

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**



NAMA : OKTA ARYO
NIM : 193020503035
KELAS : A
MODUL : III (POLIMORFOSIS)

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2020**

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**



Nama : OKTA ARYO
NIM : 193020503035
Kelas : A
Modul : III (POLIMORFOSIS)

Komposisi	MAX	Nilai
BAB I Tujuan dan Landasan Teori	10	8
BAB II Pembahasan	60	49
BAB III Kesimpulan	20	13
Daftar Pustaka	5	4
Lampiran	5	3
Jumlah	100	

Penilai
Asisten Praktikum

Diana

BAB I

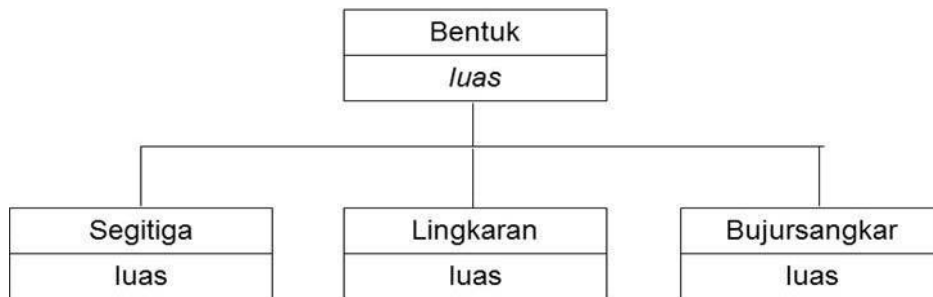
TUJUAN DAN LANDASAN TEORI

1.1.TUJUAN

1. Mahasiswa diharapkan mampu membuat polimorfosis

1.2.LANDASAN TEORI

Polimorfisme Memiliki arti “banyak bentuk”, melakukan hal yang sama untuk berbagai data yang berbeda, mengirimkan pesan yang sama ke berbagai objek yang berbeda karena tiap objek memberi respons dengan cara yang berbeda. Berikut ini merupakan contoh polimorfisme.



Polimorfisme memiliki syarat-syarat sebagai berikut:

- Ada hirarki pewarisan
- Kelas dalam hirarki pewarisan harus memiliki fungsi virtual (virtual method) dengan signature kelas.
- Menggunakan pointer atau rujukan ke kelas induk. Pointer digunakan untuk memanggil fungsi virtual

Polimorfisme dapat diimplementasikan dengan menggunakan dasar function overriding (melakukan redefinisi suatu fungsi di kelas anak, fungsi yang di-override memiliki signature sama, signature sama : tipe balik, nama fungsi, parameter sama) dan pewarisan. Suatu kelas disebut abstrak apabila memiliki

minimal satu fungsi abstrak. Fungsi abstrak merupakan fungsi yang tidak memiliki definisi (hanya deklarasi fungsi)/menggunakan fungsi virtual (pure virtual).

virtual balikan namaFungsi (parameter) = 0
--

Polymorphism adalah kemampuan untuk menggunakan operator atau fungsi dalam berbagai cara. Polimorfisme memberikan arti yang berbeda atau fungsinya kepada operator atau fungsi. Poly, merujuk ke banyak, menandakan banyak kegunaan dari operator dan fungsi ini. Fungsi tunggal penggunaan atau operator berfungsi dalam banyak cara bisa disebut polimorfisme. Polimorfisme mengacu pada kode, operasi atau benda yang berperilaku berbeda dalam konteks yang berbeda.

Polimorfisme adalah fitur yang kuat dari bahasa pemrograman berorientasi obyek C++ . Sebuah operator + berperilaku berbeda dalam konteks yang berbeda, seperti integer, float atau string yang mengacu konsep polimorfisme. Konsep di atas mengarah ke operator overloading. Konsep overloading juga merupakan cabang dari polimorfisme. Ketika keluar operator atau fungsi yang beroperasi pada tipe data baru itu kelebihan beban. Polimorfisme fitur ini mengarah pada konsep metode virtual.

