

LAPORAN PRAKTIKUM

Modul 3

Abstract Data Type



Disusun Oleh:

Eduardo Bagus Prima Julian

2311104025 S1SE-07-01

Dosen :

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

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. Code

1. Jelaskan apa yang dimaksud dengan pointer!

Pointer merupakan sebuah variabel yang digunakan untuk menyimpan alamat memori objek. Pointer memiliki tiga tujuan utama antara lain yaitu untuk mengalokasikan objek baru pada tumpukan, meneruskan fungsi ke fungsi lain, untuk melakukan iterasi atas elemen dalam array atau struktur data lainnya.

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

Membuat variabel pointer yang kemudian pointer tersebut menunjuk ke objek yang akan ditampilkan alamat memorinya. Contoh seperti berikut:

```
int *ptr = nullptr;
int x = 2;
ptr = &x;
cout << ptr;
```

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

Kurang lebih sama saat kita ingin menampilkan alamat memori dari suatu variabel, bedanya kita menambah * pada saat memanggil pointer tersebut di cout.

```
int *ptr = nullptr;
int x = 2;
ptr = &x;
cout << *ptr;
```

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

ADT merupakan sebuah konsep yang memisahkan spesifikasi dan implementasi dari suatu struktur data. ADT memungkinkan kita untuk fokus pada apa yang bisa dilakukan oleh struktur data, contohnya seperti menambahkan data, menghapus data, mencari data.

5. Berikan contoh ilustrasi sederhana di dalam dunia nyata, tetapi di luar konteks pemrograman!

Salah satu contoh sederhana yang ada di dunia nyata adalah tumpukan piring (stack) yang memiliki prinsip kerja LIFO (Last In First Out), yang memiliki artinya elemen yang dimasukan terakhir akan dikeluarkan pertama. Hal yang bisa dilakukan dalam stack pada contoh tumpukan piring adalah Push (menambahkan piring ke dalam tumpukan), Pop (mengambil piring dari atas tumpukan), dan Peek (melihat piring atas tanpa mengeluarkannya).

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

```
kerucut.h
#ifndef KERUCUT_H
#define KERUCUT_H
```

```
class Kerucut {
private:
    double radius;
    double tinggi;
```

```

public:
    Kerucut(double r, double h);

    double getRadius();

    double getTinggi();

    void setRadius(double r);

    void setTinggi(double t);

    double getVolume();

    double getLuasPermukaan();
};

#endif

```

```

kerucut.cpp
#include "kerucut.h"
#include <cmath>

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

double Kerucut::getRadius() {
    return radius;
}

double Kerucut::getTinggi() {
    return tinggi;
}

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

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

double Kerucut::getVolume() {
    return (1.0/3) * M_PI * pow(radius, 2) * tinggi;
}

```

```
double Kerucut::getLuasPermukaan() {  
    double garisPelukis = sqrt(pow(radius, 2) + pow(tinggi, 2));  
    return M_PI * radius * (radius + garisPelukis);  
}
```

main.cpp

```
#include <iostream>
```

```
#include "kerucut.h"
```

```
using namespace std;
```

```
int main() {
```

```
    Kerucut kerucut(3, 10);
```

```
    cout << "Jari-jari: " << kerucut.getRadius() << endl;
```

```
    cout << "Tinggi: " << kerucut.getTinggi() << endl;
```

```
    cout << "Volume Kerucut: " << kerucut.getVolume() << endl;
```

```
    cout << "Luas Permukaan Kerucut: " << kerucut.getLuasPermukaan() << endl;
```

```
    return 0;
```

```
}
```