

**LAPORAN PRAKTIKUM**  
**PERTEMUAN 1**



**Nama :**

Razhendriya Vania Ramadhan Suganjarsarwat      2311104048

**Dosen :**

**WAHYU ANDI SAPUTRA**

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

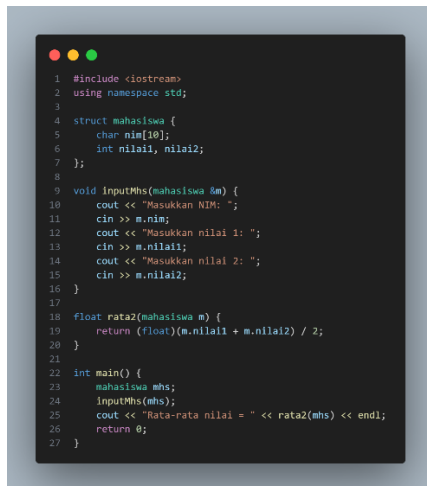
## I. TUJUAN

- **Memahami Struktur Data:** Kode ini memperkenalkan konsep *struct* (struktur) yang digunakan untuk mengelompokkan data yang berhubungan, seperti NIM, nilai1, dan nilai2 menjadi satu kesatuan yang disebut "mahasiswa". Ini adalah langkah awal untuk memahami cara mengorganisasi data dalam pemrograman.
- **Mengenal Fungsi:** Konsep fungsi (seperti `inputMhs` dan `rata2`) sangat penting dalam pemrograman. Fungsi memungkinkan kita untuk memecah masalah menjadi bagian-bagian yang lebih kecil dan dapat digunakan kembali. Dengan fungsi, kode menjadi lebih terstruktur dan mudah dipahami.
- **Mempelajari Algoritma:** Kode ini mengimplementasikan algoritma sederhana untuk menghitung rata-rata. Algoritma adalah langkah-langkah logis untuk menyelesaikan suatu masalah. Dengan mempelajari algoritma, Anda akan terbiasa dengan cara berpikir komputasional.
- **Memahami Input dan Output:** Kode ini menggunakan `cin` untuk menerima input dari pengguna (NIM dan nilai) dan `cout` untuk menampilkan hasil (rata-rata). Ini adalah dasar dari interaksi antara program dan pengguna.
- **Menerapkan Konsep Pemrograman Berorientasi Objek (OOP):** Meskipun sederhana, kode ini sudah mulai menerapkan konsep OOP dengan menggunakan struktur data. OOP adalah paradigma pemrograman yang populer dan banyak digunakan dalam pengembangan perangkat lunak.

## II. LANDASAN TEORI

Kode ini memperkenalkan konsep dasar pemrograman C++ seperti struktur data, fungsi, algoritma, dan input/output. Struktur data digunakan untuk mengelompokkan data, fungsi untuk memecah kode menjadi bagian-bagian yang lebih kecil, algoritma untuk menyelesaikan masalah, dan input/output untuk berinteraksi dengan pengguna. Kode ini juga memperkenalkan konsep pemrograman berorientasi objek (OOP) melalui penggunaan struktur data.

### III. GUIDE



```
1 #include <iostream>
2 using namespace std;
3
4 struct mahasiswa {
5     char nim[10];
6     int nilai1, nilai2;
7 };
8
9 void InputMhs(mahasiswa m) {
10     cout << "Masukkan NIM: ";
11     cin >> m.nim;
12     cout << "Masukkan nilai 1: ";
13     cin >> m.nilai1;
14     cout << "Masukkan nilai 2: ";
15     cin >> m.nilai2;
16 }
17
18 float rata2(mahasiswa m) {
19     return (float)(m.nilai1 + m.nilai2) / 2;
20 }
21
22 int main() {
23     mahasiswa mhs;
24     InputMhs(mhs);
25     cout << "Rata-rata nilai = " << rata2(mhs) << endl;
26     return 0;
27 }
```

Kode ini dirancang untuk:

- Mengumpulkan data mahasiswa: Nama, NIM, dan dua nilai.
- Menghitung rata-rata nilai.
- Menampilkan hasil perhitungan.

