

**FINAL EXAMINATION / PEPERIKSAAN AKHIR
SEMESTER II – SESSION 2018 / 2019
PROGRAM KERJASAMA**

COURSE CODE : DDWC 2623 / DDPC 2623 WC
KOD KURSUS

COURSE NAME : OBJECT - ORIENTED PROGRAMMING USING JAVA
NAMA KURSUS PENGATURCARAAN BERORIENTASIKAN OBJEK MENGGUNAKAN JAVA

YEAR / PROGRAMME : 2 DDWC / 2 DDPC / 2 DDWZ / 2 DDPZ
TAHUN / PROGRAM

DURATION : 2 HOURS AND 30 MINUTES
TEMPOH

DATE : APRIL 2019
TARIKH

**INSTRUCTIONS:
ARAHAN**

- The question paper consists of 3 sections: A, B and C: Answer **ALL** questions in the question paper.
*Kertas soalan terdiri daripada 3 bahagian: A, B dan C. Jawab **SEMUA** soalan dalam kertas soalan.*
- Candidates are required to follow all instructions given out by examination invigilators.
Calon dikehendaki mematuhi semua arahan daripada penyelia peperiksaan.

(You are required to write your name and your lecturer's name on your answer script)
(Pelajar dikehendaki tuliskan nama dan nama pensyarah pada skrip jawapan)

NAME / NAMA	:
I.C NO. / NO. K/PENGENALAN	:
YEAR / COURSE TAHUN / KURSUS	:
COLLEGE NAME NAMA KOLEJ	:
LECTURER'S NAME NAMA PENSYARAH	:

This examination paper consists of 12 pages including the cover.
Kertas soalan ini mengandungi 12 muka surat termasuk kulit hadapan.



PUSAT PROGRAM KERJASAMA

**PETIKAN DARIPADA PERATURAN AKADEMIK
ARAHAN AM - PENYELEWENGAN AKADEMIK**

1. SALAH LAKU SEMASA PEPERIKSAAN

1.1 Pelajar tidak boleh melakukan mana-mana salah laku peperiksaan seperti berikut :-

- 1.1.1 memberi dan/atau menerima dan/atau memiliki sebarang maklumat dalam bentuk elektronik, bercetak atau apa jua bentuk lain yang tidak dibenarkan semasa berlangsungnya peperiksaan sama ada di dalam atau di luar Dewan Peperiksaan melainkan dengan kebenaran Ketua Pengawas; atau
- 1.1.2 menggunakan maklumat yang diperolehi seperti di atas bagi tujuan menjawab soalan peperiksaan; atau
- 1.1.3 menipu atau cuba untuk menipu atau berkelakuan mengikut cara yang boleh ditafsirkan sebagai menipu semasa berlangsungnya peperiksaan; atau
- 1.1.4 lain-lain salah laku yang ditetapkan oleh Universiti (seperti membuat bising, mengganggu pelajar lain, mengganggu Pengawas menjalankan tugasnya).

2. HUKUMAN SALAH LAKU PEPERIKSAAN

2.1 Sekiranya pelajar didapati telah melakukan pelanggaran mana-mana peraturan peperiksaan ini, setelah diperakukan oleh Jawatankuasa Peperiksaan Fakulti dan disabitkan kesalahannya, Senat boleh mengambil tindakan dari mana-mana satu yang berikut :-

- 2.1.1 memberi markah SIFAR (0) bagi keseluruhan keputusan peperiksaan kursus yang berkenaan (termasuk kerja kursus); atau
 - 2.1.2 memberi markah SIFAR (0) bagi semua kursus yang didaftarkan pada semester tersebut.
- 2.2 Jawatankuasa Akademik Fakulti boleh mencadangkan untuk diambil tindakan tatatertib mengikut peruntukan Akta Universiti dan Kolej Universiti, 1971, Kaedah-kaedah Universiti Teknologi Malaysia (Tatatertib Pelajar-pelajar), 1999 bergantung kepada tahap kesalahan yang dilakukan oleh pelajar.
- 2.3 Pelajar yang didapati melakukan kesalahan kali kedua akan diambil tindakan seperti di perkara 2.1.2 dan dicadang untuk diambil tindakan tatatertib mengikut peruntukan Akta Universiti dan Kolej Universiti, 1971, Kaedah-kaedah Universiti Teknologi Malaysia (Tatatertib Pelajar-pelajar), 1999.

