

**LAPORAN PRAKTIKUM STRUKTUR DATA
DAN ALGORITMA**

**MODUL I
TIPE DATA**



Disusun Oleh :

NAMA : MUHAMAD NASRULLOH

NIM : 2311102044

KELAS : IF 11-B

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 merupakan sebuah kategori yang mengklasifikasikan data berdasarkan karakteristik dan kegunaannya. Tipe data ini penting dalam pemrograman karena menentukan bagaimana data tersebut akan diolah. Jenis tipe data sebagai berikut :

1| Tipe Data Primitif

Seperti namanya, adalah tipe data yang sederhana dan disediakan langsung oleh bahasa pemrograman. Tipe data ini bisa menyimpan satu nilai seperti digit, byte, atau karakter hanya saja tidak bisa menjangkau data dengan tingkat kerumitan tinggi.

- Integer (bilangan bulat berupa positif atau negatif). Contoh 45, 10, -70
- Float (angka pecahan atau desimal). Contoh 22.5, 54.3, 60.2
- Character (angka/huruf/karakter spesial). Contoh 9, a, %
- Boolean (benar: true/salah: false)

2| Tipe Data Abstrak

Dibuat berdasarkan tipe data primitif atau tipe data abstrak lainnya, tidak didefinisikan secara langsung oleh programmer tapi disediakan oleh bahasa pemrograman atau pustaka eksternal. Contohnya seperti String, Data, File, Queue, Stack.

3| Tipe Data Kolektif

Dapat digunakan untuk menyimpan kumpulan data, biasanya terdiri dari beberapa elemen dengan tipe data yang sama atau berbeda. Contoh array (kumpulan tipe data sama), list (tipe data sama, urutan data penting), set (tipe data sama, urutan tidak penting), tuple (tipe data berbeda), dan dictionary (kumpulan data dengan pasangan kunci-nilai).

B. GUIDED

I - Source Code

```
#include <iostream>
using namespace std;
// Main program
int main () {
    char op;
    float num1, num2;

    // It allows user to enter operator i.e. +, -, *, /
    cout << "Masukkan operator (+, -, *, /): ";
    cin >> op;

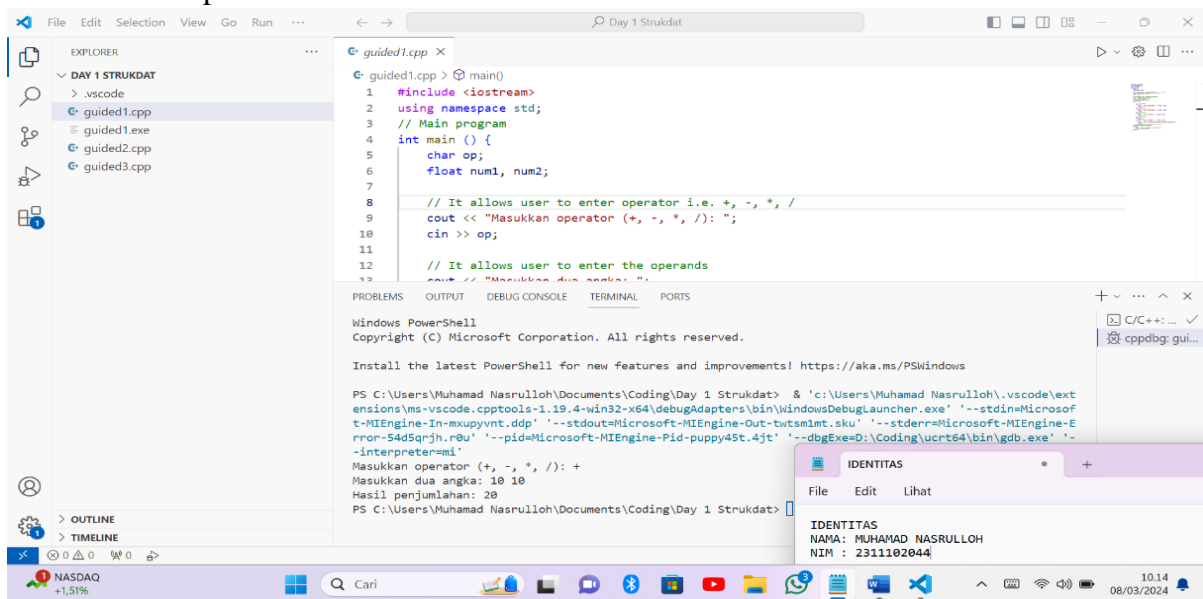
    // It allows user to enter the operands
    cout << "Masukkan dua angka: ";
    cin >> num1 >> num2;
    // Switch statement begins
    switch (op)
    {
        // If user enter +
        case '+':
            cout << "Hasil penjumlahan: " << num1 + num2;
            break;
```

```

// If user enter -
case '-':
    cout << "Hasil pengurangan: " << num1 - num2;
    break;
// If user enter *
case '*':
    cout << "Hasil perkalian: " << num1 * num2;
    break;
// If user enter /
case '/':
    if (num2 !=0)
        cout << "Hasil pembagian: " << num1 / num2;
    else
        cout << "Error! Tidak dapat melakukan pembagian dengan nol!";
        break;
// If the operator is other than +, -, * or /,
// error message will display
default:
    cout << "Error! operator is not correct";
} // switch statement ends
return 0;
}