### UNGUIDED



```
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 struct Mahasiswa {
7     string nama;
8     int nim, uts, uas, tugas;
9     double nilaiAkhir;
10 };
11
12 double hitungNilaiAkhir(int uts, int uas, int tugas) {
13     return 0.3 * uts + 0.4 * uas + 0.3 * tugas;
14 }
15
16 int main() {
17     Mahasiswa mhs[10];
18     // ... (kode untuk mengisi data mahasiswa)
19     for (int i = 0; i < 10; i++) {
20         mhs[i].nilaiAkhir = hitungNilaiAkhir(mhs[i].uts, mhs[i].uas, mhs[i].tugas);
21     }
22     // ... (kode untuk menampilkan data mahasiswa)
23     return 0;
24 }
```

- **Struktur Mahasiswa:**
  - Ini adalah cetak biru untuk membuat objek yang mewakili seorang mahasiswa.
  - Setiap objek Mahasiswa memiliki atribut seperti nama, NIM, nilai UTS, UAS, tugas, dan nilai akhir.
- **Fungsi hitungNilaiAkhir:**
  - Fungsi ini menghitung nilai akhir seorang mahasiswa berdasarkan bobot nilai UTS, UAS, dan tugas yang telah ditentukan.
- **Array mhs:**
  - Ini adalah kumpulan dari 10 objek Mahasiswa. Setiap elemen dalam array mewakili satu mahasiswa.

Bagian kode ini dirancang untuk mengelola data mahasiswa. Struktur Mahasiswa mendefinisikan format data yang akan disimpan, sedangkan fungsi hitungNilaiAkhir digunakan untuk melakukan perhitungan nilai akhir. Array mhs digunakan untuk menyimpan data dari beberapa mahasiswa.



```
1 struct Pelajaran {
2     string namaMapel;
3     string kodeMapel;
4 };
5
6 Pelajaran createPelajaran(string nama, string kode) {
7     Pelajaran pel;
8     pel.namaMapel = nama;
9     pel.kodeMapel = kode;
10    return pel;
11 }
12
13 void tampilPelajaran(Pelajaran pel) {
14     cout << "Nama Pelajaran: " << pel.namaMapel << endl;
15     cout << "Kode Pelajaran: " << pel.kodeMapel << endl;
16 }
```

- **Struktur Pelajaran:**
  - Struktur ini merepresentasikan data tentang sebuah pelajaran, seperti nama mata pelajaran dan kode mata pelajaran.
- **Fungsi createPelajaran:**
  - Fungsi ini digunakan untuk membuat objek Pelajaran baru dengan memberikan nama dan kode mata pelajaran.
- **Fungsi tampilPelajaran:**
  - Fungsi ini digunakan untuk menampilkan informasi tentang sebuah objek Pelajaran ke layar.

Bagian kode ini membuat sebuah Abstrak Data Type (ADT) untuk mewakili data pelajaran. ADT ini memudahkan dalam mengelola data pelajaran dalam program.

```

1  #include <iostream>
2
3  using namespace std;
4
5  void tampilArray2D(int arr[][3], int baris, int kolom) {
6      // ... (kode untuk menampilkan isi array)
7  }
8
9  void tukarElemen(int arr1[][3], int arr2[][3], int baris, int kolom, int i, int j) {
10     // ... (kode untuk menukar elemen pada posisi i,j)
11 }
12
13 void tukarPointer(int *ptr1, int *ptr2) {
14     int temp = *ptr1;
15     *ptr1 = *ptr2;
16     *ptr2 = temp;
17 }
18
19 int main() {
20     int arr1[3][3], arr2[3][3];
21     // ... (kode untuk mengisi array)
22     tampilArray2D(arr1, 3, 3);
23     tampilArray2D(arr2, 3, 3);
24     tukarElemen(arr1, arr2, 3, 3, 1, 2);
25     // ... (kode untuk menukar menggunakan pointer)
26     return 0;
27 }

```

- **Fungsi tampilArray2D:**

- Fungsi ini digunakan untuk menampilkan isi dari sebuah array dua dimensi ke layar.

- **Fungsi tukarElemen:**

- Fungsi ini digunakan untuk menukar nilai dua elemen pada posisi tertentu dalam dua buah array dua dimensi.

- **Fungsi tukarPointer:**

- Fungsi ini digunakan untuk menukar nilai yang ditunjuk oleh dua buah pointer.

Bagian kode ini berfokus pada manipulasi array dua dimensi. Fungsi-fungsi yang disediakan memungkinkan kita untuk menampilkan isi array, menukar elemen, dan melakukan operasi lain pada array.

#### **IV. KESIMPULAN**

Kode-kode di atas memperkenalkan konsep dasar pemrograman C++ seperti struktur data, fungsi, array, pointer, dan ADT. Dengan memahami konsep-konsep ini, dapat membangun program yang lebih kompleks untuk mengelola berbagai jenis data.

**Intinya:**

- **Struktur:** Kelompokkan data terkait.
- **Fungsi:** Bagi program menjadi bagian-bagian kecil.
- **Array:** Simpan kumpulan data yang sama jenis.
- **Pointer:** Akses dan manipulasi data langsung.
- **ADT:** Buat tipe data khusus.