SECTION A [30 Marks] / BAHAGIAN A [30 Markah]

Instruction: Answer all questions with the most suitable answer in the space provided.

Arahan: Jawab semua soalan dengan jawapan yang paling sesuai pada ruang jawapan yang disediakan.

1. Briefly describe **THREE (3)** advantages of using object - oriented approach for developing software. **[6 M]**
Terangkan secara ringkas TIGA (3) kebaikan menggunakan pendekatan berorientasikan objek bagi pembangunan perisian.

Answer / Jawapan:

- a) _____

- b) _____

- c) _____

2. Draw **UML notations** for the following elements and relationships: **[6 M]**

Lukis notasi UML bagi elemen dan perhubungan yang berikut:

- a) Actor / Aktor
b) Class / Kelas
c) Aggregation relationship / Hubungan Pengagregatan
d) Generalization relationship / Hubungan Generalisasi

Answer / Jawapan:

- a)

b)

c)

d)

3. Name methods provided by Java to parse the input getting via dialog box into the following data type: **[6 M]**
Namakan kaedah – kaedah yang disediakan oleh Java untuk menukarkan input yang didapati melalui kotak dialog kepada jenis data berikut:

- a) double
b) boolean

- c) integer
- d) float

Answer / Jawapan:

- a) _____
- b) _____
- c) _____
- d) _____

4. `Character ch = new Character('i');`

You are given a Java statement as above. Using methods from **Character** class, write the syntax to do the following operations: [6 M]

*Anda diberi pernyataan Java seperti di atas. Menggunakan kaedah daripada kelas **Character**, tulis sintak untuk melakukan operasi berikut:*

- a) To check either the character is a digit or not.
Untuk semak samada aksara tersebut adalah digit atau tidak.
- b) To change the character to uppercase letter.
Untuk tukar aksara tersebut kepada huruf besar.
- c) To check either the character is a letter or not.
Untuk semak samada aksara tersebut adalah huruf atau tidak.

Answer / Jawapan:

- a) _____
- b) _____
- c) _____

5. Briefly explain the meaning of the following terms: [6 M]

Terangkan secara ringkas maksud istilah yang berikut:

- a) Abstract class / *Kelas abstrak*
- b) Abstract method / *Kaedah abstrak*
- c) **Overriding method** / *Kaedah overriding*

Answer / Jawapan:

- a) _____

- b) _____

- c) _____

SECTION B [50 Marks] / BAHAGIAN B [50 Markah]

Instruction: Answer all questions in the space provided.

Arahan: Jawab semua soalan pada ruang jawapan yang disediakan.

1. You are required to develop Information Management System for a new college. The requirement specification is as follows. A college has one or more departments. Each department has a lecturer who is a department head. A lecturer can be the head of no more than one department. Every student may attend three to five courses per semester and every course may have any numbers of students. For every course must be taught by at least one lecturer. Each lecturer may teach at most four courses per semester. **Draw UML class diagram** which consists of all classes in your system, relationships among classes and multiplicity specification. **[7 M]**

*Anda dikehendaki membangunkan Sistem Pengurusan Maklumat bagi kolej baru. Spesifikasi keperluan seperti berikut. Kolej mempunyai satu atau lebih jabatan. Setiap jabatan mempunyai pensyarah yang mana adalah sebagai ketua jabatan. Pensyarah boleh menjadi ketua satu jabatan sahaja. Setiap pelajar boleh hadir tiga hingga lima kursus per semester dan setiap kursus boleh mempunyai sebarang bilangan pelajar. Bagi setiap kursus mesti diajar oleh sekurang – kurang seorang pensyarah. Seorang pensyarah boleh mengajar paling banyak empat kursus per semester. **Lukis gambarajah kelas UML** yang mengandungi semua kelas bagi sistem yang akan anda bangunkan, perhubungan antara kelas dan spesifikasi 'multiplicity'.*

Answer / Jawapan:

4. You are given a program fragment as below. Write Java statements to perform the following tasks by using methods from the **String**, **StringBuffer** and **Character**.