```

Screenshot Output



Deskripsi

Program diatas menampilkan hasil operasi matematika (penjumlahan, pengurangan, perkalian, pembagian) dari 2 angka yang di input oleh user. Dengan tipe data yang di deklarasikan di awal adalah character – variabel op, dan float – variabel num1 dan num2, yang merupakan jenis tipe data primitif.

II – 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;
}

```

Screenshot Output

The screenshot displays the Visual Studio Code interface. The Explorer panel on the left shows a project named 'DAY 1 STRUKDAT' with files like 'guided1.cpp', 'guided1.exe', 'guided2.cpp', 'guided2.exe', and 'guided3.cpp'. The main editor window shows the source code of 'guided2.cpp', which includes a struct definition for 'Mahasiswa' and a 'main' function that initializes two student objects and prints their details. The Output panel at the bottom shows the program's execution, displaying the struct definition and the printed data for two students: Dian (Mataram, 22) and Bambang (Surabaya, 23). A small 'IDENTITAS' window is also visible in the bottom right corner, showing the student's name as MUHAMMAD NASRULLOH and NIM as 2311102044.

Deskripsi

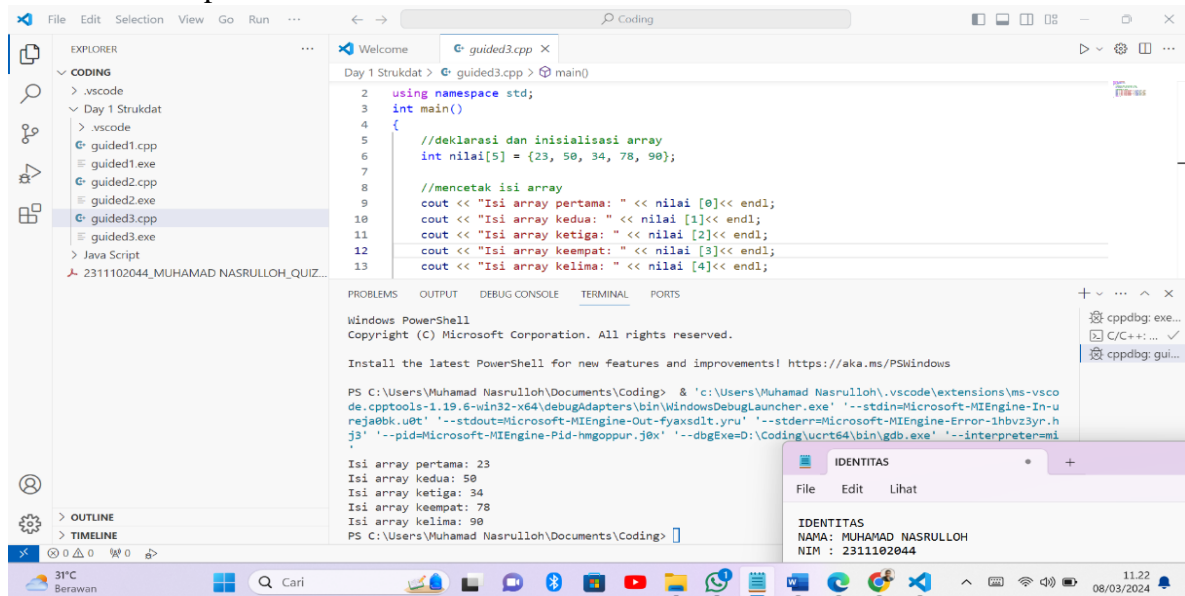
Program diatas menampilkan informasi biodata mahasiswa (nama, alamat, dan umur). Code diawali dengan struct, tipe data abstrak yang fungsinya untuk membungkus berbagai variable dan tipe data seperti diatas (character di ikuti oleh pointer *). Sedangkan const sebelum char itu menunjukkan bahwa nilai variabelnya nya paten, tidak dapat dirubah.

III – Source Code

