

**FINAL EXAMINATION / PEPERIKSAAN AKHIR  
SEMESTER I – SESSION 2022 / 2023  
PROGRAM KERJASAMA**

COURSE CODE : DDWD 2623  
KOD KURSUS

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

YEAR / PROGRAMME : 2 DDWD  
TAHUN / PROGRAM

DURATION : 2 HOURS AND 30 MINUTES  
TEMPOH

DATE : DECEMBER 2022 / JANUARY 2023  
TARIKH

**INSTRUCTIONS:  
ARAHAN**

1. The question paper consists of 3 sections: A, B and C.  
*Kertas soalan terdiri daripada 3 bahagian: A, B dan C.*
2. Answer **ALL** questions in the question paper.  
*Jawab **SEMUA** soalan dalam kertas soalan.*
3. 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 KOLEJ	:	.....
LECTURER'S NAME NAMA PENSYARAH	:	.....

This examination paper consists of 13 pages including the cover.  
*Kertas soalan ini mengandungi 13 muka surat termasuk kulit hadapan.*



**UTM**  
UNIVERSITI TEKNOLOGI MALAYSIA

School of  
Professional and  
Continuing  
Education  
(SPACE)

## **PUSAT PRGORAM 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/Bilik Peperiksaan melainkan dengan kebenaran Ketua Pengawas; atau
- 1.1.2. menggunakan maklumat yang diperoleh 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 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 answers in the spaces provided.

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

1. List and explain any four (4) basic principle of object-oriented . [6 M]

*Nyatakan dan jelaskan empat (4) prinsip asas berorientasikan objek.*

**Answer / Jawapan:**

- a. \_\_\_\_\_  
\_\_\_\_\_
- b. \_\_\_\_\_  
\_\_\_\_\_
- c. \_\_\_\_\_  
\_\_\_\_\_
- d. \_\_\_\_\_  
\_\_\_\_\_

2. A private data field cannot be directly accessed by an object. Name a method that can enable:- [3 M]

*Medan data persendirian tidak boleh dicapai secara terus oleh objek. Namakan satu kaedah yang membolehkan:-*

- a. a private data to be accessed.  
*data persendirian boleh dicapai.*
- b. a private data to be updated.  
*data persendirian boleh dikemaskini.*

**Answer / Jawapan:**

- a. \_\_\_\_\_
- b. \_\_\_\_\_

3. Name the keyword or operator provided by Java to : [3 M]

*Namakan kata kunci atau operator yang disediakan oleh Java untuk:*

- a. Make sure a class cannot be extended.  
*Memastikan kelas tidak boleh dilanjutkan.*
- b. Invoke a superclass constructor.  
*Panggil kaedah sesuatu objek.*

**Answer / Jawapan:**

- a. \_\_\_\_\_
- b. \_\_\_\_\_

4. State the method that should be used to perform the following tasks:

[6 M]

*Nyatakan kaedah yang perlu digunakan untuk menjalankan tugas-tugas berikut:*

a. To compare a content of two **String** objects.

*Untuk membandingkan kandungan dua objek **String**.*

b. To convert a character to uppercase in a **String** object.

*Untuk menukar satu aksara kepada huruf besar dalam objek **String**.*

c. To check either the specific character is digit in a **Character** object.

*Untuk menyemak samada aksara adalah nombor dalam objek **Character**.*

d. To delete characters from start index to end index in a **StringBuffer** object.

*Untuk menghapus aksara-aksara dari indek mula hingga indek akhir dalam objek **StringBuffer**.*

**Answer / Jawapan:**

- a. \_\_\_\_\_
- b. \_\_\_\_\_
- c. \_\_\_\_\_
- d. \_\_\_\_\_

5. Briefly explain the purpose of the following **Object** class's methods:

[3 M]

*Terangkan secara ringkas tujuan kaedah-kaedah kelas **Object** yang berikut:*

a. toString() method / kaedah toString()

b. equals() method / kaedah equals()

**Answer / Jawapan:**

- a. \_\_\_\_\_
- b. \_\_\_\_\_

6. Describe the difference between method overloading and method overriding.

[3 M]

*Terangkan perbezaan antara kaedah 'overloading' dan kaedah 'overriding'.*

**Answer / Jawapan:**

---

---

---

7. Determine whether the following statements are **true** or **false**:

[3 M]

*Tentukan samada pernyataan-pernyataan berikut adalah **benar** atau **palsu**:*

a. An interface can only be extended to one interface.

