



MALAYSIAN INSTITUTE OF INFORMATION TECHNOLOGY

**FINAL EXAMINATION
JANUARY 2016 SEMESTER**

SUBJECT CODE : ISB16003
SUBJECT TITLE : OBJECT-ORIENTED PROGRAMMING
LEVEL : BACHELOR
TIME / DURATION : 9.00 am – 11.30 am
(2 ½ HOURS)
DATE : 23 MAY 2016

INSTRUCTIONS TO CANDIDATES

1. Please read the instructions given in the question paper CAREFULLY.
2. This question paper is printed on both sides of the paper.
3. This question paper consists of 4 questions.
4. Answer All questions.
5. Please write your answers on the ANSWER BOOKLET provided.

THERE ARE 8 PAGES OF QUESTIONS, EXCLUDING THIS PAGE.

Question 1

Given two classes which represent types of payment : `CashPayment` and `CreditCardPayment`. A class named `Payment` will serve as the superclass for both types of payments. Define all common methods in `Payment` class, and specify all differing methods in such a way that every subclass is forced to implement them. No objects of type `Payment` will be created.

a. Implement the following in `Payment` class.

- i. Declare `Payment` class.
- ii. Declare `amount` as a variable which is only accessible by its subclass.
- iii. Define a constructor by an `amount`.
- iv. Define a constructor by a `paymentID` and `amount`.

(8 marks)

b. Implement the following in `CashPayment` class.

- i. Declare `CashPayment` class.
- ii. Declare a `tenderedAmount` as a variable which is accessible by its class only.
- iii. Define a constructor by `paymentID` and `amount`.
- iv. Change is calculated based on `amountTendered` and `amount`. Write the complete method definition for `calculateChange` method where `amountTendered` is the input and `change` is the output of the method.

(10 marks)

c. The test class is needed to test question 1(a) and 1(b). Implement the following in `TestPayment` class.

- i. Instantiate an object called `cash` of class `CashPayment` with parameters of `paymentID=1001, amount=2000`
- ii. Write the `toString` method to display the following :
Payment ID: 1001, Amount: RM1000, Change: RM200
[Hint: Assume that the `toString` method in superclass is already declared]
- iii. Is the following statement valid? Explain why.

```
Payment pay = new Payment (1011,3000);
```

(7 marks)

Question 2

a. Once a class is defined, you can declare reference variables of class type. The following question is based on the `Subject` class. The class contains all private data members named `subject ID`, `subject name` and `credit hour`. Re-draw the diagram at **Figure 1.0** and fill-up necessary data based on the following java statement.

[Hint: You can use one drawing for all the questions]

- i. `Subject mySubject = new Subject("ISB16003")`
- ii. `mySubject.setSubjectName("OOP");`
- iii. `System.out.println(mySubject.getCreditHour());`

(4 marks)

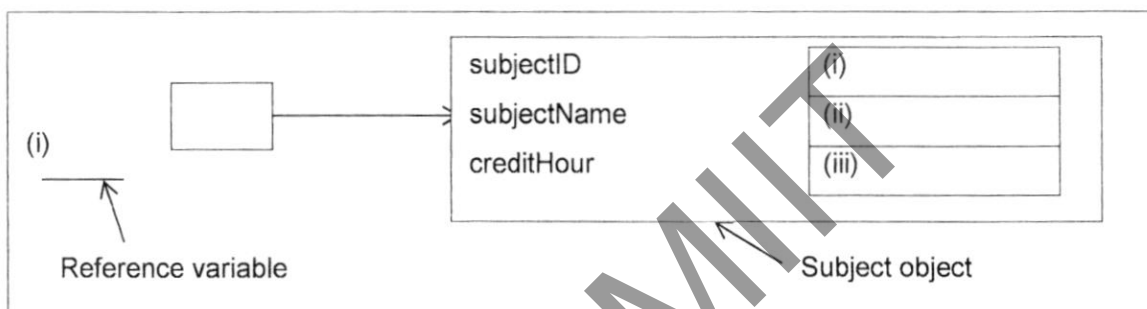


Figure 1.0

b. The following question is based on **Snippet 1.0**.

```

1 System.out.print("Please enter pay rate : ");
2 pay_rate=scanner.nextDouble();
3 if(pay_rate > 25) throw new PayRateException(pay_rate);
    
```

Snippet 1.0

i. Write the suitable user-defined exception class for **Figure 2.0**.