```
#include <iostream>
using namespace std;
int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5] = {23, 50, 34, 78, 90};

    //mencetak isi 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;
}
```

Screenshot Output



Deskripsi

Program diatas menampilkan data yang tersimpan dalam array yang berjumlah 5, dan tipe data yang di deklarasi diawal berupa interger. Meskipun arraynya berisi 5 nilai tapi yang ditampilkan dalam cout- itu ada 4 indeks karena indeks array selalu dimulai dari 0.

C. UNGUIDED

1. Buatlah program menggunakan tipe data primitif minimal 2 fungsi dan bebas.
Menampilkan program, jelaskan program tersebut dan ambil kesimpulan dari materi tipe data primitif!

Source Code

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

float celciusToFahrenheit(float celcius) {
    return (celcius * 9/5) + 32;
}
```

```

}
float fahrenheitToCelcius(float fahrenheit) {
    return (fahrenheit - 32) * 5/9;
}

int main() {
    float suhu, hasil;
    int pilihan;

    // Menampilkan menu program
    cout << "Mau konversi yang mana?" << endl;
    cout << "1. Celcius ke Fahrenheit" << endl;
    cout << "2. Fahrenheit ke Celcius" << endl;
    cout << "Pilihan kamu: ";
    cin >> pilihan;

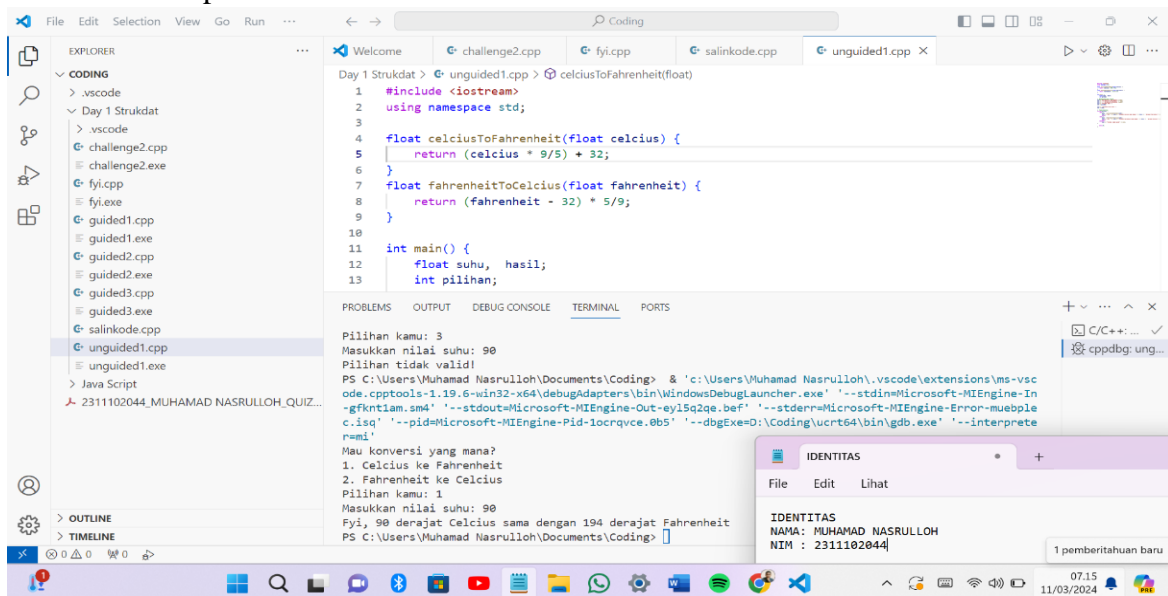
    cout << "Masukkan nilai suhu: ";
    cin >> suhu;

    // Proses konversi
    switch (pilihan) {
        case 1:
            hasil = celciusToFahrenheit(suhu);
            cout << "Fyi, " << suhu << " derajat Celcius sama dengan " <<
            hasil << " derajat Fahrenheit " << endl;
            break;
        case 2:
            hasil = fahrenheitToCelcius(suhu);
            cout << "Fyi, " << suhu << " derajat Fahrenheit sama dengan "
            << hasil << " derajat Celcius " << endl;
            break;
        default:
            cout << "Pilihan tidak valid!" << endl;
    }

    return 0;
}