*Anda diberikan keratan aturcara seperti di bawah. Tulis pernyataan java untuk menjalankan tugas – tugas yang berikut dengan menggunakan kaedah – kaedah dari **String**, **StringBuffer** dan **Character**.*

```
String message1 = "hi there!";  
String message2 = new String("learn java 2019");  
StringBuffer message3 = new StringBuffer("2019 new year");  
count(message2);
```

- a) Display the length of the message1 and the message2. [2 M]

Papar panjang message1 dan message2.

Answer / Jawapan:

- b) Display the position index of the third character of 'r' in message2. [2 M]

Papar indeks kedudukan aksara pertama 'r' dalam message2.

Answer / Jawapan:

- c) Display the result of concatenation of the content message1 with the message2. [2 M]

Papar hasil pencantuman kandungan message1 dengan message2.

Answer / Jawapan:

- d) Display the result of insertion word " happy " at end of message3. [2 M]

Paparkan hasil penyelitan perkataan " happy " pada akhir kandungan message3.

Answer / Jawapan:

- e) Write a method to count and display the number of occurrences of 'a' in message2 using the following method header: `public static int countLetter(String msg)` [5 M]

Tulis satu kaedah untuk kira dan paparkan bilangan kewujudan 'a' dalam message2 menggunakan pengepala kaedah berikut: `public static int countLetter(String msg)`

Answer / Jawapan:

5. Fill in the blanks with Java code based on the given comments in the program below. Your program should be coded to produce the output as given at the end of program. [7 M]

Isikan tempat kosong dengan kod Java berdasarkan komen yang diberikan dalam aturcara di bawah. Aturcara anda perlu dikodkan untuk menghasilkan output seperti diberikan pada akhir aturcara.

```
public interface BasketItem {
    public String getDescription();
}
// Class Book will implement getDescription() from BasketItem
// Kelas Book akan melaksanakan getDescription() dari BasketItem
public class Book _____ {
    private String title;
    private String author;
    // Constructor that accepts two String arguments and will assign them to
    // the Book's data fields.
    // Konstruktor yang menerima dua argument String dan akan diumpukkannya
    // kepada medan data buku.
    public Book(String title, String author) {
        _____
        _____
    }
    // Prepare implementation for getDescription() that returns a description
    // of Book as formatted in the output below.
    // Sediakan pelaksanaan bagi getDescription() yang mengembalikan
    // keterangan Book seperti diformatkan dalam output di bawah.
    public _____ getDescription() {
        _____
        _____
    }
}
public class TestBook{
    public static void main(String[] args) {
        // Create two objects of Book with a given title and author values.
        // Cipta dua objek Book dengan nilai - nilai title dan author diberikan.
        _____
        _____
        // Display output as given below / Papar output seperti di bawah
        _____
        _____
    }
}
```

Sample output / Contoh Output:

Long Long Ago, by John P.
Introduction to java Programming, by Daniel Liang.

SECTION C [20 Marks] / BAHAGIAN C [20 Markah]

Instruction: Write a complete program based on the question requirement given.

Arahan: Tulis aturcara lengkap berdasarkan keperluan soalan yang diberikan.

