

LAPORAN PRAKTIKUM
STRUKTUR DATA
ABSTRACT DATA TYPE



Nama :

Muhammad Mahrus Ali (2311104006)

Dosen :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

A. Soal Tugas Pendahuluan

1. Jelaskan apa yang dimaksud dengan pointer

Pointer adalah sebuah variabel dalam pemrograman yang menyimpan alamat memori dari variabel lain. Dalam bahasa pemrograman seperti C atau C++, pointer digunakan untuk merujuk ke lokasi memori tertentu secara langsung. Alih-alih menyimpan nilai data itu sendiri, pointer menyimpan alamat dari data tersebut.

2. Bagaimana cara menampilkan Alamat memori dari suatu variable dalam program C++? Berikan contoh!

Untuk menampilkan alamat memori dari suatu variabel menggunakan **&**. Jadi kita bisa tau alamat itu berada dimana pada suatu variabel seperti contoh dibawah :



```
#include <iostream>

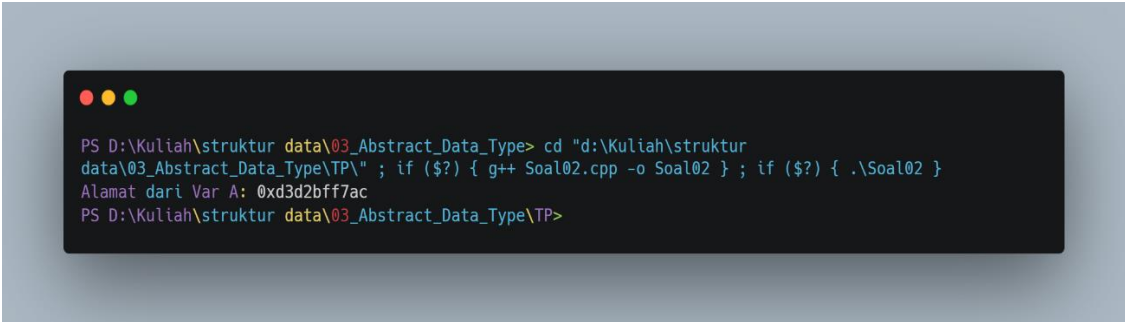
using namespace std;
// agar tahu alamat suatu Variabel

int main(){
    int a = 12;

    cout << "Alamat dari Var A: " << &a << endl;

    return 0;
}
```

Outputnya:



```
PS D:\Kuliah\struktur data\03_Abstract_Data_Type> cd "d:\Kuliah\struktur
data\03_Abstract_Data_Type\TP\" ; if ($?) { g++ Soal02.cpp -o Soal02 } ; if ($?) { .\Soal02 }
Alamat dari Var A: 0xd3d2bff7ac
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\TP>
```

3. Bagaimana cara menggunakan pointer dalam program C++? Berikan contoh!

Untuk memakai pointer kita hanya menambahkan * di awal variable pada variable yang telah dirujuk. Contoh variable a nilainya 12 lalu kita buat variable `int* ptr` yang berisikan Alamat variable a lalu telah terhubung kedua variable ini maka kita bisa mengakses isi value variable a melalui pointer `*ptr`

```
#include <iostream>

using namespace std;
// Contoh Penggunaan Pointer and Friends

int main(){
    int a = 12;
    int* ptr = &a;

    cout << "Nilai Dari num: " << a << endl;
    cout << "Alamat Memori Dari num: " << ptr << endl;
    cout << "Nilai Yang Dirujuk: " << *ptr << endl;
    *ptr = 36;
    cout << "Setelah Di Pointer: " << a << endl;

    return 0;
}
```

Output dari program diatas adalah:

```
PS D:\Kuliah\struktur data\03_Abstract_Data_Type> cd "d:\Kuliah\struktur
data\03_Abstract_Data_Type\TP\" ; if ($?) { g++ Soal03.cpp -o Soal03 } ; if ($?) { .\Soal03 }
Nilai Dari num: 12
Alamat Memori Dari num: 0xe3ba1ff9b4
Nilai Yang Dirujuk: 12
Setelah Di Pointer: 36
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\TP>
```

4. Jelaskan apa yang dimaksud dengan Abstract Data Type (ADT)?

Abstract Data Type (ADT) adalah konsep yang mengacu pada tipe data yang didefinisikan secara abstrak, di mana hanya sifat atau perilakunya yang dijelaskan tanpa memperhatikan implementasi detail internalnya. ADT mendefinisikan apa yang dapat dilakukan oleh suatu tipe data (operasi apa yang bisa diterapkan), tetapi tidak menjelaskan bagaimana operasi tersebut diimplementasikan.

