

**LAPORAN HASIL PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN II**



**NAMA : JERI**  
**NIM : 193030503055**  
**KELAS : A**  
**MODUL : II (PEWARISAN)**

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

**LAPORAN HASIL PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN II**



Nama : JERI  
NIM : 193030503055  
Kelas : A  
Modul : PEWARISAN

Komposisi	MAX	Nilai
BAB I Tujuan dan Landasan Teori	10	7
BAB II Pembahasan	60	48
BAB III Kesimpulan	20	11
Daftar Pustaka	5	4
Lampiran	5	5
Jumlah	100	75

Penilai  
Asisten Praktikum

Diana

# BAB I

## TUJUAN DAN LANDASAN

### TEORI



#### 1.1. TUJUAN

Setelah menyelesaikan modul ini, mahasiswa diharapkan mampu :



1. Memahami dasar-dasar pemrograman berorientasi obyek.
2. Memahami enkapsulasi.
3. Membuat kelas dan objek.


#### 1.2. LANDASAN TEORI

Dalam PBO, kita mengambil realita sehari-hari. Kita melakukan pengamatan bahwa manusia secara alami sering melakukan pengelompokan atas objek atau benda. Sejauh ini kita mengetahui cara untuk melakukan pengelompokan – pengelompokan atas objek-objek yang serupa (menjadi kelas objek).

Selain melakukan katagorisasi terhadap objek yang memiliki sekumpulan atribut dan perilaku yang sama, manusia sering melakukan pengelompokan terhadap objek yang memiliki kesamaan atas beberapa (**tidak semua**) atribut/perilaku. Contoh: pengelompokan atas kendaraan bermotor, kemudian menggrupkannya berdasarkan suatu tipe atau jenis (mobil, truk, sepeda motor, dll.). setiap subkatagori ini merupakan kelas atas objek-objek yang serupa.

Ada beberapa karakteristik yang di-share oleh semua kelompok. Relasi antara kelas-kelas ini disebut dengan **relasi “is-a”**, dalam setiap kasus, objek yang dikelompokkan bersama dengan satu sub-katagori merupakan anggota dari katagori yang lebih umum. Contohnya adalah seperti di bawah ini.

-  Mobil adalah (“is-a”) kendaraan bermotor.
-  Truk adalah (“is-a”) kendaraan bermotor.

 Sepeda motor (“is-a”) kendaraan bermotor.

Objek yang dikelompokkan dalam satu kelas men-share sekumpulan atribut dan perilaku. Jadi, seluruh objek kendaraan bermotor memiliki sekumpulan atribut dan perilaku yang juga dimiliki oleh objek dari mobil. Keterkaitan antar kelas dalam relasi “is-a” berasal dari kenyataan bahwa sub kelas memiliki atribut dan perilaku yang dimiliki kelas induk, ditambah atribut dan perilaku yang dimiliki oleh sub kelas tersebut. Superclass (“kelas dasar” atau “kelas induk”) merupakan kelas yang lebih general dalam relasi “is-a”. subclass ( “kelas turunan” atau “kelas anak”) merupakan kelas yang lebih spesifik dalam relasi “is-a”. objek yang dikelompokkan dalam sub kelas memiliki atribut dan perilaku sub kelas induk, dan juga atribut dan perilaku tambahan. (jadi, kumpulan atribut dan perilaku sub kelas lebih besar dari super kelas-nya). Relasi “is-a” antar superclass dan subclasses-nya disebut dengan **pewarisan** dan *inheritance*.

Subclass “mewarisi” sesuatu superclass (atau juga bisa dikatakan sebuah subclass “turunan dari” suatu superclass) karena reusabilitas perangkat lunak, membuat kelas baru (kelas turunan) dari kelas yang sudah ada (kelas dasar), kelas turunan mewarisi kelas induk yang mendapatkan data dan perilaku , merupakan bentuk special dari kelas induk, dan diperluas dengan perilaku tambahan.

