

UNIVERSITI MALAYA
UNIVERSITY OF MALAYA

PEPERIKSAAN IJAZAH SARJANA MUDA SAINS KOMPUTER / SARJANA MUDA
TEKNOLOGI MAKLUMAT
EXAMINATION FOR THE DEGREE OF BACHELOR OF COMPUTER SCIENCE / BACHELOR
OF INFORMATION TECHNOLOGY

SESI AKADEMIK 2012/2013 : SEMESTER I
ACADEMIC SESSION 2012/2013 : SEMESTER I

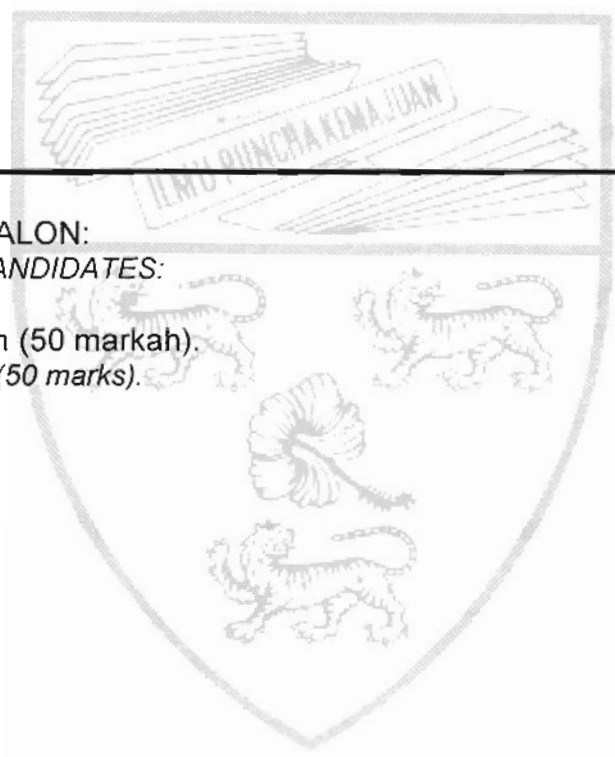
WXES1116 : Pengaturcaraan I
Programming I

Januari 2013
January 2013

Masa: 2 jam
Time: 2 hours

ARAHAN KEPADA CALON:
INSTRUCTIONS TO CANDIDATES:

Jawab **SEMUA** soalan (50 markah).
Answer **ALL** questions (50 marks).



(Kertas soalan ini mengandungi 4 soalan dalam 5 halaman yang dicetak)
(This question paper consists of 4 questions on 5 printed pages)

1. Aturcara dalam fail **Q1.java** mengandungi banyak ralat. Betulkan kesemua ralat tersebut.

(Salin fail **Q1.java** dari direktori akaun peperiksaan anda. Selepas program dibetulkan, namakan fail tersebut sebagai [**matricNumberQ1.java**; contoh: **WEK100001Q1.java**] dan salin fail ini ke direktori akaun peperiksaan.)

The program in the Q1.java file contains many errors. Correct all the errors.

(Copy the Q1.java file from your exam account directory. After the program has been corrected, name the file as [matricNumberQ1.java; example: WEK100001Q1.java] and copy this file to your exam account directory.)

```
// Filename: Q1.java
public class Q1 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.out);
        char letter;
        String temp;
        int count=100;
        do {
            System.out.print("Enter a letter (Q to quit): ");
            letter = s.nextChar();
            if (letter='Q')
                break;
            else
                count--;
        } while (letter!='Q')
        System.out.printf("The total of number of character you enter is : %s", count);
    }
}
```

(8 markah/marks)

2. Tulis satu aturcara yang menerima kemasukan lima nama pelajar. Kemudian paparkan nama pelajar tersebut dalam urutan abjad dengan mengabaikan huruf kecil/besar.

(Simpan aturcara tersebut dalam fail **Main.java**. Salin fail ini ke direktori akaun peperiksaan anda dan namakan semula sebagai [**matricNumberQ2.java**; contoh: **WEK100001Q2.java**].

Write a program that accepts the input of five student names. Then display the student names in alphabetical order ignoring the case.

(Save the program in Main.java file. Copy this file to your exam account directory and rename as [matricNumberQ2.java; example: WEK100001Q2.java].

Contoh output:

Sample output:

```
Enter student name:Tan Hong Tex
Enter student name:Raja Kunan
Enter student name:Ahmad Zairif
Enter student name:Chan Swee Ling
Enter student name:Mutu Samy
The student names in alphabetical order are:
Ahmad Zairif
Chan Swee Ling
Mutu Samy
Raja Kunan
Tan Hong Tex
```

(10 markah/marks)

3. Tuliskan satu aturcara yang membaca maklumat matawang asing daripada fail **Q3.txt** dan menukarnya ke Ringgit Malaysia dalam dua tempat perpuluhan menggunakan jadual kadar pertukaran asing di bawah. Output dipaparkan di skrin dan ditulis ke satu fail bernama **ringgit.txt**.