*Satu antaramuka hanya boleh dilanjutkan kepada satu antaramuka.*

b. A class that contains abstract methods must be defined as abstract.

*Satu kelas yang mengandungi kaedah abstrak mesti ditakrifkan sebagai abstrak*

c. A subclass can be an abstract class if its superclass is non-abstract.

*Sub kelas boleh menjadi kelas abstrak jika kelas supernya adalah bukan abstrak.*

**Answer / Jawaban:**

- a. \_\_\_\_\_
- b. \_\_\_\_\_
- c. \_\_\_\_\_

8. State why interface's variables in Java are static and final.

**[3 M]**

*Nyatakan mengapa pemboleh ubah antaramuka dalam Java adalah static dan final.*

**Answer / Jawaban:**

---

---

---

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

**Instruction: Answer all questions in the spaces provided.**

**Arahan: Jawab semua soalan pada ruang jawapan yang disediakan.**

1. You are requested to develop a Car Management Sharer System for Webstar Sdn.Bhd. Draw a **class diagram** without attributes and methods based on the following requirements:

"Clients must register to be a member of car sharer. The members of car sharers can register as many journeys as they want and also can share their journey with other registered car sharers. In order to share the similar journey with other car sharers, the system will search the journey information based on the start and end destination of the car sharer address. Then the system will match up the car sharers and formalize things with a sharing agreement."

**[6 M]**

*Anda diminta untuk membangunkan Sistem Pengurusan Kongsi Kereta untuk Webstar Sdn. Bhd. Lukis satu gambarajah kelas tanpa atribut dan kaedah berdasarkan keperluan berikut:*

*"Pelanggan mesti mendaftar untuk menjadi ahli kongsi kereta. Ahli kongsi kereta boleh mendaftar sebanyak perjalanan yang mereka inginkan dan juga boleh berkongsi perjalanan mereka dengan ahli berdaftar yang lain. Untuk berkongsi perjalanan yang sama dengan ahli kongsi kereta lain, sistem akan mencari maklumat perjalanan berdasarkan mula dan tamat destinasi alamat ahli kongsi kereta. Kemudian sistem akan padankan ahli kongsi kereta yang berkongsi perjalanan dan semua perkara akan dijelaskan dengan perjanjian perkongsian."*

**Answer / Jawapan:**

2. Rewrite the following **switch..case** statements to the **if..else if** statements.

[6 M]

Tulis semula pernyataan **switch..case** berikut kepada pernyataan **if .. else if**.

```
switch(choice) {  
    case 1:  
        charges = months * 40;  
        break;  
    case 2:  
    case 3:  
    case 4:  
        charges = months * 20;  
        break;  
    case 5:  
        charges = months * 10;  
    case 6:  
        charges = months * 5;  
        break;  
    default:  
        charges = -1;  
        break;  
}
```

**Answer / Jawapan:**

---

---

---

---

---

---

---

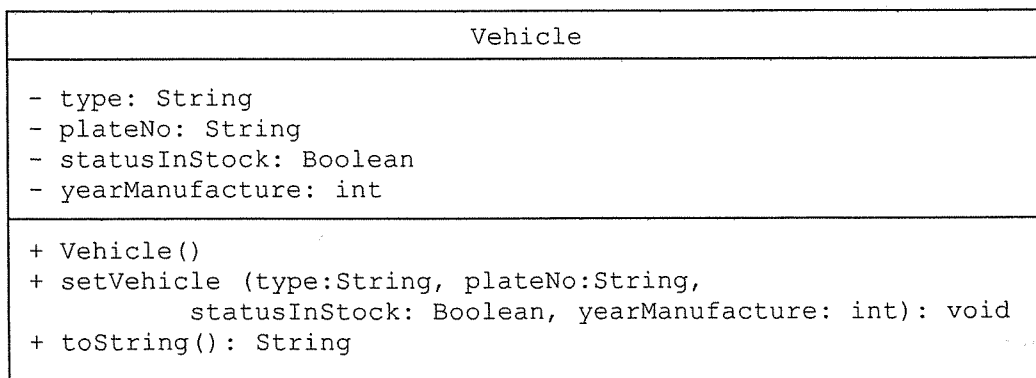
---

---

---

3. Based on the UML diagram for the class `Vehicle` shown below, write a class test program to do the following:

Berdasarkan gambarajah UML untuk kelas `Vehicle` yang ditunjukkan di bawah, tulis program kelas ujian untuk melakukan yang berikut:



a. Create an array of ten `Vehicle` objects.

