebagai sebuah array berukuran 5 elemen bertipe integer. Kemudian, setiap elemen dari array tersebut diisi dengan nilai tertentu. Setelah itu, program mencetak isi dari setiap elemen array ke layar menggunakan perintah cout.

## 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;

float hitungLuasSegitiga(float alas, float tinggi) {
    return 0.5 * alas * tinggi;
}

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

int main() {
    float alasSegitiga, tinggiSegitiga, sisiPersegi;

    cout << "Masukkan panjang alas segitiga: ";
    cin >> alasSegitiga;

    cout << "Masukkan tinggi segitiga: ";
    cin >> tinggiSegitiga;

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

    cout << "Luas segitiga dengan alas " << alasSegitiga << "
dan tinggi " << tinggiSegitiga << " adalah: " <<
hitungLuasSegitiga(alasSegitiga, tinggiSegitiga) << endl;
```

```

        cout << "Keliling persegi dengan sisi " << sisiPersegi << "
adalah: " << hitungKelilingPersegi(sisiPersegi) << endl;

        return 0;
    }

```

## Screenshoot

```

1  #include <iostream>
2  using namespace std;
3
4  float hitungLuasSegitiga(float alas, float tinggi) {
5      return 0.5 * alas * tinggi;
6  }
7
8  float hitungKelilingPersegi(float sisi) {
9      return 4 * sisi;
10 }
11
12 int main() {
13     float alasSegitiga, tinggiSegitiga, sisiPersegi;
14
15     cout << "Masukkan panjang alas segitiga: ";
16     cin >> alasSegitiga;
17
18     cout << "Masukkan tinggi segitiga: ";
19     cin >> tinggiSegitiga;
20
21     cout << "Masukkan panjang sisi persegi: ";
22     cin >> sisiPersegi;
23
24     cout << "Luas segitiga dengan alas " << alasSegitiga << " dan tinggi " << tinggiSegitiga << " adalah: " << hitungLuasSegitiga(alasSegitiga,
tinggiSegitiga) << endl;
25     cout << "Keliling persegi dengan sisi " << sisiPersegi << " adalah: " << hitungKelilingPersegi(sisiPersegi) << endl;
26
27     return 0;
28 }

```

```

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd "c:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1" & .\strukdat_unguided1 } ; if ($?) { .\strukdat_unguided1 }
Masukkan panjang alas segitiga: 5
Masukkan tinggi segitiga: 7
Masukkan panjang sisi persegi: 6
Luas segitiga dengan alas 5 dan tinggi 7 adalah: 17.5
Keliling persegi dengan sisi 6 adalah: 24
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1>

```

## Deskripsi program

- Program ini memiliki dua fungsi, yaitu `hitungLuasSegitiga()` dan `hitungKelilingPersegi()`, yang digunakan untuk menghitung luas segitiga dan keliling persegi, masing-masing.

- Di dalam fungsi-fungsi tersebut, dilakukan operasi sederhana dengan tipe data primitif, yaitu float untuk menyimpan alas, tinggi, sisi, dan hasil perhitungan.
- Pada bagian main(), program meminta pengguna memasukkan panjang alas segitiga, tinggi segitiga, dan panjang sisi persegi.
- Kemudian, program memanggil fungsi-fungsi tersebut dan mencetak hasil perhitungan ke layar.

### **Kesimpulan**

- Tipe data primitif seperti float digunakan untuk menyimpan data sederhana seperti bilangan pecahan.
- Dengan menggunakan tipe data primitif, kita dapat melakukan perhitungan matematika dasar secara efisien dan efektif. Program-program yang menggunakan tipe data primitif memungkinkan kita untuk melakukan operasi matematika dan pemrosesan data dengan mudah.

## **2. Unguided 2**

Jelaskan fungsi dari class dan struktur secara detail dan berikan contohnya

Jawaban :

Class dan Struct adalah dua konsep penting dalam pemrograman berorientasi objek (OOP) yang digunakan untuk membungkus data dan fungsionalitas terkait bersama menjadi satu unit. Class adalah blueprint atau cetakan yang digunakan untuk membuat objek. Secara umum, class berisi data (variabel anggota) dan fungsi (metode) yang terkait bersama.

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

class mahasiswa {
public:
    string nama;
    string nim;
    int semester;
    void ambilmatakuliah(string mata_kuliah) {
        cout << nama << " Mengambil mata kuliah " <<
mata_kuliah << endl;
        cout << "NIM : " << nim << " " << " Semester : " <<
semester << endl;
    }
};

int main(){
    mahasiswa mhs1;
    mhs1.nama = "Angga";
    mhs1.nim = "12345";
    mhs1.semester = 3;

    mahasiswa mhs2;
    mhs2.nama = "Rafi";
    mhs2.nim = "67891";
    mhs2.semester = 2;

    mhs1.ambilmatakuliah("Pemrograman Berorientasi Objek");
    mhs2.ambilmatakuliah("sistem operasi");

    return 0;
}
```



