
UNIVERSITI SAINS MALAYSIA

First Semester Examination
Academic Session 2004/2005

Oktober 2004

CIT502 – Object-Oriented Programming and Software Engineering

Duration : 2 hours

INSTRUCTION TO CANDIDATE:

- Please ensure that this examination paper contains **FOUR** questions in **FIVE** printed pages before you start the examination.
 - Answer **ALL** questions.
 - You can choose to answer either in Bahasa Malaysia or English.
-

ENGLISH VERSION OF THE QUESTION PAPER

1. (a) Explain the difference between:

- (i) Deep and shallow copying
- (ii) Direct and indirect recursion
- (iii) Base case and the general case of a recursive method
- (iv) Private and protected members of a class
- (v) Overloading a method name and overriding a method name

(10/100)

(b) Consider the following method:

```
public static int mystery(int x, double y, char ch)
{
    int u;
    if ('A' <= ch && ch <= 'R')
        return (2 * x + (int) (y));
    else
        return( (int) (2 * y) - x);
}
```

What is the output of the following Java statements?

- (i) `System.out.println(mystery(4, 9.7, 'v'));`
- (ii) `System.out.println(2*mystery(6, 3.7, 'D'));`

(2/100)

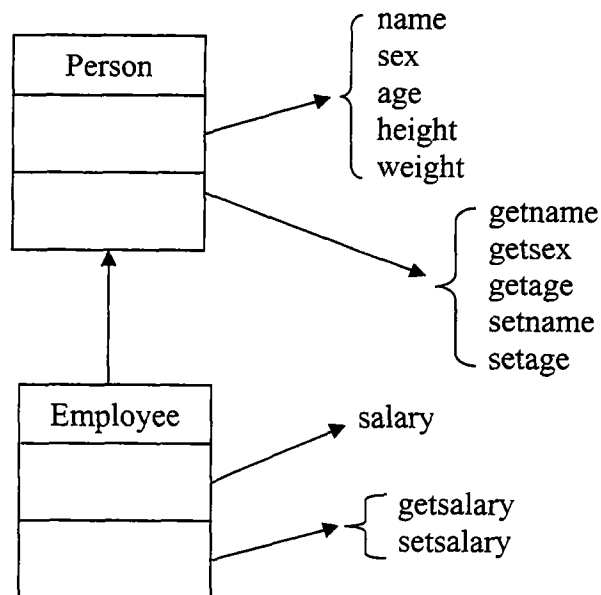
(c) Write a Java program to perform the following steps:

- (i) Prompt the user to input two integers: **firstNum** and **secondNum**. (**firstNum** must be less than the **secondNum**)
- (ii) Output all the even numbers between **firstNum** and **secondNum**.
- (iii) Output the numbers and their square even between **firstNum** and **secondNum**.
- (iv) Output all the prime numbers between **firstNum** and **secondNum**.
- (v) Output the sum of square of odd numbers between **firstNum** and **secondNum**.
- (vi) Output all lowercase letters.

(13/100)

2. (a) Write a method in Java to sort a list. The method will receive the list in the form of an array and the length of the list from the calling program.
(10/100)
- (b) What are the **three (3)** different ways you can implement an interface?
(3 /100)
- (c) Create a class **Rectangle**. The class has attributes length and width, each of which defaults to 1. It has methods that calculate the parameter and the area of the rectangle. It has the get and set method for both length and width. The set method should verify that length and width are each floating point numbers larger than 0.0 and less than 20.0. Include a predicate method **isSquare** which determines whether the rectangle is a square. (Predicate methods typically test a condition and do not modify the object on which they are called.) Write the program to test the class **Rectangle**.
(12/100)
3. (a) Draw a class diagram to represent the following:
- A customer has a first name, a last name, middle names, a daytime telephone number and a home address. A customer can place orders, and a record of all orders is kept. An order has a billing address and a shipping address, which can be the same address – by default the customer's home address is used for both. An order has one or more order items. Each order item is for specific product. Each order item records the quantity of each product ordered, and the subtotal for that item (quantity * unit price). An order also calculates the total amount payable + 17.5% tax. An order can be either unconfirmed, confirmed, shipped or received. A confirmed order has a unique reference, which is a unique integer (e.g., last reference + 1). A record of the order confirmation date is kept, as well as the shipped date and the date the order was received.
(10/100)

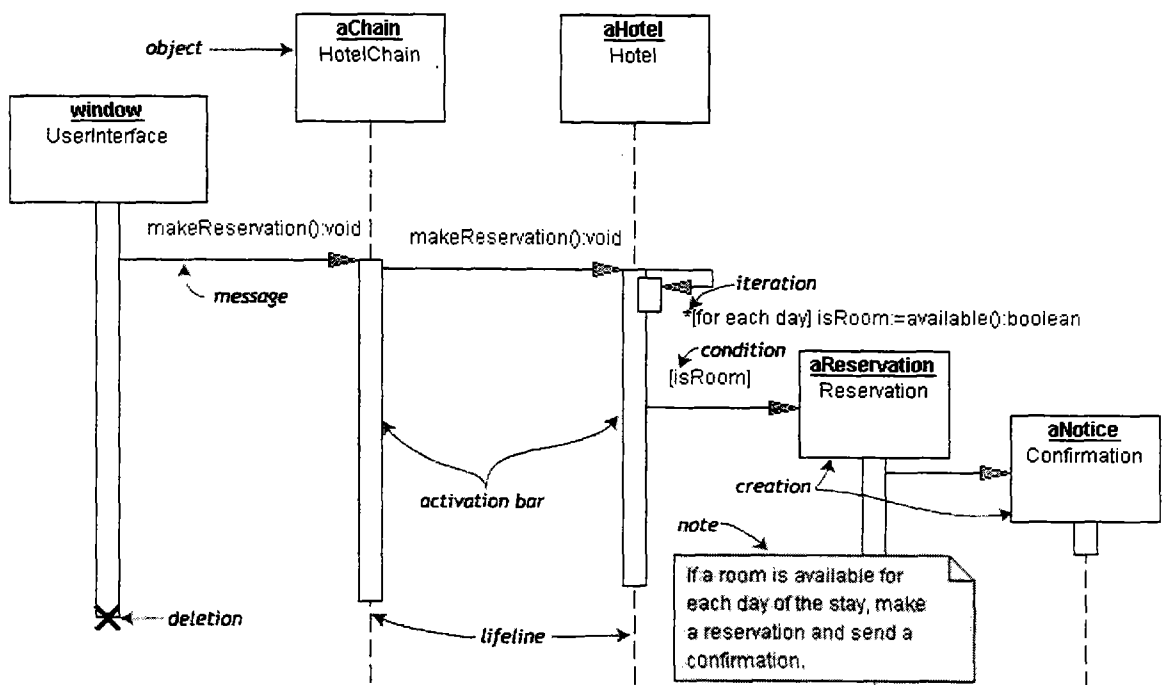
- (b) Consider the following class diagram which shows the inheritance between two different classes **Person** and **Employee** together with its attributes and operations.



Implement the above class diagram into its equivalent JAVA code and write a small program which illustrates the use of the above classes.

(12/100)

4.



- (a) Draw a class diagram for the above sequence diagram.
(8/100)
- (b) Outline the implementation in JAVA for **Reservation** case.
(12/100)
- (c) Write a test scenario for the **Reservation** case.
(8/100)

- oooOooo -