```

Screenshot Output



Deskripsi

Program diatas menampilkan konversi suhu dari Celcius ke Fahrenheit, atau sebaliknya. Dengan 2 fungsi celciusToFahrenheit dan fahrenheitToCelcius yang dipanggil ke dalam seleksi Switch Case untuk dapat memberikan hasil konversi dari suhu yang di inputkan user. Untuk tipe data bisa dilihat ada 2 juga float dan interger.

Kesimpulan

Tipe data primitif (seperti float, int) adalah elemen dasar didalam pemrograman yang berfungsi untuk menyimpan berbagai jenis data seperti bilangan bulat, karakter maupun nilai boolean. Pemahaman tipe data primitif cukup penting dalam membuat program yang kompleks dan berperan menentukan seleksi program yang tepat.

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

Class dan struct ini merupakan jenis struktur data yang digunakan untuk mendefinisikan tipe data baru. Selain itu, class berfungsi sebagai encapsulation (pembungkusan, menggabungkan data yang terkait dalam satu unit), inheritance (pewarisan, membuat kelas baru/turunan yang menurunkan sifat dan perilaku dari kelas yang sudah ada), polymorphism (memanipulasi objek dari kelas turunan menggunakan pointer atau referensi kelas dasar). Sedangkan struct sebenarnya hampir sama dengan class, hanya saja berguna dalam pembuatan kode program yang lebih sederhana/ringkas dan mudah dibaca. Perbedaan lainnya ada pada default aksesibilitas anggotanya, class bersifat private/pribadi dan struct bersifat public.

Contoh program

```
#include <iostream>
#include <string>

using namespace std;

// untuk menyimpan informasi alamat
struct Alamat {
    string jalan;
    string kota;
    string negara;
    int kodePos;
};

// untuk menyimpan informasi tanggal
struct Tanggal {
    int hari;
    int bulan;
    int tahun;
};

// untuk menyimpan biodata orang
class Orang {
public:
    void setNama(string namaLengkap) {
```

```

        this->namaLengkap = namaLengkap;
    }
    string getNama() {
        return namaLengkap;
    }
    void setTanggal(Tanggal tanggalLahir) {
        this->tanggalLahir = tanggalLahir;
    }
    Tanggal getTanggal() {
        return tanggalLahir;
    }
    void setAlamat(Alamat alamatRumah) {
        this->alamatRumah = alamatRumah;
    }
    Alamat getAlamat() {
        return alamatRumah;
    }
private:
    string namaLengkap;
    Tanggal tanggalLahir;
    Alamat alamatRumah;
};

int main() {
    Orang dataOrang;

    // Inisialisasi
    dataOrang.setNama("Lin Yi");
    Tanggal tanggalLahir = {11, 1, 1999};
    dataOrang.setTanggal(tanggalLahir);
    Alamat alamatRumah = {"Shenyang", "Liaoning", "China", 12345};
    dataOrang.setAlamat(alamatRumah);

    // Menampilkan biodata
    cout << "======" << endl;
    cout << "BIODATA DIRI" << endl;
    cout << "Nama: " << dataOrang.getNama() << endl;
    cout << "Tanggal Lahir: " << dataOrang.getTanggal().hari << "/" <<
dataOrang.getTanggal().bulan << "/" << dataOrang.getTanggal().tahun <<
endl;
    cout << "Alamat: " << dataOrang.getAlamat().jalan << ", " <<
dataOrang.getAlamat().kota << ", " << dataOrang.getAlamat().negara << "
" << dataOrang.getAlamat().kodePos << endl;

    return 0;
}

```

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;

int main() {

```



```

// Deklarasi (nama sebagai key, harga sebagai value)
map<string, double> menu = {
    {"Nasi Goreng", 20000.0},
    {"Mie Ayam", 18000.0},
    {"Bakso", 15000.0},
    {"Es Teh", 5000.0},
    {"Jus Alpukat", 8000.0},
};