Pewarisan ada dua jenis yaitu pewarisan tunggal dan pewarisan jamak. Pada *protected acces*, **Protected** members dapat diakses oleh member kelas dasar, friend kelas dasar, member kelas turunan, dan friend kelas turunan. Kelas turunan dapat merujuk/mengakses langsung **public** dan **protected** data member kelas induk dengan menggunakan nama atribut yang diakses.

## **BAB II**

### **PEMBAHASAN**

#### **2.1. LANGKAH KERJA**

Buatlah 2 program dengan tema buah bagi yang nomor ganjil dan jika nomor genap maka tema hewan dengan menggunakan prinsip PEWARISAN.

#### **2.2. PEMBAHASAN**

##### **2.2.1. Program Hewan Pertama**

###### **Input Program**

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

class hewan{
protected:
    int berat , tinggi;
public:
    void set_values(int a, int b){
        berat = a;
        tinggi = b;
    }
};

class Tampilkan{
public:
    void Say(string i){
        cout << "" << i << "" << endl;
    }
};
```

```

class gajah: public hewan, public Tampilkan{
public:
    int tinggi(){
        return (berat);
    }
};

```

```

class Anakgajah: public hewan, public Tampilkan{
public:
    int berat(){
        return (tinggi);
    }
};

```

```

int main(){
    gajah gajahku;
    Anakgajah Anakgajahku;

```

```

    cout<<endl<<"===== "<<endl;
    gajahku.set_values(12,12);
    Anakgajahku.set_values(12,12);

```

```

    cout << gajahku.tinggi() << endl;
    cout << Anakgajahku.berat() << endl;
    cout<<endl<<"=====Modul II===== "<<endl;
    gajahku.Say("ini adalah Gajah ku");
    Anakgajahku.Say("Ini adalah Anak gajah ku");

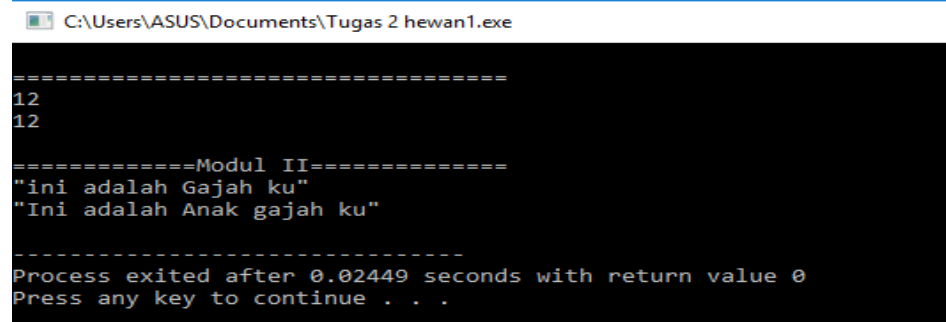
```

```

    return 0;
}

```

## Output Program



```
=====  
12  
12  
  
=====-Modul II=====  
"ini adalah Gajah ku"  
"Ini adalah Anak gajah ku"  
  