## Screenshot Program

```
strukdat_unguided2.cpp > mahasiswa > ambilmatakuliah(string)
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  class mahasiswa {
6  public:
7      string nama;
8      string nim;
9      int semester;
10     void ambilmatakuliah(string mata_kuliah) {
11         cout << nama << " Mengambil mata kuliah " << mata_kuliah << endl;
12         cout << "NIM : " << nim << " " << " Semester : " << semester << endl;
13     }
14 };
15
16 int main(){
17     mahasiswa mhs1;
18     mhs1.nama = "Angga";
19     mhs1.nim = "12345";
20     mhs1.semester = 3;
21
22     mahasiswa mhs2;
23     mhs2.nama = "Rafi";
24     mhs2.nim = "67891";
25     mhs2.semester = 2;
26
27     mhs1.ambilmatakuliah("Pemrograman Berorientasi Objek");
28     mhs2.ambilmatakuliah("sistem operasi");
29
30     return 0;
31 }
```

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd
unguided2 } ; if ($?) { .\strukdat_unguided2 }
Angga Mengambil mata kuliah Pemrograman Berorientasi Objek
NIM : 12345 Semester : 3
Rafi Mengambil mata kuliah sistem operasi
NIM : 67891 Semester : 2
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1>
```

Struktur atau struct, pada dasarnya, sama seperti class dengan perbedaan terletak pada hak akses anggota. Dalam struct, semua anggota dideklarasikan sebagai publik secara default, sedangkan dalam class, anggota biasanya dideklarasikan sebagai private. Selain itu, struct juga dapat digunakan untuk merepresentasikan satu set data yang terkait, tetapi dengan fungsionalitas yang lebih sedikit daripada class.

```
#include <iostream>

#include <cmath>

using namespace std;

struct Point {
    double x;
    double y;
};

double jarak(Point p1, Point p2){
    double dx = p1.x - p2.x;
    double dy = p1.y - p2.y;
    return sqrt(dx * dx + dy * dy);
}

int main(){
    Point p1 = {0, 0};
    Point p2 = {3, 4};

    double jarak_p1_p2 = jarak(p1,p2);

    cout << "Jarak antara (" << p1.x << ", " << p1.y << ")
dan (" << p2.x << ", " << p2.y << ") adalah " << jarak_p1_p2 <<
endl;

    return 0;
}
```

## Screenshot Program

```
strukdat_unguidedver2.cpp > jarak(Point, Point)
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  struct Point {
6      double x;
7      double y;
8  };
9
10 double jarak(Point p1, Point p2){
11     double dx = p1.x - p2.x;
12     double dy = p1.y - p2.y;
13     return sqrt(dx * dx + dy * dy);
14 }
15
16 int main(){
17     Point p1 = {0, 0};
18     Point p2 = {3, 4};
19
20     double jarak_p1_p2 = jarak(p1,p2);
21
22     cout << "Jarak antara (" << p1.x << ", " << p1.y << ") dan (" << p2.x << ", " << p2.y << ") adalah " << jarak_p1_p2 << endl;
23
24     return 0;
25 }
```

```
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd
at_unguidedver2 } ; if ($?) { .\strukdat_unguidedver2 }
Jarak antara (0, 0) dan (3, 4) adalah 5
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1>
```

Program ini menggunakan struct untuk membuat tipe data baru bernama Point yang memiliki dua variabel double x dan y untuk menyimpan koordinat suatu titik. Kemudian program juga menggunakan fungsi jarak() yang menerima dua parameter bertipe Point dan mengembalikan jarak antara kedua titik tersebut. Di dalam fungsi main(), program menginisialisasi dua variabel Point, yaitu p1 dan p2. Setelah itu, program memanggil fungsi jarak() dan menyimpan hasilnya ke dalam variabel jarak\_p1\_p2. Hasilnya kemudian ditampilkan menggunakan fungsi cout dengan format yang jelas dan rapi.

### 3. Unguided 3

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

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

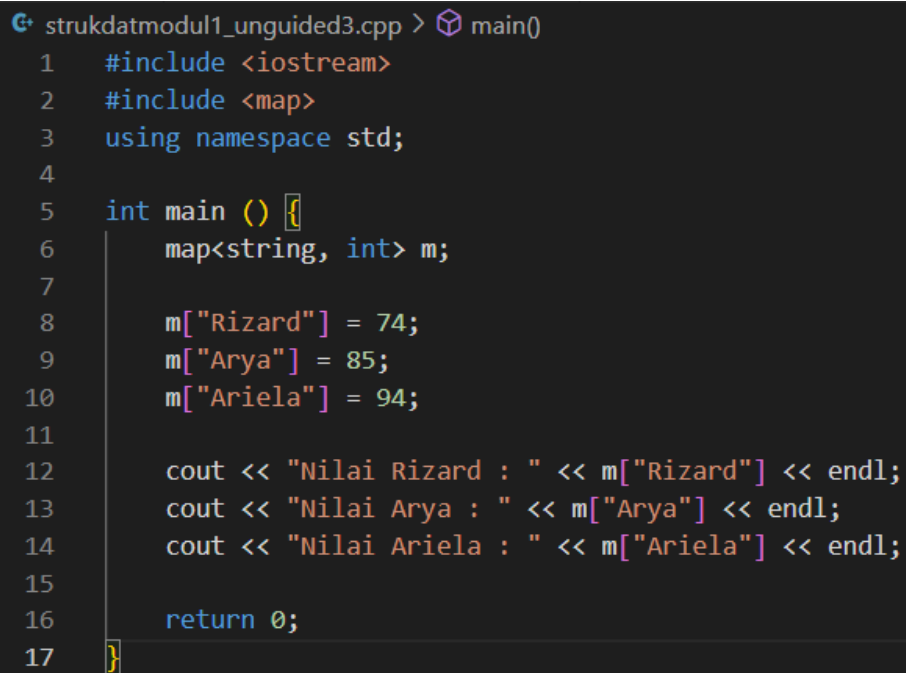
int main () {
    map<string, int> m;

    m["Rizard"] = 74;
    m["Arya"] = 85;
    m["Ariela"] = 94;

    cout << "Nilai Rizard : " << m["Rizard"] << endl;
    cout << "Nilai Arya : " << m["Arya"] << endl;
    cout << "Nilai Ariela : " << m["Ariela"] << endl;

    return 0;
}
```