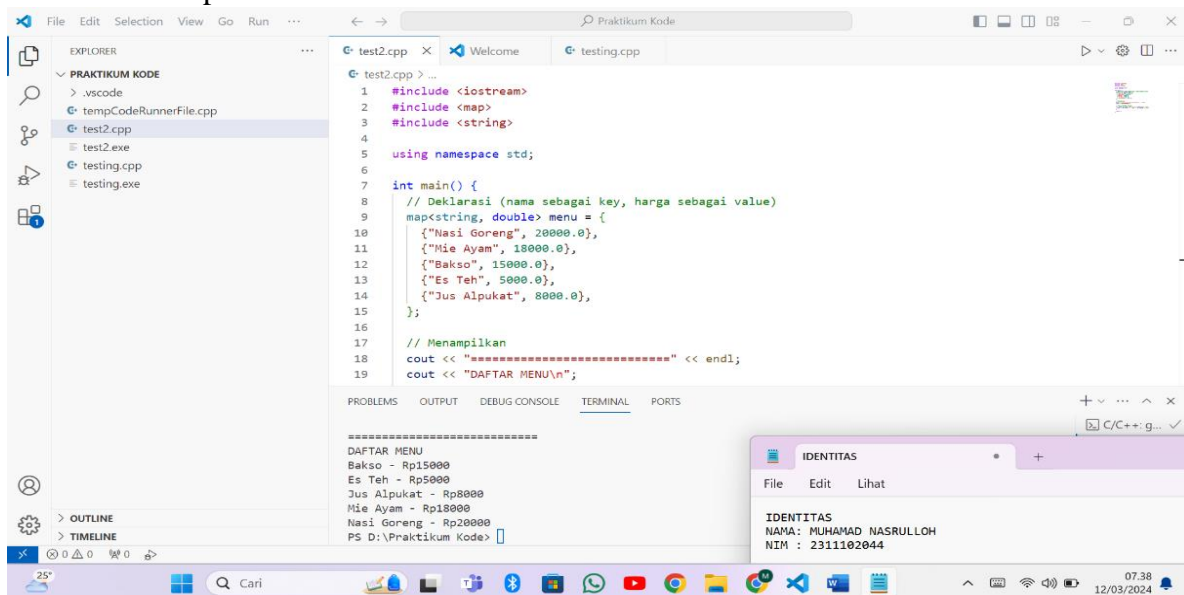
// Menampilkan
cout << "======" << endl;
cout << "DAFTAR MENU\n";

// Iterasi dengan iterator
for (auto it = menu.begin(); it != menu.end(); ++it) {
    cout << it->first << " - Rp" << it->second << endl;
}

return 0;
}

```

Screenshot Output



Deskripsi

Program diatas menampilkan sebuah daftar menu dan harga. Menggunakan fungsi map yang dinamai menu dengan key bertipe string dan value bertipe double. Dalam program ini, di deklarasikan map untuk menyimpan informasi jenis makanan dan minuman yang ditawarkan.

Perbedaan Array dengan Map

- Struktur : Array linear berurutan, map non-linear pasangan key-value
- Tipe data elemen : Array sama, map key dan value dapat berbeda
- Pengaksesan elemen : Array indeks numerik, map key unik
- Modifikasi data : Array dengan indeks, map dengan key

D. KESIMPULAN

Tipe data merupakan fondasi penting dalam pemrograman. Sama halnya ketika membangun rumah, memilih bahan baku yang tepat sangatlah penting untuk membangun rumah yang kokoh dan efisien. Diibaratkan bahan baku itu adalah tipe data, dan rumah adalah programnya. Sehingga pemahaman dalam tipe data ini cukup penting sebelum melaju menjadi programmer yang profesional dan kode yang semakin kompleks.

E. REFERENSI

[1] Team (2023), Macam-Macam Tipe Data Pemrograman dan Fungsinya, Programmer Wajib Tahu. Bekasi: Coding Studio.

[2] Suci Wulandari (2023), 10 Jenis Tipe Data Pemrograman Populer 2023. DKI Jakarta: Dibimbing.

[3] Teddy Tri Saputro (2018), Pemrograman Berorientasi Objek Dengan C++. Tangerang Selatan: Rumah Embedded