

Tipe: B	Assessment 1 Pemrograman Berorientasi Objek Fakultas Ilmu Terapan, Universitas Telkom Semester Ganjil Tahun Ajaran 2014-2015	
--------------------	---	---

Materi Kajian 1

<p>No: 1 Materi: Tipe Data, Operasi Aritmatik Sederhana, Program Sederhana Poin: 25</p>
<p>Dibaca sebuah jarak dalam centimeter. Konversikanlah masukan menjadi jarak tempuh dalam kilometer, meter dan centimeter. 1 km = 100 m, 1 m = 100 centimeter. Contoh: 123.456 cm = 12 km 34 m 56 cm.</p> <p>Kodekanlah kasus di atas dengan menggunakan bahasa pemrograman Java. Gunakan kelas Scanner atau bantuan argumen untuk meminta masukan.</p>

<p>No: 2 Materi: Pengkondisian, Perulangan dan Logika Poin: 25</p>
<p>Buatlah sebuah program untuk menghitung jumlah bilangan ganjil dari 1 sampai N. Bilangan N merupakan masukan pengguna. Contoh: Masukkan batas bilangan: 10 (input) 1+3+5+7+9 = 25 (output) Jumlah bilangan ganjil 1-10 adalah 25 (output)</p> <p>Masukkan batas bilangan: 7 (input) 1+3+5+7 = 16 (output) Jumlah bilangan ganjil 1-7 adalah 16 (output)</p> <p>Kodekan kasus di atas dengan menggunakan bahasa pemrograman Java. Gunakan kelas Scanner untuk meminta masukan pengguna.</p>

<p>No: 3 Materi: Array Poin: 25</p>
<p>Buatlah sebuah program yang meminta masukan dari pengguna berupa n bilangan. Bilangan-bilangan tersebut disimpan ke dalam sebuah array, dan berdasarkan array tersebut akan ditentukan nilai maksimal dan indeks nilai maksimal pada array tersebut.</p> <p>Contoh: Masukkan jumlah bilangan: 3 (input) Bilangan ke-1: 4 (input) Bilangan ke-2: 72 (input) Bilangan ke-3: 12 (input) Nilai maksimal adalah 72 pada indeks ke-2 (output)</p>

Tipe: B	Assessment 1 Pemrograman Berorientasi Objek Fakultas Ilmu Terapan, Universitas Telkom Semester Ganjil Tahun Ajaran 2014-2015	
--------------------	---	---

Kodekan kasus di atas dengan menggunakan bahasa pemrograman Java. Gunakan kelas Scanner untuk meminta masukan pengguna. Gunakan prosedur/fungsi untuk mencari nilai maksimal dan indeksinya.

No: 4
Materi: Konsep Kelas dan Objek
Poin: 25

Seorang anak ingin membuat pendataan buku yang ia miliki. Buku tersebut memiliki judul, pengarang, tahun terbit dan jumlah halaman. Buku tersebut akan dibangun menggunakan konsep enkapsulasi. Terdapat 2 konstruktor untuk mengeset atribut dari buku. Konstruktor pertama digunakan untuk mengeset judul, pengarang dan tahun terbit. Konstruktor kedua digunakan untuk mengeset judul dan pengarang.

Rancanglah class diagram dari kasus di atas, dan kodekan menggunakan bahasa pemrograman java. Bentuklah 2 objek buku dengan data yang diasumsikan sendiri.

Materi Kajian 2

No: 1
Materi: Array of Object
Poin: 25

Seorang mahasiswa ingin mendata nilai yang pernah ia dapatkan. Nilai tersebut memiliki atribut nama mata kuliah, jumlah SKS, nilai mutu, dan semester ambil. Pendataan dilakukan dengan menggunakan konsep enkapsulasi.

Kodekanlah bentuk pendataan nilai menggunakan bahasa pemrograman java. Bentuklah 3 objek dengan data yang diasumsikan sendiri. Ketiga objek tersebut tersimpan ke dalam array of object. Tampilkanlah 3 data objek yang dibentuk dengan mengakses array tersebut, dan hitunglah IPK dari 3 objek nilai.

No: 2
Materi: Inheritance
Poin: 15

Diketahui sebuah program memiliki kelas-kelas dengan kode sebagai berikut

```

class Bird{
    protected String tweet;
    public Bird(){
        System.out.println("This is Bird");
    }
}
class Parrot extends Bird{
    public Parrot(String tweet){
        super.tweet = tweet;
        System.out.println("This is Parrot Species");
    }
}

```

```

public void tweet(){
    System.out.println(tweet);
}
}

```

Jika terdapat sebuah driver class sebagai berikut,

```

public class Main {
    public static void main(String[] args) {
        Parrot p = new Parrot("I'm a Parrot");
        p.tweet();
    }
}

```

Apakah program dapat dijalankan? Jika dapat, tuliskan dan jelaskan keluaran dari program di atas.