(5 marks)

```

Please enter pay rate : 45.0
Error : The pay rate 45.0 exceeds RM25.00
    
```

Figure 2.0

ii. Write the catch statement for **Figure 3.0**.

(2 marks)

```

Please enter pay rate : twenty five
Error: wrong data type
    
```

Figure 3.0.

c. Produce the output of **Listing 1.0**.

(5 marks)

```
1 public class ThermalReading {
2     public double celcius;
3
4     public ThermalReading() {
5     }
6
7     public ThermalReading(double celcius) {
8         this.celcius=celcius;
9     }
10
11    public double CelToKelvin()
12    {
13        return celsius + 273.15;
14    }
15    }//end of ThermalReading.java
16
17
18    public class TemperatureTest {
19
20    public static void main(String[] args)
21    {
22        ThermalReading t1 = new ThermalReading();
23        ThermalReading t2 = new ThermalReading(100);
24
25        System.out.println(t1.celcius+" celcius = "+t1.CelToKelvin()+
26            " kelvin.");
27        System.out.println(t2.celcius+" celcius = "+t2.CelToKelvin()+
28            " kelvin.");
29        t2.celcius = 50.0;
30
31        System.out.println(t2.celcius+" t2.celcius");
32    }//end main
33    }//end class
```

Listing 1.0

d. Answer the following question based on the **Listing 2.0**.

```
1 public class NumericalValues
2 {
3 public NumericalValues()
4 { System.out.print("Passing negative value : ");
5 }
6
7 public NumericalValues(int num)
8 {System.out.println("Passing positive value : " + num);
9 }
10 }//end class
11
12
13 public class PositiveNumericValues extends NumericalValues
14 {
15 public PositiveNumericValues()
16 { this(1);
17 }
18
19 public PositiveNumericValues(int x)
20 { super(x);
21 System.out.println("****");
22 }
23 }//end class
24
25
26
27 public class NegativeNumericValues extends NumericalValues
28 {
29 public NegativeNumericValues()
30 { this(-1);
31 }
32
33 public NegativeNumericValues(int x)
34 { super(x);
35 System.out.println(x);
36 }
37
38 public void print()
39 { System.out.println("Printing continues...");
40 }
41 }//end class
42
43
44 public class TestOutput
45 {
46 public static void main(String[] args)
47 {
48 NegativeNumericValues nv = new NegativeNumericValues(-1);
49 nv.print();
50 new PositiveNumericValues();
51 }
52 }//end class
```

Listing 2.0.

- i. Write the output for **Listing 2.0**. (5 marks)
- ii. Explain line number 16, 34,49 and 50. (4 marks)

Question 3

- a. A class named `Fan` contains a method `setSpeed()` that takes an `int` argument . Write the following java statements.
 - i. To create an array of 20 `Fan` objects named `fan`. (2 marks)
 - ii. To assign a speed value of 3 to the first `fan` object. (2 marks)
 - iii. To assign a speed value to the last `fan` object based on user input. [Hint: `Scanner userInput = new Scanner(System.in);`] (2 marks)
- b. The following program segment (**Figure 4.0**) is the composition relationship between `Cat` and `ScientificName`. In `Cat` class, add the two-args constructor and a `toString()` method that will return the scientific name of `cat`. (5 marks)



```

public class ScientificName
{
    private String familyName;
    private String speciesName;
    :
    :
    :
}
public class Cat
{
    private int id;
    :
    :
    :
}
  
```

Figure 4.0

- c. Based on the following programs in **Listing 3.0** and **Listing 4.0**, produce the output when the test program is run. Determine which one is method overloading and method overriding (Briefly explain).