5. Berikan contoh ilustrasi sederhana didalam dunia nyata, tetapi diluar konteks pemrograman!

Abstract Data Type (ADT) dalam dunia nyata bisa diilustrasikan dengan penggunaan mesin ATM. Ketika seseorang menggunakan ATM, mereka hanya berfokus pada layanan yang bisa diakses, seperti cek saldo, tarik tunai, setor tunai, dan transfer uang. Pengguna tidak perlu tahu bagaimana mesin ATM bekerja di dalamnya, seperti bagaimana mesin memproses saldo atau menghitung uang. Mereka hanya berinteraksi dengan antarmuka sederhana yang memberikan akses ke layanan tersebut. Inilah yang disebut abstraksi, di mana detail teknis disembunyikan, sementara fungsi-fungsi yang dapat dilakukan tetap terlihat jelas bagi pengguna. Sama halnya dengan ADT, yang mendefinisikan sekumpulan operasi yang bisa digunakan tanpa memaparkan detail implementasinya. Proses-proses seperti perhitungan, pemindahan uang, dan pengelolaan transaksi semuanya terselubung, sementara pengguna hanya berfokus pada hasil dari operasi yang mereka pilih. Ini mencerminkan prinsip ADT di mana pengguna hanya melihat apa yang bisa dilakukan, tanpa mengetahui bagaimana itu dilakukan.

6. Tuliskan ADT dari bangun ruang kerucut dalam Bahasa C++!

Inputan:

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

class Kerucut {
private:
    double radius;
    double tinggi;

public:
    Kerucut(double r, double t) {
        radius = r;
        tinggi = t;
    }

    double hitungVolume() {
        return (1.0 / 3.0) * M_PI * radius * radius * tinggi;
    }

    double hitungLuasPermukaan() {
        double s = sqrt((radius * radius) + (tinggi * tinggi));
        return M_PI * radius * (radius + s);
    }

    double getRadius() {
        return radius;
    }

    double getTinggi() {
        return tinggi;
    }

    void setRadius(double r) {
        radius = r;
    }

    void setTinggi(double t) {
        tinggi = t;
    }
};

int main() {
    Kerucut kerucut1(3, 7);

    cout << "Volume kerucut: " << kerucut1.hitungVolume() << endl;
    cout << "Luas permukaan kerucut: " << kerucut1.hitungLuasPermukaan() << endl;

    return 0;
}
```

Outputnya:

```
PS D:\Kuliah\struktur data\03_Abstract_Data_Type> cd "d:\Kuliah\struktur
data\03_Abstract_Data_Type\TP\" ; if ($?) { g++ Soal06.cpp -o Soal06 } ; if ($?) { .\Soal06 }
Volume kerucut: 65.9734
Luas permukaan kerucut: 100.051
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\TP>
```

B. Latihan Unguided

1. Buat program yang dapat menyimpan data mahasiswa (max10) kedalam sebuah array dengan field nama, nim, uts, uas, tugas, dan nilai akhir. Nilai akhir diperoleh dari fungsi dengan rumus $0.3 \cdot \text{uts} + 0.4 \cdot \text{uas} + 0.3 \cdot \text{tugas}$.

Inputnya:

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

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

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

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

    cout << "Masukkan Jumlah Mahasiswa : ";
    cin >> jumlahMahasiswa;

    if(jumlahMahasiswa > MAKS)
    {
        cout << "Jumlah Maksimal 10" << endl;
        return 1;
    }

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

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

    cout << "\nHasil Akhir Nilai Mahasiswa\n";
    for (int i = 0; i < jumlahMahasiswa; i++)
    {
        cout << "Mahasiswa Ke- " << i+1 << endl;
        cout << "Nama: " << mahasiswa[i].nama << endl;
        cout << "NIM: " << mahasiswa[i].nim << endl;
        cout << "Nilai Tugas: " << mahasiswa[i].tugas << endl;
        cout << "Nilai UTS: " << mahasiswa[i].uts << endl;
        cout << "Nilai UAS: " << mahasiswa[i].uas << endl;
        cout << "Nilai Akhir: " << mahasiswa[i].nilaiAkhir << endl;
    }

    return 0;
}
```

Outputnya seperti dibawah:

```
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\Unguided> cd "d:\Kuliah\struktur
data\03_Abstract_Data_Type\Unguided\" ; if ($?) { g++ LAT_01.cpp -o LAT_01 } ; if ($?) { .\LAT_01 }
Masukkan Jumlah Mahasiswa : 1

Masukan Mahasiswa ke 1
Nama: Mahrus
NIM: 2311104006
Nilai Tugas: 80
Nilai UTS: 90
Nilai UAS: 90

Hasil Akhir Nilai Mahasiwa
Mahasiswa Ke- 1
Nama: Mahrus
NIM: 2311104006
Nilai Tugas: 80
Nilai UTS: 90
Nilai UAS: 90
Nilai Akhir: 24
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\Unguided>
```

2. Jawaban ada dibawah

Pelajaran.h

```
#ifndef PELAJARAN_H
#define PELAJARAN_H

#include <string>

using namespace std;

struct pelajaran
{
    string namaMatkul;
    string kodeMatkul;
};

pelajaran create_pelajaran(string mataKuliah, string kodeMataKuliah);

void tampil_pelajaran(const pelajaran& pel);

#endif
```

File header yang mendefinisikan tipe data ADT pelajaran serta deklarasi fungsi dan prosedur terkait.