[2 M]

Cipta satu tatasusunan sepuluh objek `Vehicle`.

**Answer / Jawapan:**

---

- b. Assign `type`, `plateNo`, `statusInStock` and `yearManufacture` to the each object based on the user's entered via console. [8 M]

*Umpukan nilai `type`, `plateNo`, `statusInStock` dan `yearManufacture` kepada setiap objek tersebut berdasarkan kemasukan pengguna melalui konsol.*

**Answer / Jawapan:**

---

---

---

---

---

---

---

---

---

---

- c. Display the object information by invoking its `toString()` method. [3 M]

*Papar maklumat objek dengan memanggil kaedah `toString()`.*

**Answer / Jawapan:**

---

---

4. Answer all questions (a) to (f) based on the program fragment given as follow:

*Jawab semua soalan (a) hingga (f) berdasarkan keratan aturcara diberikan seperti berikut:*

```
StringBuffer s1 = new StringBuffer("JAVA IS VERSATILE PROGRAMMING LANGUAGE.");  
String s2 = "Java is pure object - oriented";  
int count = findNumberOfLetter(s2, 'e');
```

- a. Write Java statement to extract a word "ject" from `s2`. [2 M]

*Tuliskan pernyataan Java untuk mendapatkan perkataan "ject" dari `s2`.*

**Answer / Jawapan:**

---

- b. Write Java statement to delete a word "PROG" from `s1`. [2 M]

*Tulis pernyataan Java untuk menghapuskan perkataan "PROG" dari `s1`.*

**Answer / Jawapan:**

---

- c. Write Java statement to insert the word "Portable" at position 10 in `s1`. [2 M]

*Tulis pernyataan Java untuk memasukkan perkataan "Portable" pada kedudukan 10 dalam `s1`.*

**Answer / Jawapan:**

---



- d. Write Java statement to replace all the letters of o with ZZ in s2. [2 M]

Tulis pernyataan Java untuk menggantikan semua huruf o dengan ZZ dalam s2.

Answer / Jawapan:

---

- e. Write Java statement to retrieve the first letter of 'o' in the s2. [2 M]

Tulis pernyataan Java untuk memaparkan huruf 'o' yang pertama dalam s2.

Answer / Jawapan:

---

- f. Write a method that finds and return the number of occurrence of 'e' letter in the s2 using the following header: [4 M]

Tulis satu kaedah yang mencari dan memulangkan bilangan kewujudan huruf 'e' dalam s2 menggunakan pengepala kaedah berikut:

```
public static int findNumberOfLetter(String msg2, char letter)
```

Answer / Jawapan:

---

---

---

---

---

---

---

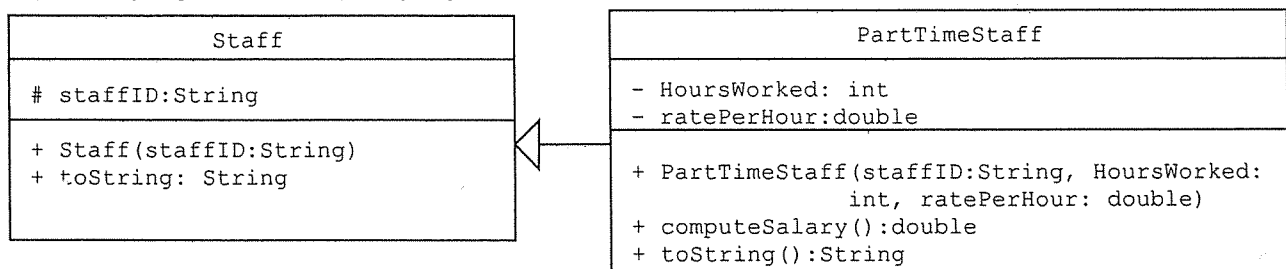
---

---

---

5. Write Java statements to describe PartTimeStaff class based on the UML class diagram and given requirements as follow:

Tulis pernyataan Java untuk menerangkan kelas PartTimeStaff berdasarkan rajah kelas UML dan keperluan yang diberikan seperti yang berikut:



- a. Define class PartTimeStaff. / Takrif kelas PartTimeStaff. [2 M]

Answer / Jawapan:

---

- b. Write a parameterized constructor for a class `PartTimeStaff` that accepts three arguments for each its data fields. [3 M]

*Tulis satu konstruktor berparameter bagi kelas `PartTimeStaff` yang menerima tiga argumen bagi setiap medan datanya.*