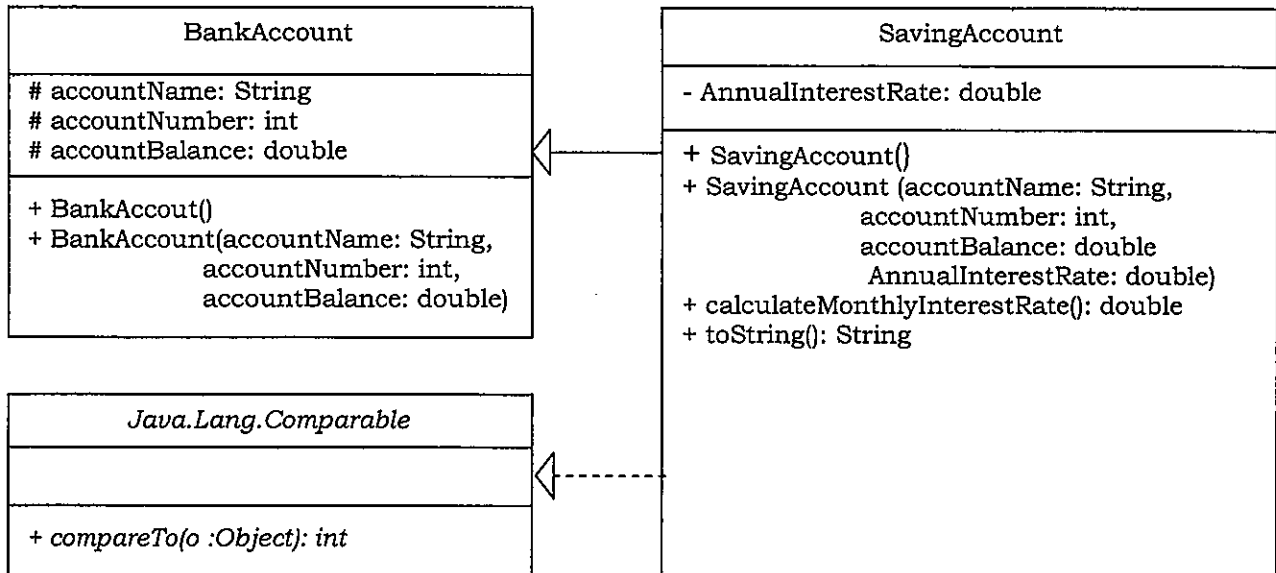


Figure 1: Class Diagram / Rajah 1: Gambarajah kelas

Write a complete program for the **SavingAccount** class and test program based on the **Figure 1** and the following requirements: Create a **SavingAccount** class that derived from **BankAccount**. This class inherits all data fields from its superclass and also has its own data field named **AnnualInterestRate**. This data field is used to hold the value of annual interest rate. Besides that, the **SavingAccount** has two constructors, method **calculateMonthlyInterestRate()** and method **toString()**. The method **calculateMonthlyInterestRate()** returns the calculation result of monthly interest rate by using the following formula: Annual interest rate / 12 * account balance. The method **toString()** will return the information of **SavingAccount** object. The class **SavingAccount** will implement the **compareTo()** method from interface **Comparable** to compare the monthly interest rate of the two objects. This method returns 1 if the monthly interest rate of the first object is larger than the second object, return -1 if smaller than the second object and return 0 if both objects have the equal monthly interest rate.

Write a test program that creates two objects of **SavingAccount** with any values of the **accountName**, **accountNumber**, **accountBalance** and **AnnualInterestRate**. Display information for the both objects including the monthly annual interest and the result of comparison.

Tulis satu aturcara lengkap bagi kelas **SavingAccount** dan ujian berdasarkan **Rajah 1** dan keperluan yang berikut: Cipta kelas **SavingAccount** yang dihasilkan dari **BankAccount**. Kelas ini mewarisi semua medan data dari kelas supernya dan juga mempunyai medan data sendiri yang bernama **AnnualInterestRate**. Medan data ini digunakan untuk memegang nilai kadar bunga tahunan. Selain itu, **SavingAccount** mempunyai dua konstruktor, kaedah **calculateMonthlyInterestRate()** dan kaedah **toString()**. Kaedah **calculateMonthlyInterestRate()** mengembalikan hasil pengiraan kadar bunga bulanan dengan menggunakan formula berikut: Kadar bunga tahunan / 12 * baki akaun. Kaedah **toString()** akan kembalikan maklumat objek

Mukasurat ini sengaja dibiarkan kosong

[This page is purposely left blank]

Mukasurat ini sengaja dibiarkan kosong

[This page is purposely left blank]