

Tugas Proceeding  
Modul 3 Struktur Data  
ABSTRACT DATA TYPE



**Disusun Oleh:**  
**Christian Felix Saliman Sugiono (2311104031)**  
**S1SE0701**

**Dosen:**  
**Yudha Islami Sulistya**

**Program Studi S1 Software Engineering**  
**Fakultas Informatika**  
**Telkom University**  
**Purwokerto**

**2024**  
**Tugas Pendahuluan Modul 3**  
**STRUKTUR DATA - Ganjil 2024/2025**  
**" ABSTRACT DATA TYPE"**

**A. Ketentuan Tugas Pendahuluan**

1. Tugas Pendahuluan dikerjakan secara **Individu**.
2. TP ini bersifat **WAJIB**, tidak mengerjakan = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
3. Hanya **MENGUMPULKAN** tetapi **TIDAK MENGERJAKAN** = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
4. Deadline pengumpulan TP Modul 2 adalah Senin, 23 September 2024 pukul 06.00 WIB.
5. **TIDAK ADA TOLERANSI KETERLAMBATAN, TERLAMBAT ATAU TIDAK MENGUMPULKAN TP MAKA DIANGGAP TIDAK MENGERJAKAN**.
6. **DILARANG PLAGIAT (PLAGIAT = E)**.
7. Kerjakan TP dengan jelas agar dapat dimengerti.
8. File diupload di LMS menggunakan format **PDF** dengan ketentuan:  
**TP\_MOD\_[XX]\_NIM\_NAMA.pdf**

**CP (WA):**

- Andini (082243700965)
- Imelda (082135374187)

**SELAMAT MENGERJAKAN^^**

## B. Soal Tugas Modul 3

1 Inputan mahasiswa dan nilai, untuk menyelesaikan masalah ini kita dapat menuliskan kodingan seperti berikut ini:

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

struct Mahasiswa {
    string nama;
    string nim;
    float uts, uas, tugas, nilai_akhir;
};

float hitungNilaiAkhir(float uts, float uas, float tugas) {
    return (0.3 * uts) + (0.4 * uas) + (0.3 * tugas);
}

int main() {
    const int MAX_MAHASISWA = 10;
    Mahasiswa mahasiswa[MAX_MAHASISWA];
    int jumlahMahasiswa;

    cout << "Masukkan jumlah mahasiswa (maksimum 10): ";
    cin >> jumlahMahasiswa;

    if (jumlahMahasiswa > MAX_MAHASISWA || jumlahMahasiswa <= 0) {
        cout << "Jumlah mahasiswa tidak valid!";
        return 1;
    }

    for (int i = 0; i < jumlahMahasiswa; i++) {
        cout << "\nMahasiswa " << i + 1 << endl;
        cout << "Nama: ";
        cin.ignore();
        getline(cin, mahasiswa[i].nama);
        cout << "NIM: ";
        getline(cin, mahasiswa[i].nim);
        cout << "Nilai UTS: ";
        cin >> mahasiswa[i].uts;
        cout << "Nilai UAS: ";
        cin >> mahasiswa[i].uas;
        cout << "Nilai Tugas: ";
        cin >> mahasiswa[i].tugas;

        mahasiswa[i].nilai_akhir = hitungNilaiAkhir(mahasiswa[i].uts, mahasiswa[i].uas, mahasiswa[i].tugas);
    }

    cout << "\nData Mahasiswa dan Nilai Akhir:\n";
    for (int i = 0; i < jumlahMahasiswa; i++) {
        cout << "Nama: " << mahasiswa[i].nama << endl;
        cout << "NIM: " << mahasiswa[i].nim << endl;
        cout << "Nilai UTS: " << mahasiswa[i].uts << endl;
        cout << "Nilai UAS: " << mahasiswa[i].uas << endl;
        cout << "Nilai Tugas: " << mahasiswa[i].tugas << endl;
        cout << "Nilai Akhir: " << mahasiswa[i].nilai_akhir << endl;
        cout << "-----" << endl;
    }