**Answer / Jawapan:**

---

---

---

---

---

- c. Write `computeSalary()` method that should be implemented in a `PartTimeStaff` class to compute the salary of `PartTimeStaff` object by multiplying `hoursWorked` with `ratePerHour`. [3 M]

*Tulis kaedah `computeSalary()` yang perlu dilaksanakan dalam kelas `PartTimeStaff` untuk mengira gaji objek `PartTimeStaff` dengan mendarabkan `hoursWorked` dengan `ratePerHour`.*

**Answer / Jawapan:**

---

---

---

- d. Write `toString()` method for the class `PartTimeStaff` that overrides `toString()` defined in the `Staff` class. This method will return the information of `PartTimeStaff` that contains `staffID`, `hourWorked`, `ratePerHour` and salary. [3 M]

*Tulis kaedah `toString()` bagi kelas `PartTimeStaff` yang 'override' `toString()` yang ditakrifkan dalam kelas `Staff`. Kaedah ini mengembalikan maklumat `PartTimeStaff` yang mengandungi `staffID`, `hourWorked`, `ratePerHour` dan gaji.*

**Answer / Jawapan:**

---

---

---

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.

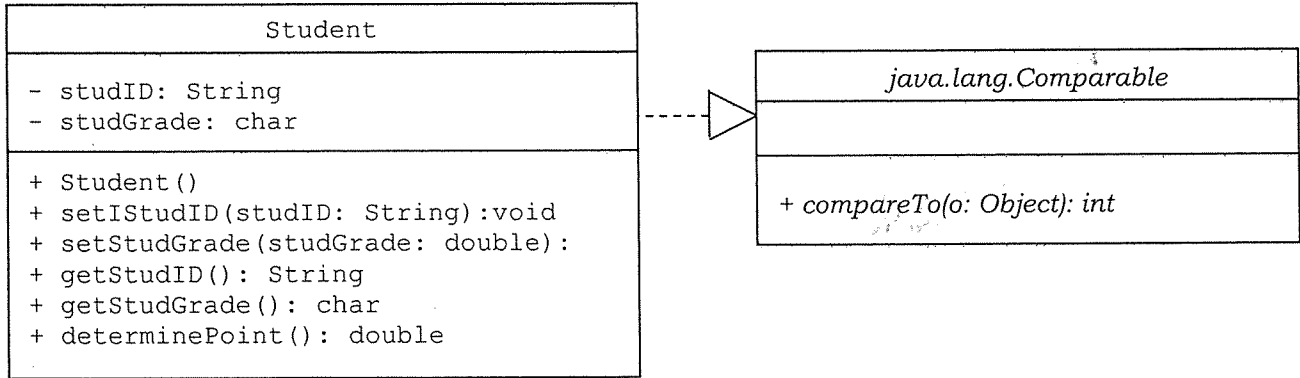


Figure 1: Shows the relationship between class `Student` and interface `Comparable`.

Rajah 1: Menunjukkan perhubungan antara kelas `Student` dan antaramuka `Comparable`.

Write a program for a `Student` class based on the Figure 1. The `Student` class has two attributes, one constructor, two mutator methods, two accessor methods and `determinePoint()` methods. The `determinePoint()` method will return a point obtained based on the grade. The grade and its associated point can be referred on a table below. The `Student` class implements `compareTo()` method from `java.lang.Comparable` interface to compare the point obtained for two `Student` objects. Write a test program that creates two `Student` objects with values of attributes are entered by the user. Display the information of all objects includes the point comparison result.

Tulis aturcara bagi kelas `Student` berdasarkan Rajah 1. Kelas `Student` mempunyai dua atribut, satu konstruktor, dua kaedah 'mutator', dua kaedah 'accessor' dan kaedah `determinePoint()`. Kaedah `determinePoint()` akan memulangkan mata nilai yang diperolehi berdasarkan kepada gred. Gred dan mata nilai berkaitan boleh dirujuk pada jadual di bawah. Kelas `Student` melaksanakan kaedah `compareTo()` dari antaramuka `java.lang.Comparable` untuk membandingkan mata nilai yang diperolehi bagi dua objek `Student`. Tulis satu aturcara ujian yang mencipta dua objek `Student` dengan nilai-nilai atribut yang dimasukkan oleh pengguna. Papar maklumat semua objek termasuklah hasil perbandingan mata nilai.

Grade / Gred	Point / Mata Nilai
A	4.00
B	3.00
C	2.00
D	1.00