-----  
Process exited after 0.02449 seconds with return value 0  
Press any key to continue . . .
```

Gambar 2.1. Output program hewan 1

Penjelasan :

Pada awal program terdapat perintah **#include<iostream>**, (#) merupakan sebuah *preprocessor*, Preprocessor memberitahukan compiler untuk melakukan sesuatu sebelum melakukan kompilasi. Preprocessor yang berguna untuk standar input-output pada C++, kedua perintah tersebut digunakan untuk memanggil file header (include file) yang didalamnya terdapat perintah, fungsi atau prototype yang bisa digunakan dalam program yang dibuat. Jika perintah **#include** ini tidak ditulis, maka computer tidak mengerti perintah-perintah yang ditulis. Didalam coding program terdapat public: **,public** adalah tingkat akses yang berfungsi untuk memberikan hak akses secara umum kepada kelas-kelas turunannya maupun terhadap lingkungan luar didalam program. untuk block source kode harus diawali dengan tanda kurung kurawal { dan akhirnya dengan tanda tutup kurung kurawal } seperti yang terdapat pada coding program. Untuk mendeklarasikan lebih dari satu variabel, variabel tersebut dipisahkan dengan tanda koma (,) contoh nya **void set\_values(int a, int b)** artinya jika kita ingin mendeklarasikan variabel a dan b kita harus menggunakan tanda koma.

Didalam program tersebut juga terdapat fungsi **cout<<endl**, Fungsi dari **cout** yaitu untuk mencetak ke layar monitor, dan fungsi dari **endl** yaitu merupakan suatu fungsi manipulator yang digunakan untuk menyisipkan karakter Newline atau mengatur pindah baris.

Contohnya `cout << gajahku.tinggi() << endl;`, `cout << Anakgajahku.berat() << endl`. Pada akhir program terdapat fungsi **return 0**; yaitu untuk menyatakan hasil keluaran dari fungsi `main()` adalah 0. Pada C++ `return 0` dalam `main()` menyatakan bahwa program berakhir dengan normal.

### 2.2.2 Program hewan kedua

#### Input Program

```
#include <iostream>
#include <conio.h>
using namespace std;
class makhluk
{
    public:
        void berkembang();
};
class hewan : public makhluk
{
    public:
        void bergerak();
};
class kuda : public hewan
{
    public:
        void berlari();
};
main()
{
    cout<<endl<<"=====Modul II===== "<<endl;
    makhluk mk; hewan hw; kuda kd;
    cout<<endl<<" Sifat-sifat dari Makhluk adalah : "<<endl;
```

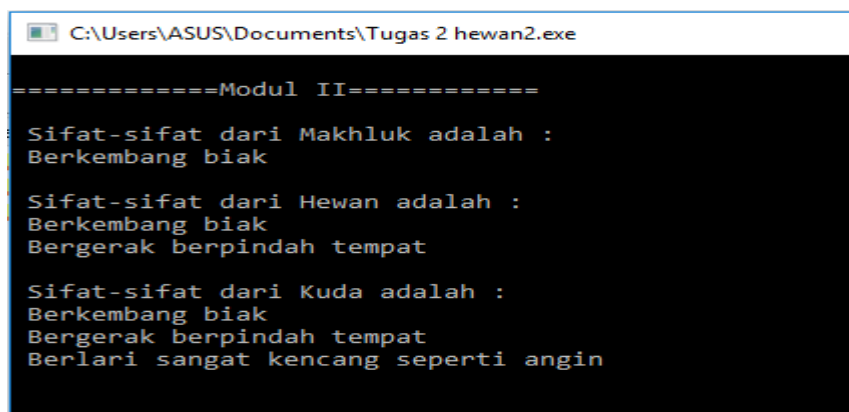


```

mk.berkembang();
cout<<endl<<" Sifat-sifat dari Hewan adalah : "<<endl;
hw.berkembang(); hw.bergerak();
cout<<endl<<" Sifat-sifat dari Kuda adalah : "<<endl;
mk.berkembang(); hw.bergerak(); kd.berlari();
}
void makhluk::berkembang()
{
    cout<<" Berkembang biak"<<endl;
}
void hewan::bergerak()
{
    cout<<" Bergerak berpindah tempat"<<endl;
}
void kuda::berlari()
{
    cout<<" Berlari sangat kencang seperti angin"<<endl;
    getch ();
}

```

## Output Program