    return 0;
}
```



Christian Felix  
x

hal pertama yang perlu kita gunakan adalah membuat struct mahasiswa sebagai awalan dari kodingan yang dimana struct ini akan berisi nama, nim, dan nilai nilai yang akan kita hitung sebagai nilai ahirnya, setelah membuat struct langkah berikutnya yaitu untuk

membuat fungsi untuk menghitung nilai akhir dengan inputan nilai uts, uas, dan nilai tugas yang masing masing akan dikali dengan bobotnya masing-masing.

Lalu setelah itu kita akan membuat fungsi mainnya yang dimana kita akan meng set limit dari mahasiswa sebanyak 10 orang, lalu kita akan membuat input untuk jumlah mahasiswa yang akan di inputkan oleh user. Lalu kita akan menggunakan if untuk mengecek jumlah inputan, apabila inputan >10 kita akan mengeluarkan jumlah tidak valid, tetapi bila terdapat jumlah yang <=10 maka program akan jalan dengan perulangan sebanyak x kali (x dari inputan user) untuk perulangan memasukan data mahasiswa dari nama nim dan nilai, setelah itu kita akan mengeprint outputnya dengan berapa saja jumlah mahasiswa yang telah dimasukan oleh user, untuk ouputnya seperti berikut ini:

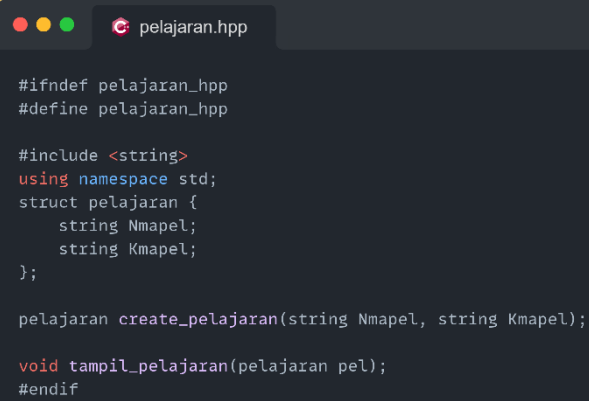
```
Mahasiswa 1
Nama: Tsaqif
NIM: 2311104024
Nilai UTS: 88
Nilai UAS: 83
Nilai Tugas: 81

Mahasiswa 2
Nama: Maulana
NIM: 2311104034
Nilai UTS: 84
Nilai UAS: 93
Nilai Tugas: 90

Data Mahasiswa dan Nilai Akhir:
Nama: Tsaqif
NIM: 2311104024
Nilai UTS: 88
Nilai UAS: 83
Nilai Tugas: 81
Nilai Akhir: 83.9
-----
Nama: Maulana
NIM: 2311104034
Nilai UTS: 84
Nilai UAS: 93
Nilai Tugas: 90
Nilai Akhir: 89.4
-----
```

2 Adt pelajaran dengan 3 jenis file, pelajaran.hpp, pelajaran.cpp, main.cpp, untuk kodingan dari masing masing file ini adalah sebagai berikut ini:

### Pelajaran.hpp



```
#ifndef pelajaran_hpp
#define pelajaran_hpp

#include <string>
using namespace std;
struct pelajaran {
    string Nmapel;
    string Kmapel;
};

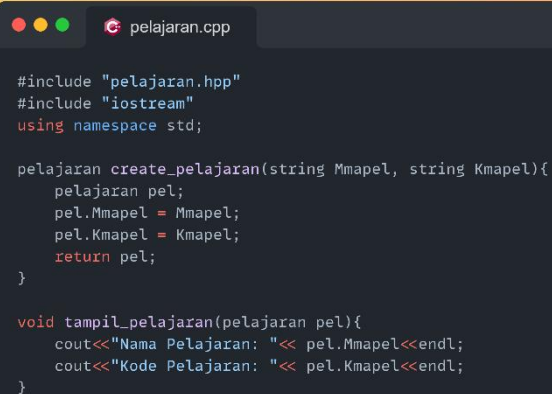
pelajaran create_pelajaran(string Nmapel, string Kmapel);

void tampil_pelajaran(pelajaran pel);
#endif
```



Christian Felix  
X

### Pelajaran.cpp



