

MODUL PRAKTIKUM ALGORITMA PEMROGRAMAN II



**Tim Penyusun:
Dosen Teknik Informatika**

**JURUSAN TEKNIK
INFORMATIKA FAKULTAS
TEKNIK UNIVERSITAS
PALANGKARAYA 2020**

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**

Cover
bersihkan



NAMA : Ardi Zulfan Nainggolan
NIM : 193010503014
KELAS : B
**MODUL : I (DASAR PEMROGRAMAN
BERORIENTASI OBJEK)**

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

2019

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

Pemrograman berorientasi objek (object-oriented programming disingkat OOP) merupakan pemograman yang berorientasikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya.

Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program, dan digunakan luas dalam teknik piranti lunak skala besar. Lebih jauh lagi, pendukung OOP mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya, dan pendekatan OOP lebih mudah dikembangkan dan dirawat.

Class adalah kumpulan atas definisi data dan fungsi-fungsi dalam suatu unit untuk suatu tujuan tertentu. Sebagai contoh 'class of dog' adalah suatu unit yang terdiri atas definisi-definisi data dan fungsi-fungsi yang menunjuk pada berbagai macam perilaku/turunan dari anjing. Sebuah class adalah dasar dari modularitas dan struktur dalam pemrograman berorientasi object. Sebuah class secara tipikal sebaiknya dapat dikenali oleh seorang non-programmer sekalipun terkait dengan domain permasalahan yang ada, dan kode yang terdapat dalam sebuah class sebaiknya (relatif) bersifat mandiri dan independen (sebagaimana kode tersebut digunakan jika tidak menggunakan OOP). Dengan modularitas, struktur dari sebuah program akan terkait dengan aspek-aspek dalam masalah yang akan diselesaikan melalui

program tersebut. Cara seperti ini akan menyederhanakan pemetaan dari masalah ke sebuah program ataupun sebaliknya.

Enkapsulasi digunakan untuk memastikan pengguna sebuah objek tidak dapat mengganti keadaan dalam dari sebuah objek dengan cara yang tidak layak; hanya metode dalam objek tersebut yang diberi izin untuk mengakses keadaannya. Setiap objek mengakses interface yang menyebutkan bagaimana objek lainnya dapat berinteraksi dengannya. Objek lainnya tidak akan mengetahui dan tergantung kepada representasi dalam objek tersebut.

Attribute adalah segala sesuatu (data)yang berhubungan dengan karakteristik objek dalam setiap kelas. Tiap atribut memiliki nilai berbeda untuk setiap objek. Method merupakan tingkah laku dari suatu objek, jika bersifat static berarti tingkah laku semua objek dalam kelas tersebut.

Identifier adalah suatu pengenalan atau pengidentifikasi yang kita deklarasikan agar kompiler dapat mengenalinya. Identifier sendiri dapat berupa nama variabel, konstanta fungsi, kelas, template maupun namespace. Biasanya identifier ini terdiri dari 3 bentuk, yaitu privat public dan protected

1. public: anggota pribadi dari kelas dasar hanya dapat diakses oleh anggota yang kelas dasar.
2. private: anggota publik dari kelas dasar dapat diakses oleh anggota yang kelas dasar, anggota kelas turunan serta anggota yang berada di luar kelas dasar dan kelas turunan.
3. protected: anggota dilindungi dari kelas dasar dapat diakses oleh anggota kelas dasar serta anggota kelas turunan.

Constructor adalah anggota class yang bertugas melakukan inisialisasi obyek (instance) dari suatu class C. Constructor mempunyai nama yang sama dengan nama class, dan tidak mempunyai return value. Sebuah class dapat mempunyai lebih dari satu constructor. Constructor yang tidak mempunyai argumen, disebut default constructor, sebaliknya constructor yang mempunyai lebih dari satu argumen adalah non-default constructor. Constructor dengan satu default argument tetap merupakan sebuah default constructor.