(Salin fail **Q3.txt** dari direktori akaun peperiksaan anda. Simpan aturcara dalam fail **Main.java**. Salin fail **Main.java** ini ke akaun direktori peperiksaan anda and namakan ia semula sebagai [**matricNumberQ3.java**; contoh: **WEK100001Q3.java**].

Write a program that reads foreign currency information from **Q3.txt** file and converts it to Malaysian Ringgit in two decimal places using the foreign exchange rates table below. Output is displayed on screen and written to a file named **ringgit.txt**.

(Copy the **Q3.txt** file from your exam account directory. Save the program in **Main.java** file. Copy this **Main.java** file to your exam account directory and rename it as [**matricNumberQ3.java**; example: **WEK100001Q3.java**].

Currency	Code	Ringgit
1 Australian Dollar	AUD	RM 3.25
1 Singapore Dollar	SG	RM 2.53
1 Euro	EU	RM 4.03
1 US Dollar	USD	RM 3.11

Table 1: Foreign Exchange Rates

Contoh output:

Sample output:

```
EU 51.2 = RM 206.34
SG 32.7 = RM 82.73
EU 46.9 = RM 189.01
USD 30.2 = RM 93.92
AUD 22.3 = RM 72.48
SG 100.23 = RM 253.58
```

(12 markah/marks)

4. Reka bentuk satu kelas **Staff** yang terdiri daripada ahli berikut:
- Satu medan untuk nama kakitangan.
 - Satu medan untuk IC kakitangan.
 - Satu pembina dan kaedah pencapai serta kaedah *mutator* yang sesuai.
 - Satu kaedah *toString* yang memaparkan nama kakitangan dan IC.

*Design a **Staff** class that consists of the following members:*

- *A field for the name of the staff.*
- *A field for the IC of the staff.*
- *A constructor and appropriate accessor methods and mutator methods.*
- *A toString method that displays the staff's name and IC.*

(7 markah/marks)

Reka bentuk satu kelas **PermanentStaff** yang mewarisi kelas **Staff**. Kelas **PermanentStaff** sepatutnya mempunyai ahli berikut:

- Satu medan untuk gred kakitangan.
- Satu pembina.
- Kaedah pencapai dan kaedah *mutator* untuk gred kakitangan.
- Kaedah *getSalary* yang mengira gaji berdasarkan jadual di bawah.
- Satu kaedah *toString* yang memaparkan nama kakitangan, IC, gred dan gaji.

*Design a **PermanentStaff** class that extends the **Staff** class. The **PermanentStaff** class should have the following members:*

- *A field for the grade of the staff.*
- *A constructor.*
- *Accessor method and mutator method for the grade of the staff.*
- *A getSalary method that computes salary based on the table below.*
- *A toString method that displays the staff's name, IC, grade and salary.*

Grade	Salary (RM)
A01	5000
A02	4000
A03	3000
A04	2000

Table 2: Salary

(7 markah/marks)

Reka bentuk satu kelas **PartTimeStaff** yang mewarisi kelas **Staff**. Kelas **PartTimeStaff** sepatutnya mempunyai ahli berikut:

- Satu medan untuk bilangan jam bekerja (satu integer).
- Satu pembina.
- Kaedah pencapai dan kaedah *mutator* untuk bilangan jam bekerja.
- Kaedah *getSalary* yang mengira gaji berdasarkan RM50 sejam.
- Satu kaedah *toString* yang memaparkan nama kakitangan, IC dan gaji.

Design a class **PartTimeStaff** that extends the **Staff** class. The **PartTimeStaff** class should have the following members:

- A field for the number of working hours (an integer).
- A constructor.
- Accessor method and mutator methods for the number of working hours.
- A *getSalary* method that computes salary based on RM50 per hour.
- A *toString* method that displays the staff's name, IC and salary.

(6 markah/marks)

Bina satu kelas *Main* menggunakan kod di bawah untuk menguji aturcara tersebut.

(Salin fail **Staff.java**, **PermanentStaff.java** dan **PartTimeStaff.java** ke dalam direktori akaun peperiksaan anda.)

Create a *Main* class using the code below to test the program.

(Copy the **Staff.java**, **PermanentStaff.java** dan **PartTimeStaff.java** files to your exam account directory.)

```
public static void main(String[] args) {
    PermanentStaff p1 = new PermanentStaff("Ahmad Taheri", "801001-14-1765", "A03" );
    System.out.println(p1.toString());
    PartTimeStaff p2 = new PartTimeStaff("Tengku Ali", "811223-08-3301", 23);
    System.out.println(p2.toString());
}
```

Contoh output:

Sample output:

Full Name: Ahmad Taheri

IC: 801001-14-1765

Grade: A03

Salary: RM 3000

Full Name: Tengku Ali

IC: 811223-08-3301

Number of working hours: 23

Salary: RM 1150

TAMAT
END