```
#include "pelajaran.hpp"
#include "iostream"
using namespace std;

pelajaran create_pelajaran(string Mmapel, string Kmapel){
    pelajaran pel;
    pel.Mmapel = Mmapel;
    pel.Kmapel = Kmapel;
    return pel;
}

void tampil_pelajaran(pelajaran pel){
    cout<<"Nama Pelajaran: "<< pel.Mmapel<<endl;
    cout<<"Kode Pelajaran: "<< pel.Kmapel<<endl;
}
```



Christian Felix  
X

## Main.cpp



Dari file pertama pelajaran.hpp, kegunaan dari file ini adalah sebagai file yang merupakan pembuat kelas, untuk file pelajaran.cpp berguna sebagai tempat menjalankan file pelajaran.hpp yang dimana kita membuat var baru yakni pel sebagai string penyimpanan data yang akan kita isi pada main.cpp, dan untuk file main.cpp berguna sebagai pengesekusi dari kedua file yang tersedia dengan pemanggilan file.hpp dan.cpp pelajaran yang dimana kita akan mengisi string data dengan apa yang kita mau, namun dalam kasus ini kita akan mengisinya dengan Struktur data sebagai nama mapelnya dan STD sebagai kode matkulnya, dan untuk output dari apa yang telah kita buat dari 3 file ini adalah sebagai berikut ini.

```
nama pelajaran : Struktur Data
nilai : STD
```

3 Array bersyarat, untuk menyelesaikan permasalahan berikut ini dapat di selesaikan dengan kodingan seperti berikut ini,

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

void tampilkanArray(int arr[3][3]) {
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
    cout << endl;
}

void tukarArray(int arr1[3][3], int arr2[3][3], int baris, int kolom) {
    int temp = arr1[baris][kolom];
    arr1[baris][kolom] = arr2[baris][kolom];
    arr2[baris][kolom] = temp;
}

void tukarPointer(int* p1, int* p2) {
    int temp = *p1;
    *p1 = *p2;
    *p2 = temp;
}

int main() {

    int array1[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
    int array2[3][3] = { {9, 8, 7}, {6, 5, 4}, {3, 2, 1} };

    int a = 10, b = 20;
    int* p1 = &a;
    int* p2 = &b;

    cout << "Array 1 sebelum ditukar:" << endl;
    tampilkanArray(array1);
    cout << "Array 2 sebelum ditukar:" << endl;
    tampilkanArray(array2);

    tukarArray(array1, array2, 1, 1);

    cout << "Array 1 setelah ditukar elemen [1][1]: " << endl;
    tampilkanArray(array1);
    cout << "Array 2 setelah ditukar elemen [1][1]: " << endl;
    tampilkanArray(array2);

    cout << "Nilai sebelum pointer ditukar: " << endl;
    cout << "p1: " << *p1 << ", p2: " << *p2 << endl;

    tukarPointer(p1, p2);

    cout << "Nilai setelah pointer ditukar: " << endl;
    cout << "p1: " << *p1 << ", p2: " << *p2 << endl;
    return 0;
}
```