(6 marks)

<pre> class Fish { public void Eat(int i) {System.out.println("It eats " +i+ " times"); } } //end class class Shark extends Fish { public void Eat(int i) { System.out.println("Shark eats " +i+ " times"); } } //end class public class Test { public static void main(String[] args) { Shark a = new Shark(); a.Eat(10); } } //end class </pre>	<pre> class Fish { public void Eat(int i) {System.out.println("It eats "+i+ times"); } } //end class class Shark extends Fish { public void Eat(double i) { System.out.println("Shark eats "+i+ " times"); } } //end class public class Test { public static void main(String[] args) { Shark a = new Shark(); a.Eat(10); } } //end class </pre>
<p>Listing 3.0</p>	<p>Listing 4.0</p>

- d. The following actionPerformed method (**Snippet 2.0**) is based on **Figure 5.0**. Write the complete method definition of calculate method.

(8 marks)

```

public void actionPerformed(ActionEvent e)
{
    String actionCommand = e.getActionCommand();

    int num1 = (Integer.parseInt(jtfNum1.getText().trim()));
    int num2 = (Integer.parseInt(jtfNum2.getText().trim()));
    int result=0;

    if (e.getSource() instanceof JMenuItem) {
        // Handle menuactionCommand item events

        if ("Add".equals(actionCommand))
            result=calculate('+',num1,num2);
        else if ("Subtract".equals(actionCommand))
            result=calculate('-',num1,num2);
        else if ("Multiply".equals(actionCommand))
            result=calculate('*',num1,num2);
        else if ("Divide".equals(actionCommand))
            result=calculate('/',num1,num2);
        else if ("Close".equals(actionCommand))
            System.exit(0);
        jtfResult.setText(String.valueOf(result));
    } //end of JMenuItem
} //end of actionPerformed() method

```

Snippet 2.0

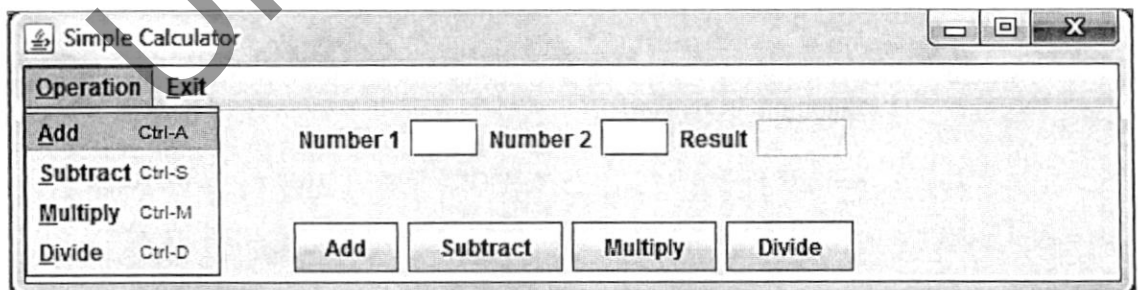


Figure 5.0

Question 4

- a. Based on the following java statement, write the java statement that allows the conversion of kilograms to grams and print the output to "yourFile.txt" (use for loop) as shown at **Figure 6.0**.

(5 marks)

```
PrintWriter outfile ;
        outfile= new PrintWriter(new File("yourFile.txt"));
```

kilograms	grams
10	10000
8	8000
6	6000
4	4000

Figure 6.0

- b. Create TWO Thread classes: ThreadDay and ThreadEven.
- i. The ThreadDay class that creates a thread that prints integer day 1 to day 7. Day 1 starts with Sunday (5 marks)
 - ii. The ThreadEven class creates a thread that prints even numbers from 1 to 7. (4 marks)
- c. The following question is based on the Client and Server class.

- i. The following java statement is to create a buffered output stream to send data to the server in the Client class. Write the java statement to accept values of gender and waist circumference from the keyboard and send both values to Server class.

```
PrintWriter osToServer = new
PrintWriter(connectToServer.getOutputStream(), true);
```

(5 marks)

- ii. In the Server class, write a complete static method definition when the following statement is invoked.

```
String message = DetermineObesity(gender,waist);
```

gender	waist	message
M or m	>90 cm	Obese
F or f	>80 cm	Obese

Otherwise, the message will be "Not Obese".

(6 marks)

END OF QUESTIONS