

LAPORAN PRAKTIKUM

MODUL 3

ABSTRACT DATA TYPE



Disusun Oleh:

Prajna paramitha - 2311104016

SE 07 01

Dosen :

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

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY

PURWOKERTO

2024

A. Soal Tugas Pendahuluan

1. Jelaskan apa yang dimaksud dengan pointer!

Jawaban : pointer merupakan salah satu basic tipe variable yang menyimpan alamat dalam hexadecimal.

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

Jawaban : dengan memberikan & didepan variable

```
pA = &a;

cout << "Nilai dari a adalah : " << a << endl;
cout << "Lokasi memori var a adalah : " << &a << endl;
```

3. Bagaimana cara menggunakan pointer dalam program C++? Berikan contoh cara menampilkan nilai yang tersimpan pada suatu alamat melalui pointer!

Pertama membuat variable untuk pointernya, dan memanggil data nilai yg dituju pointer dengan menggunakan (*)

```
cout << "Nilai dari yg dituju p1 adalah : " << *p1 << endl;

int *p1 = nullptr;
int *p2 = nullptr;
```

4. Jelaskan apa yang dimaksud dengan Abstract data type (ADT)!

Abstract Data Type (ADT) adalah suatu model atau konsep yang mendefinisikan tipe data berdasarkan perilaku atau operasi yang dapat dilakukan. ADT merupakan definisi dari static, dan ADT adalah TYPE dan sekumpulan primitive terhadap type itu sendiri.

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

Ilustrasi ADT : playlist music

Tipe data ADT : koleksi lagu dalam playlistnya

Operasi :

- Menambahkan lagi : menambahkan lagu
- Menghapus lagu : menghapus lagu
- Memutar lagu : memutar lagu
- Mengacak lagu : memutar lagu secara acak

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

```
#include "kerucut.h" //memanggil library file kerucut.h
#include <iostream>
#include <cmath>

const double PI = 3.14159265358979323846;

// Konstruktor
Kerucut::Kerucut(double r, double t) : jariJari(r), tinggi(t) {}

// Selektor
double Kerucut::getJariJari() const { return jariJari; }
double Kerucut::getTinggi() const { return tinggi; }

// Mutator
void Kerucut::setJariJari(double r) { jariJari = r; }
void Kerucut::setTinggi(double t) { tinggi = t; }

// Penghitung
double Kerucut::hitungVolume() const {
    return (1.0 / 3.0) * PI * jariJari * jariJari * tinggi;
}

double Kerucut::hitungLuasPermukaan() const {
    double sisiMiring = sqrt(jariJari * jariJari + tinggi * tinggi);
    return PI * jariJari * (jariJari + sisiMiring);
}

double Kerucut::hitungLuasSelimut() const {
    double sisiMiring = sqrt(jariJari * jariJari + tinggi * tinggi);
    return PI * jariJari * sisiMiring;
}

// Input/Output
void Kerucut::input() {
    std::cout << "Masukkan jari-jari kerucut: ";
    std::cin >> jariJari;
    std::cout << "Masukkan tinggi kerucut: ";
    std::cin >> tinggi;
}

void Kerucut::tampilkan() const {
    std::cout << "Properti Kerucut:" << std::endl;
    std::cout << "Jari-jari: " << jariJari << std::endl;
    std::cout << "Tinggi: " << tinggi << std::endl;
    std::cout << "Volume: " << hitungVolume() << std::endl;
    std::cout << "Luas Permukaan: " << hitungLuasPermukaan() << std::endl;
    std::cout << "Luas Selimut: " << hitungLuasSelimut() << std::endl;
}
```

Didalam kerucut.h

```
#ifndef KERUCUT_H
#define KERUCUT_H

class Kerucut {
private:
    double jariJari;
    double tinggi;

public:
    // Konstruktor
    Kerucut(double r, double t);

    // Selektor
    double getJariJari() const;
    double getTinggi() const;

    // Mutator
    void setJariJari(double r);
    void setTinggi(double t);

    // Penghitung
    double hitungVolume() const;
    double hitungLuasPermukaan() const;
    double hitungLuasSelimut() const;

    // Input/Output
    void input();
    void tampilkan() const;
};

#endif // KERUCUT_H
```

Didalam file main

```
#include <iostream>
#include "kerucut.h"
// include library file kerucut h yg berisikan hitungan

int main() {
    Kerucut kerucutSaya(0, 0); // set nilai awal data 0, karena akan diinputkan oleh user

    std::cout << "Kalkulator Kerucut" << std::endl;
    kerucutSaya.input();
    // menginputkan angka

    std::cout << std::endl;
    kerucutSaya.tampilkan();

    return 0;
}
```

Kalkulator Kerucut

Masukkan jari-jari kerucut: 21
Masukkan tinggi kerucut: 15

Properti Kerucut:

Jari-jari: 21
Tinggi: 15
Volume: 6927.21
Luas Permukaan: 3088.02
Luas Selimut: 1702.58

Semoga Selalu diberi kemudahan^^