BAB II

PEMBAHASAN

2.1 Program Pertama

Ditambah Penjelasan

```
#include<stdio.h>

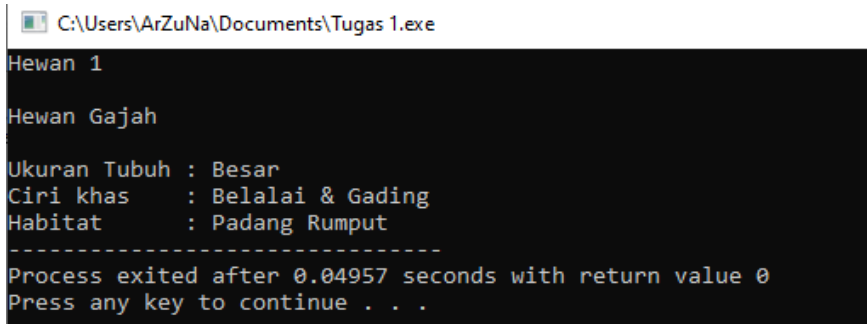
int main(void)
{
    char nama[20] = "Gajah";
    printf("Hewan 1 \n\n");
    printf("Hewan Gajah \n\n");
    printf("Ukuran Tubuh : Besar \n");
    printf("Ciri khas      : Belalai & Gading \n");
    printf("Habitat         : Padang Rumput");

    return 0;
}
```

Penjelasan

- `#include` berfungsi untuk mengimpor fungsi-fungsi yang sudah di definisikan pada header file.
- Pada contoh di atas, kita mengimpor file `stdio.h`. File ini berisi deklarasi fungsi-fungsi dasar yang kita butuhkan untuk membuat program C, seperti `printf()`.
- `int main` adalah tipe data yang akan dikembalikan. Maka di dalam fungsi `main()`, wajib kita sertakan `return 0`. Artinya, fungsi `main` akan mengembalikan nilai 0 setelah selesai dieksekusi.
- `Char` berfungsi untuk menampilkan karakter tunggal.
- `Printf` digunakan untuk membuat sebuah output berupa tampilan.
- `return` berfungsi untuk mengakhiri eksekusi dari function tersebut, dan `return` juga dapat memberikan nilai pada saat akhir dari function kepada pemanggil.

Output :



```
C:\Users\ArZuNa\Documents\Tugas 1.exe
Hewan 1
Hewan Gajah
Ukuran Tubuh : Besar
Ciri khas      : Belalai & Gading
Habitat         : Padang Rumput
-----
Process exited after 0.04957 seconds with return value 0
Press any key to continue . . .
```

Gambar 2.1 program 1

2.2

Program Kedua

I

```
#include<stdio.h>

int main(void)
{

    char nama[20] = "Hiu";
    printf("Hewan 2 \n\n");
    printf("Hewan Jerapah \n\n");
    printf("Jenis Hewan   : Herbivora \n");
    printf("Ciri Khas       : Leher Panjang \n");
    printf("Habitat          : Padang Rumput \n");
    printf("Jumlah Kaki      : 4 \n");
    printf("Jumlah mata      : 2");

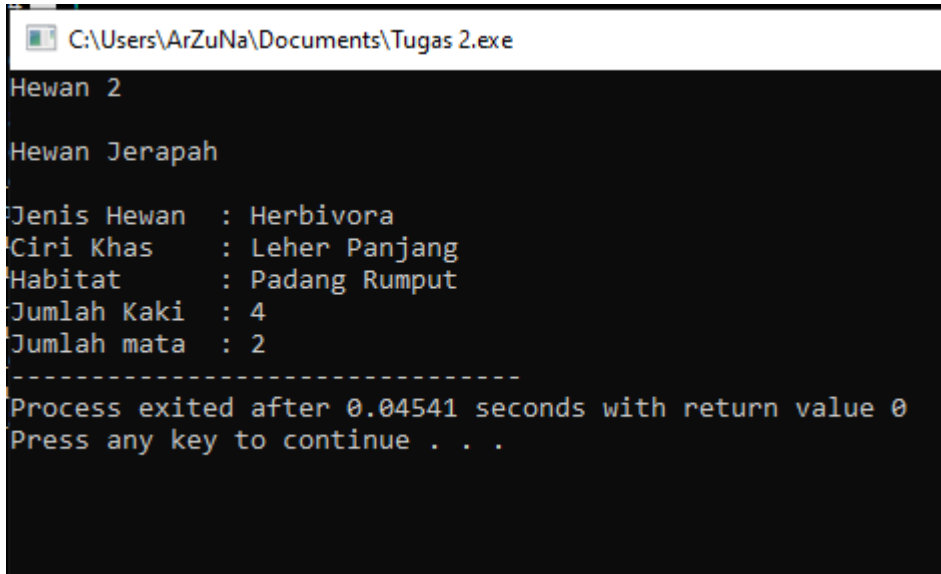
    return 0;

}
```