Christian Felix



Fungsi awal yang ada pada kodingan diatas (tampilkan array) memiliki fungsi untuk menunjukkan hasil dari array 2 dimensi dengan ukuran 3x3, dan karena menggunakan looping maka array akan ditampilkan secara berbaris, fungsi kedua dari kodingan ini yakni (tukar array) berfungsi seperti namanya yakni menukarkan posisi elemen satu dengan elemen lainya berdasarkan inputan kolom dan baris yang diberikan, fungsi ketiga (tukar pointer) fungsi dari kodingan ini yakni menukar nilai dua nilai berdasarkan apa yang ditunjuk pointer p1 dan p2, fungsi ke empat (main) fungsi dari kodingan ini yakni sebagai tempat pemanggilan semua fungsi fungsi yang terdapat pada sebelumnya dengan inputan nilainya masing masing. Untuk hasil outputnya sebagai berikut ini.

```
Array 1 sebelum ditukar:
1 2 3
4 5 6
7 8 9

Array 2 sebelum ditukar:
9 8 7
6 5 4
3 2 1

Array 1 setelah ditukar elemen [1][1]:
1 2 3
4 5 6
7 8 9

Array 2 setelah ditukar elemen [1][1]:
9 8 7
6 5 4
3 2 1

Nilai sebelum pointer ditukar:
p1: 10, p2: 20
Nilai setelah pointer ditukar:
p1: 20, p2: 10
```



### C. Jawaban dari Tp modul 3 STD

1 pointer merupakan var yang menyimpan alamat dari suatu variabel lainnya, pointer memungkinkan kita untuk mengakses dan mengganti data yang tersimpan dalam memori tertentu.

2 untuk menampilkan alamat dari memori dalam C++ kita dapat memakai simbol & untuk mengakses memori dari variabel yang kita inginkan

Contohnya yaitu sebagai berikut ini: `int x = 5`

`Cout<<"alamat dari x adalah"<<&a<<endl` hasil dari kode ini akan menampilkan alamat dari x

3 contoh dari cara penggunaan pointer dari bahasa C++ adalah sebagai berikut ini

```
#include <iostream>
```

```
Using namespace std;
```

```
Int main (){
```

```
    Int x = 20;
```

```
    Int *ptr = &x; kegunaan ptr dalam kasus ini adalah sebagai pointer yang menyimpan alamat x
```

```
    Cout<<"nilai dari x melalui pointer adalah: "<< *ptr << endl;
```

```
    Return 0;
```

```
}
```

Hasil yang akan diberikan oleh ptr adalah nilai dari x yang menyimpan dari alamat x tersebut

4 pengertian dari ADT adalah model dari data data tertentu yang dapat mendefinisikan data dan operasi yang dapat dijalankan dalam tipe data tersebut, yang dilakukan oleh ADT adalah cara data tersebut berperilaku bukan bagaimana cara ia di implementasikan dalam sebuah tipe data. Contohnya yaitu bisa stack, queue dan list.

5 contoh dari ADT dalam dunia nyata adalah seperti sebuah smartphone kita mengerti cara melakukan operasi operasi pada smartphone kita seperti mengambil gambar, membuka aplikasi, dan menyalakan senter pada smartphone, tetapi kita tidak terlalu mementingkan bagaimana ia melakukan semua task task tersebut.

6 contoh kodingan C++ untuk bangun ruang kerucut adalah sebagai berikut ini

```
#include iostream
```

```
Class kerucut {
```

```
Private:
```

```
    Double tinggi;
```

```
    Double radius;
```

```
Public:
```

```
    Kerucut(double t, double r) : tinggi (t), radius (r) {}
```

```
    Double calcvolume(){
```

```
        return (1.0 / 3.0) * M_PI * pow(radius, 2) * tinggi;
```

```
    }
```

```

        Double calcsurface(){
            Double selimutKerucut = sqrt(pow(radius, 2) + pow(tinggi, 2));
            return M_PI * radius * (radius + selimutKerucut);
        }
};

int main() {
    Kerucut k(3.0, 5.0);
    std::cout << "Volume kerucut: " << k.hitungVolume() << std::endl;
    std::cout << "Luas permukaan kerucut: " << k.hitungLuasPermukaan() <<
    std::endl;
    return 0;
}

```

Dan kodingan tersebut adalah untuk perhitungan volume dari kerucut beserta selimutnya