```

C:\Users\ASUS\Documents\Tugas 2 hewan2.exe
=====Modul II=====
Sifat-sifat dari Makhluk adalah :
Berkembang biak
Sifat-sifat dari Hewan adalah :
Berkembang biak
Bergerak berpindah tempat
Sifat-sifat dari Kuda adalah :
Berkembang biak
Bergerak berpindah tempat
Berlari sangat kencang seperti angin

```

Gambar 2.2. Output program hewan 2

Penjelasan :

Pada program yang ke dua ini saya membuat program untuk sifat-sifat dari makhluk, hewan dan hewan pilihan kita.. Pertama-tama yang kita masukan adalah **#include<iostream>** dan **#include<conio.h>**. Kode yang diawali dengan **#** merupakan sebuah *preprocessor*, Preprocessor memberitahukan compiler untuk melakukan sesuatu sebelum melakukan kompilasi. Preprocessor yang berguna untuk standard input – output pada C++. Dan selanjutnya kita menentukan classnya yaitu **Class Makhluk** ,untuk block source kode harus diawali dengan tanda kurung kurawal { dan akhirnya dengan tanda tutup kurung kurawal } seperti yang terdapat pada coding program. Dalam program ini terdapat public:, **public** adalah tingkatan akses yang berfungsi memberikan hak akses secara umum kepada kelas – kelas turunannya maupun terhadap lingkungan luar didalam program seperti terdapat dalam program **class Makhluk {** artinya public memberikan akses secara umum kepada kelas turunan (subclass).

Didalam program juga terdapat **void** (fungsi tanpa nilai balik), fungsi yang **void** sering disebut juga **prosedur**. Disebut void karena fungsi tersebut tidak mengembalikan suatu nilai keluaran yang didapat dari hasil proses fungsi tersebut contohnya **void berkembang();** Dalam program hewan juga terdapat Fungsi dari **cout** yaitu untuk mencetak ke layar monitor. Contohnya **cout<<" Berkembang biak"<<endl;**, **cout<<" Bergerak berpindah tempat"<<endl;**, **cout<<" Berlari sangat kencang seperti angin"<<endl;**. Dan pada program terakhir terdapat fungsi **getch** yaitu membaca tipe karakter tanpa menampilkan karakter tersebut kelayar dan program siap dijalankan.