Polimorfisme mengacu pada kemampuan untuk memanggil fungsi-fungsi yang berbeda dengan menggunakan hanya satu jenis fungsi panggil. Misalkan seorang programmer ingin kode kendaraan dari berbagai bentuk-bentuk seperti lingkaran, bujur sangkar, persegi panjang, dll. Salah satu cara untuk mendefinisikan masing-masing dari kelas-kelas ini adalah memiliki fungsi anggota untuk masing-masing yang membuat kendaraan dari masing-masing bentuk. Pendekatan nyaman lain pemrogram dapat mengambil adalah untuk mendefinisikan sebuah basis kelas bernama Shape dan kemudian membuat sebuah instance dari kelas tersebut.

Programmer dapat memiliki array yang menyimpan pointer ke semua objek yang berbeda dari kendaraan diikuti oleh struktur loop sederhana untuk membuat kendaraan, sesuai dengan bentuk yang dikehendaki, dengan memasukkan pointer ke array yang didefinisikan. Pendekatan ini mengarah kepada fungsi yang berbeda dieksekusi oleh panggilan fungsi yang sama. Polimorfisme digunakan untuk memberikan makna yang berbeda untuk konsep yang sama. Ini adalah dasar untuk pelaksanaan fungsi Virtual.

Dalam polimorfisme, satu fungsi atau operator berfungsi dalam banyak hal tergantung pada penggunaan berfungsi dengan benar. Agar hal ini terjadi, kondisi berikut harus berlaku:

1. Semua kelas-kelas yang berbeda harus berasal dari satu kelas dasar. Dalam contoh di atas, bentuk-bentuk kendaraan (lingkaran, segitiga, persegi panjang) yang dari satu kelas dasar yang disebut Shape.
2. Fungsi anggota harus dinyatakan virtual di kelas dasar. Dalam contoh di atas, fungsi anggota untuk membuat kendaraan harus dibuat sebagai virtual untuk kelas dasar.

BAB II

PEMBAHASAN

2.1 program polimorfisme

```
#include <iostream>
#include <conio.h>

using namespace std;

class buah{
    public:
        void data()
    {
        cout<<endl;
    }
};

class mangga: public buah{
    public:
        void data()
    {
        cout<<"=====OKTA-KUN NANI-
DESU===== "<<endl;

        cout<<"_____
_"<<endl;
        cout<<"|          infomasi Buahan          |"<<endl;

        cout<<"|_____|
"<<endl;

        cout<<"Warna kulitnya : hijau"<<endl;
        cout<<"Rasa Dari Buah Tersebut : asam/manis"<<endl;

        cout<<"=====
===== "<<endl;
    }
};

int main()
{
    buah *b;
    mangga *m;
```

```

        b->data();
        cout<<endl;
        m->data();

        getch();
    }

```

Penjelasan :

```

#include <iostream>
#include <conio.h>

using namespace std;

class buah{
    public:
        void data()
    {
        cout<<endl;
    }
};

```

Code program diatas adalah code awal dari program kelas buahan dalam polimorfosis. Dimana kelas dari program tersebut adalah buahan, dan untuk (void) berfungsi untuk mendeklarasikan data dari buahan yang di infutkan nanti.

```

class mangga: public buah{
    public:
        void data()
    {
        cout<<"=====OKTA-KUN NANI-
DESU===== "<<endl;

        cout<<" _____
_"<<endl;
        cout<<"|          infomasi Buahan          |"<<endl;

        cout<<"| _____ |
"<<endl;

```

Untuk code diatas adalah untuk penginfuttan buah yang kita inginkan dimana (class buah : public buah) ini berfungsi sebagai pembaca agar deklarasi dari buah terbaca.

```
cout<<"Warna kulitnya : hijau"<<endl;
    cout<<"Rasa Dari Buah Tersebut : asam/manis"<<endl;

    cout<<"=====
===== "<<endl;
}
};
```