Penjelasan :

- Jangan terlalu jauh*
- `#include` berfungsi untuk mengimpor fungsi-fungsi yang sudah di definisikan pada header file.
 - Pada contoh di atas, kita mengimpor file `stdio.h`. File ini berisi deklarasi fungsi-fungsi dasar yang kita butuhkan untuk membuat program C, seperti `printf()`.
 - `int main` adalah tipe data yang akan dikembalikan. Maka di dalam fungsi `main()`, wajib kita sertakan `return 0`. Artinya, fungsi `main` akan mengembalikan nilai 0 setelah selesai dieksekusi.
 - `Char` berfungsi untuk menampilkan karakter tunggal.
 - `Printf` digunakan untuk membuat sebuah output berupa tampilan.
 - `return` berfungsi untuk mengakhiri eksekusi dari function tersebut, dan `return` juga dapat memberikan nilai pada saat akhir dari function kepada pemanggil.
- Justify*

Output :



```
C:\Users\ArZuNa\Documents\Tugas 2.exe
Hewan 2
Hewan Jerapah
Jenis Hewan : Herbivora
Ciri Khas : Leher Panjang
Habitat : Padang Rumput
Jumlah Kaki : 4
Jumlah mata : 2
-----
Process exited after 0.04541 seconds with return value 0
Press any key to continue . . .
```

Gambar 2.2 program 2

• • • •

BAB III

KESIMPULAN

Pada dasarnya class adalah sebuah bluepint (cetak biru) atau yang bisa disebut arsitektur dari object. Artinya, kita bisa membuat banyak objek dari suatu class. Kelas (sintaks: class) adalah konsep struktur data yang diperluas: seperti struktur data, data tersebut dapat berisi data anggota, namun juga berisi fungsi sebagai anggota. Sedangkan, objek adalah sebuah instantiasi sebuah kelas. Objek sendiri mempunyai identitas (nama), pada umumnya memiliki pengetahuan baik tentang dirinya maupun objek lain dan memiliki ketrampilan untuk melakukan sesuatu. Dalam program, pengetahuan dari suatu objek disebut dengan state atau attribute atau field. Sedangkan ketrampilan dan perilakunya disebut dengan behaviour / function / method. Jika kita ingat variabel, maka kelas akan menjadi tipe data lalu objek akan menjadi nama variabelnya. Namun intinya: kelas adalah representasi benda(object) dari dunia nyata.

DAFTAR PUSTAKA

Dosen Teknik Informatika. Algoritma dan Pemrograman I. 2019. Modul Praktikum Algoritma dan Pemrograman II. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.

Fathvizard. 2012. *Class, Method, Attribute (Tugas OOP1)*.
<https://fvizard.wordpress.com/2012/10/17/class-method-attribute/> (diakses
pada : Selasa, 09 Maret 2019 pada pukul: 19.40)

<https://ilcmoetz.wordpress.com/materi/mengenal-fungsi-printf-scanf-dan-variabel-di-c/>

<https://www.belajarcpp.com/tutorial/cpp/tipe-data/>

<https://www.petanikode.com/c-syntak/>

LAMPIRAN

```
C:\Users\ArZuNa\Documents\Tugas 1.exe
Hewan 1
Hewan Gajah
Ukuran Tubuh : Besar
Ciri khas    : Belalai & Gading
Habitat      : Padang Rumput
-----
Process exited after 0.04957 seconds with return value 0
Press any key to continue . . .
```

Gambar 2.1 Program 1

```
C:\Users\ArZuNa\Documents\Tugas 2.exe
Hewan 2
Hewan Jerapah
Jenis Hewan  : Herbivora
Ciri Khas    : Leher Panjang
Habitat      : Padang Rumput
Jumlah Kaki   : 4
Jumlah mata  : 2
-----
Process exited after 0.04541 seconds with return value 0
Press any key to continue . . .
```

Gambar 2.2 program 2