Dan... jika terdapat sebuah driver class sebagai berikut,

```

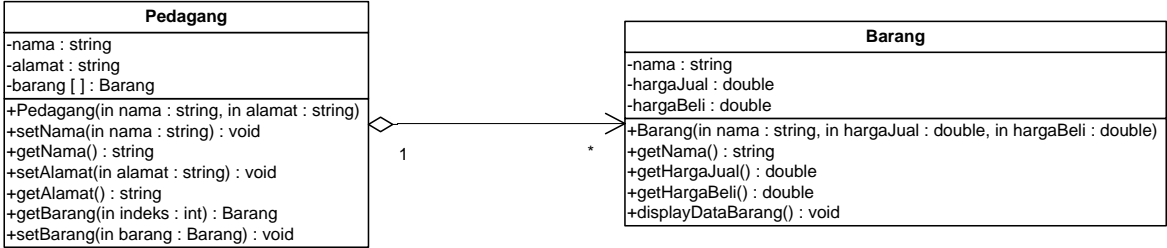
public class Main {
    public static void main(String[] args) {
        Bird p = new Parrot("I'm a Parrot");
        p.tweet();
    }
}

```

Apakah program dapat dijalankan? Jika dapat, tuliskan dan jelaskan keluaran dari program di atas.

No: 3
Materi: Hubungan antar kelas (Agregasi)
Poin: 25

Sebuah asosiasi pedagang ingin mencatat barang dagangan dan pedagang yang menjual dagangannya. Untuk ini dibangun sebuah sistem dengan class diagram sebagai berikut:



```

classDiagram
    class Pedagog {
        -nama : string
        -alamat : string
        -barang [] : Barang
        +Pedagog(in nama : string, in alamat : string)
        +setNama(in nama : string) : void
        +getNama() : string
        +setAlamat(in alamat : string) : void
        +getAlamat() : string
        +getBarang(in indeks : int) : Barang
        +setBarang(in barang : Barang) : void
    }
    class Barang {
        -nama : string
        -hargaJual : double
        -hargaBeli : double
        +Barang(in nama : string, in hargaJual : double, in hargaBeli : double)
        +getNama() : string
        +getHargaJual() : double
        +getHargaBeli() : double
        +displayDataBarang() : void
    }
    Pedagog o-- "*" Barang

```

Tipe: B	Assessment 1 Pemrograman Berorientasi Objek Fakultas Ilmu Terapan, Universitas Telkom Semester Ganjil Tahun Ajaran 2014-2015	
--------------------	---	---

Keterangan:

- a) Pedagang memiliki konstruktor nama dan alamat.
- b) Gunakan array dengan kapasitas 100 atau array list untuk menyimpan objek Barang pada Pedagang
- c) Keuntungan didapatkan dari harga jual dikurangi harga beli.
- d) Barang memiliki konstruktor berupa nama dan harga beli dan harga jual.
- e) Relasi merupakan hubungan asosiasi.

Kodekanlah class diagram di atas dan driver class-nya menggunakan bahasa pemrograman java dengan data yang diasumsikan sendiri. Tampilkan data pedagang dan barang yang dijualnya, serta hitung keuntungan yang didapat oleh tiap pedagang.

No: 4

Materi: Abstract dan Interface

Poin: 15

Terdapat sebuah interface "InterfesA" dengan method "methodNSatu", "methodNDua" dan "methodNTiga". InterfesA mengambil sifat-sifat dari interface "InterfesB" yang mempunyai method "methodASatu" dan "methodADua". Selain itu terdapat interface "InterfesC" dengan method berupa function "methodB" yang mengembalikan tipe integer dan konstanta (nama konstanta bebas) dengan tipe data String yang berisikan kata "Halo...".

Kelas "AsliA" mengimplements "InterfesA" dan "InterfesC" dengan method tambahan "MethodRSatu". Selain itu, terdapat kelas abstrak "AbstrakB" dengan method abstrak "methodAbstA" berupa prosedur, "methodAbstB" berupa function dengan tipe data balikan integer, dan method non-abstrak "methodRealB" yang merupakan function dengan tipe data balikan integer, "methodRealB" memiliki sebuah parameter masukan "x" dengan tipe data integer.

Kelas "AbstrakA" merupakan kelas yang meng-extends "AbstrakB" dengan method abstrak "methodAbstC" dan meng-override "methodRealB" dan "methodAbstB".

Kelas "AsliB" meng-extends kelas "AbstrakA" dengan method terkandung adalah method yang wajib di-override oleh kelas ini. Pada kelas "AsliB" terdapat atribut static "statikSatu" (tipe String) dengan visibilitas protected dan "statikDua" dengan tipe integer. Kelas "AsliB" merupakan kelas dengan keyword final. Selain itu terdapat atribut "atributReal" dengan tipe "AsliA". Hubungan antara kelas "AsliB" dan "AsliA" adalah whole-part (aggregation).

Tentukan method yang harus di-override oleh kelas "AsliA" dan "AsliB".