### **BAB III**

### **KESIMPULAN**

Relasi antar kelas-kelas ini disebut dengan **relasi “is-a”**. **Superclass** (“kelas dasar” atau “kelas induk”) merupakan kelas yang lebih general dalam relasi “is-a”. **Subclass** (“kelas turunan” atau “kelas anak”) merupakan kelas yang lebih spesifik dalam relasi “is-a”. **Pewarisan** atau **inheritance** adalah relasi “is-a” antar superclass dan subclasses-nya. **Pewarisan tunggal** adalah pewarisan yang hanya mewarisi satu kelas dasar. **Pewarisan jamak** adalah pewarisan yang mewarisi lebih dari satu kelas dasar. Tipe data **float** menyatakan bilangan real dengan ketelitian rendah, byte yang diperlukan 4 byte, dan jangkauan nilai numerik  $3.4E-28$  s/d  $3.4E-38$  (positif atau negatif). Tipe data **double** menyatakan bilangan real dengan ketelitian tinggi, byte yang diperlukan 8 byte, dan jangkauan nilai numerik  $1.7E-308$  s/d  $1.7E-308$  (positif atau negatif). Tanda kutip tunggal (,) berguna untuk karakter tunggal. Contohnya: „a“, „z“. Tanda kutip ganda (") berguna untuk mengapit untaian beberapa karakter. Contohnya: “Laporan sudah selesai”.

## DAFTAR PUSTAKA

Diana. Asisten Praktikum Teknik Informatika. Algoritma dan Pemrograman II. 2020. *Cover laporan hasil dan modul Algoritma dan pemrograman II*. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.

Java Mung. 2009. *Memahami konsep pewarisan*.

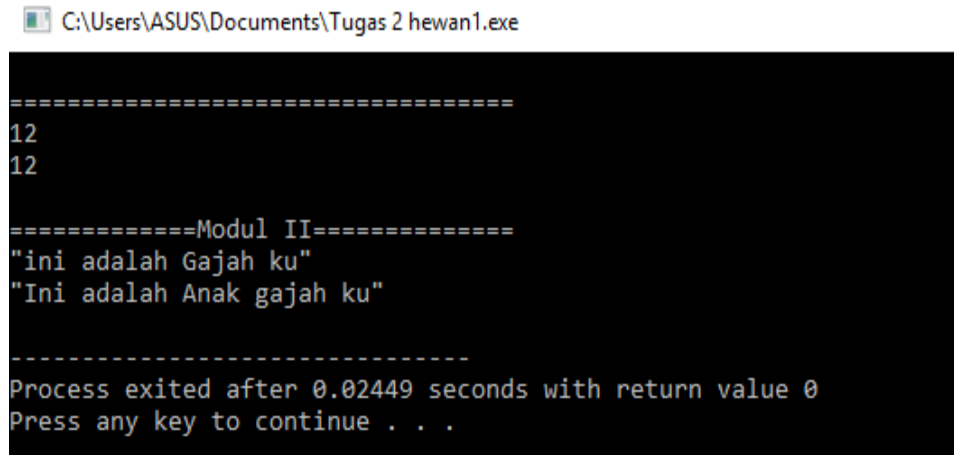
[http://mungjava.blogspot.com/2009/05/memahami konsep inheritance- pewarisan.html](http://mungjava.blogspot.com/2009/05/memahami-konsep-inheritance-pewarisan.html).

Rizki Aditya. 2011. Tutorial pemrograman berorientasi objek dan pewarisan.

[http://www.adityarizki.net/2011/03/tutorial-pemrograman-berorientasi-obyek- inheritance-pewarisan/](http://www.adityarizki.net/2011/03/tutorial-pemrograman-berorientasi-obyek-inheritance-pewarisan/)

## LAMPIRAN

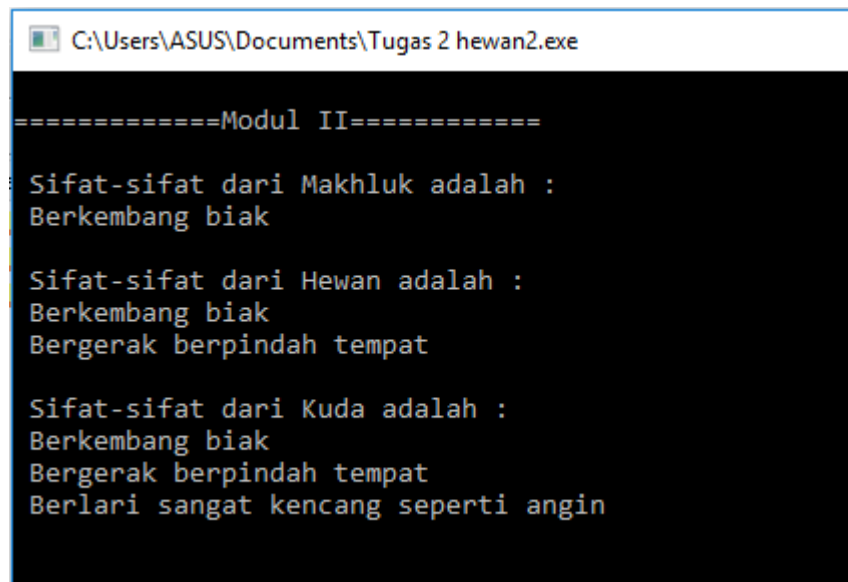
↵ enter



```
C:\Users\ASUS\Documents\Tugas 2 hewan1.exe

=====
12
12
=====Modul II=====
"ini adalah Gajah ku"
"Ini adalah Anak gajah ku"
-----
Process exited after 0.02449 seconds with return value 0
Press any key to continue . . .
```

Gambar 2.1. Output program hewan 1



```
C:\Users\ASUS\Documents\Tugas 2 hewan2.exe

=====Modul II=====

Sifat-sifat dari Makhluk adalah :
Berkembang biak

Sifat-sifat dari Hewan adalah :
Berkembang biak
Bergerak berpindah tempat

Sifat-sifat dari Kuda adalah :
Berkembang biak
Bergerak berpindah tempat
Berlari sangat kencang seperti angin
```

Gambar 2.2. Output program hewan 2