Pelajaran.cpp

```
#include "pelajaran.h"
#include <iostream>

pelajaran create_pelajaran(string mataKuliah, string kodeMataKuliah) {
    pelajaran pel;
    pel.namaMatkul = mataKuliah;
    pel.kodeMatkul = kodeMataKuliah;
    return pel;
}

void tampil_pelajaran(const pelajaran& pel) {
    cout << "Nama Mata Kuliah: " << pel.namaMatkul << endl;
    cout << "Kode Mata Kuliah: " << pel.kodeMatkul << endl;
}
```

pelajaran.cpp: Implementasi dari fungsi create_pelajaran untuk membuat instance pelajaran, dan tampil_pelajaran untuk menampilkan isi dari pelajaran.

Main.cpp

```
#include <iostream>
#include "pelajaran.h"

using namespace std;

int main() {
    string mataKuliah = "Struktur Data";
    string kodeMataKuliah = "STD";

    pelajaran pel = create_pelajaran(mataKuliah, kodeMataKuliah);

    tampil_pelajaran(pel);

    return 0;
}
```

main.cpp: Program utama yang menguji implementasi ADT pelajaran dengan menggunakan dua string, yaitu matakuliah dan kodematakuliah.

3. Buatlah program dengan ketentuan :
 - 2 buah array 2D integer berukuran 3x3 dan 2 buah pointer integer
 - fungsi/prosedur yang menampilkan isi sebuah array integer 2D

- fungsi/prosedur yang akan menukarkan isi dari 2 array integer 2D pada posisi tertentu
- fungsi/prosedur yang akan menukarkan isi dari variabel yang ditunjuk oleh 2 buah pointer

untuk menjawab soal diatas kita bisa Inputkan code berikut:

```
#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;
    }
}

void tukarArrayPadaPosisi(int arr1[3][3], int arr2[3][3], int row, int col)
{
    int temp = arr1[row][col];
    arr1[row][col] = arr2[row][col];
    arr2[row][col] = temp;
}

void tukarPointer(int* ptr1, int* ptr2) {
    int temp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = 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* ptr1 = &a;
    int* ptr2 = &b;

    cout << "Array 1 sebelum pertukaran:" << endl;
    tampilkanArray(array1);
    cout << "\nArray 2 sebelum pertukaran:" << endl;
    tampilkanArray(array2);
    tukarArrayPadaPosisi(array1, array2, 1, 1);

    cout << "\nArray 1 setelah pertukaran pada posisi [1][1]:" << endl;
    tampilkanArray(array1);
    cout << "\nArray 2 setelah pertukaran pada posisi [1][1]:" << endl;
    tampilkanArray(array2);

    cout << "\nNilai sebelum pertukaran pointer:" << endl;
    cout << "ptr1 menunjuk ke: " << *ptr1 << endl;
    cout << "ptr2 menunjuk ke: " << *ptr2 << endl;
    tukarPointer(ptr1, ptr2);

    cout << "\nNilai setelah pertukaran pointer:" << endl;
    cout << "ptr1 menunjuk ke: " << *ptr1 << endl;
    cout << "ptr2 menunjuk ke: " << *ptr2 << endl;

    return 0;
}
```


Berikut ini outputnya:

```
PS D:\Kuliah\struktur data\03_Abstract_Data_Type> cd "d:\Kuliah\struktur
data\03_Abstract_Data_Type\Unguided\" ; if ($?) { g++ LAT_03.cpp -o LAT_03 } ; if ($?) { .\LAT_03 }
Array 1 sebelum pertukaran:
1 2 3
4 5 6
7 8 9

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

Array 1 setelah pertukaran pada posisi [1][1]:
1 2 3
4 5 6
7 8 9

Array 2 setelah pertukaran pada posisi [1][1]:
9 8 7
6 5 4
3 2 1

Nilai sebelum pertukaran pointer:
ptr1 menunjuk ke: 10
ptr2 menunjuk ke: 20

Nilai setelah pertukaran pointer:
ptr1 menunjuk ke: 20
ptr2 menunjuk ke: 10
PS D:\Kuliah\struktur data\03_Abstract_Data_Type\Unguided>
```