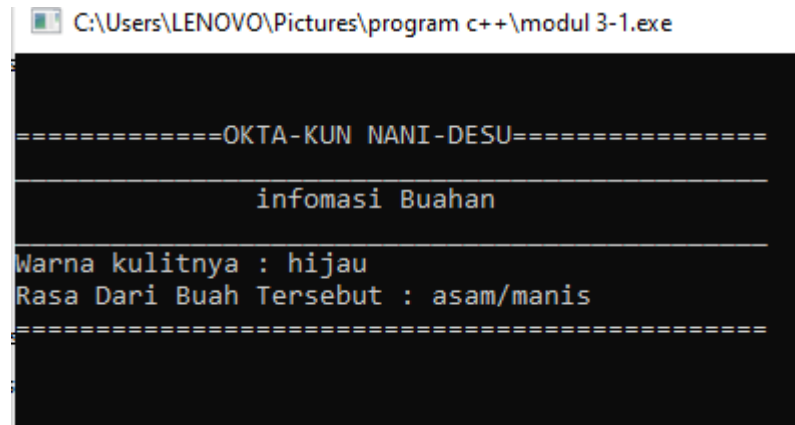
Code diatas hanya sebagai pemberi informasi dari kelas buahan.

```
int main()
{
    buah *b;
    mangga *m;

    b->data();
    cout<<endl;
    m->data();

    getch();
}
```

Code diatas adalah code utama dimana untuk deklarasi (buah) yaitu b sedangkan untuk deklarasi (manga) adalah m dan sebagai terakhir kita tidak perlu menginfutkan lagi data buah karena sudah terbaca didalam program code utama.

A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Users\LENOVO\Pictures\program c++\modul 3-1.exe". The command prompt shows the following output:

```
=====OKTA-KUN NANI-DESU=====
infomasi Buahan
Warna kulitnya : hijau
Rasa Dari Buah Tersebut : asam/manis
=====
```

Gambar 2.1. hasil tampilan output program polimorfosis

Gambar diatas adalah hasil dari utput program polimorfosis dan kita tidak perlu menginfutkan lagi karena sudah terbaca dan hasil output akan langsung tampil semua data dari kelas buah manga.

BAB III

KESIMPULAN

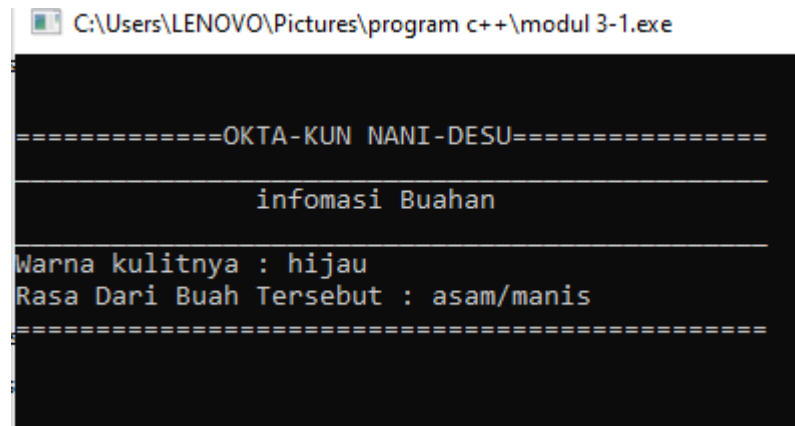
Polimorfosis didalam pemrograman berorientasi objek (PBO) berarti memiliki banyak bentuk karena didalam program polimorfosis memiliki fungsi dan operator yang sama ke semua elemen yang berbeda. Didalam c++ polimorfisme memiliki banyak turunan tetapi dari kelas dasar yang sama, apabila kelas turunan tersebut berbeda dengan kelas dasar maka kelas turunan tidak akan terdefinisi atau terdeklarasi didalam program. Polimorfosis sendiri banyak kita jumpai dikehidupan nyata salah satu contohnya adalah sebuah keluarga dimana kelas dasarnya adalah orang-tua sedangkan kelas turunannya adalah anak-anak.

DAFTAR PUSTAKA

Dosen Teknik Informatika. Algoritma dan Pemrograman II. 2020. Modul Praktikum Algoritma dan Pemrograman II. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.

<https://catataninformatikaku.blogspot.com/2017/03/polymorphism-dan-pengertiannya-pada-c.html>. 23 april 2020

LAMPIRAN



```
C:\Users\LENOVO\Pictures\program c++\modul 3-1.exe

====OKTA-KUN NANI-DESU====
infomasi Buahan
Warna kulitnya : hijau
Rasa Dari Buah Tersebut : asam/manis
=====
```

Gambar 2.1. hasil tampilan output program polimorfosis