### Screenshot Program



```
strukdatmodul1_unguided3.cpp > main()
1  #include <iostream>
2  #include <map>
3  using namespace std;
4
5  int main () {
6      map<string, int> m;
7
8      m["Rizard"] = 74;
9      m["Arya"] = 85;
10     m["Ariela"] = 94;
11
12     cout << "Nilai Rizard : " << m["Rizard"] << endl;
13     cout << "Nilai Arya : " << m["Arya"] << endl;
14     cout << "Nilai Ariela : " << m["Ariela"] << endl;
15
16     return 0;
17 }
```

```
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1> cd 'ukdatmodul1_unguided3' ; if ($?) { .\strukdatmodul1_unguided3 }
Nilai Rizard : 74
Nilai Arya : 85
Nilai Ariela : 94
PS C:\Users\ACER\Downloads\PRAKTIKUM STRUKTUR DATA\STRUKDAT1>
```

Program di atas membuat objek `m` dari class `map` dengan tipe kunci `string` dan tipe nilai `int`. Kemudian, program memasukkan tiga pasangan kunci-nilai ke dalam objek `m` menggunakan operator `[]` dan mencetak nilai dari masing-masing pasangan.

Perbedaan antara array dan map adalah sebagai berikut:

- **Ukuran** : Array memiliki ukuran yang tetap saat dideklarasikan dan tidak dapat diubah setelahnya, sementara ukuran map dapat ditambah atau dikurangi sesuai dengan kebutuhan.
- **Tipe data kunci** : Array hanya dapat menggunakan tipe data integer sebagai indeksnya, sedangkan map dapat menggunakan tipe data apapun sebagai kuncinya.
- **Inisialisasi** : Array harus diinisialisasi pada saat deklarasi atau dapat diisi dengan nilai nol atau acak jika tidak diinisialisasi. Map tidak perlu diinisialisasi pada saat deklarasi dan dapat diisi dengan pasangan kunci-nilai sepanjang waktu.

Dengan menggunakan map, kita dapat mengakses nilai dengan lebih mudah dan fleksibel karena kunci yang digunakan dapat berupa tipe data apapun, sementara array hanya dapat diakses menggunakan indeks integer langsung. Namun, map memerlukan lebih banyak memori daripada array karena harus menyimpan kunci untuk setiap nilai yang disimpan di dalamnya.

## **BAB IV**

### **KESIMPULAN**

Kesimpulannya dari ketiga tipe data ini memiliki kegunaannya masing-masing dalam pemrograman. Tipe data primitif digunakan untuk menyimpan nilai tunggal sederhana, sedangkan tipe data abstrak digunakan untuk menyimpan nilai yang lebih kompleks dan terstruktur. Sementara itu, tipe data koleksi digunakan untuk menyimpan sekumpulan nilai atau objek. Penting bagi seorang pemrogram untuk memahami perbedaan dan kegunaan masing-masing tipe data ini, agar dapat memilih tipe data yang tepat sesuai kebutuhan program yang dibuat.

## **BAB V**

### **REFERENSI**

- <https://it.telkomuniversity.ac.id/6-tipe-data-pemrograman-yang-harus-anda-pahami-apa-saja/>
- <https://revou.co/kosakata/tipe-data>
- <https://codingstudio.id/blog/macam-tipe-data-pemrograman/>