UNIVERSITI SAINS MALAYSIA

First Semester Examination
Academic Session 2003/2004

September/October 2003

CIT502 – Object-Oriented Programming and Software Engineering

Duration : 3 hours

INSTRUCTION TO CANDIDATE:

- Please ensure that this examination paper contains **FOUR** questions in **EIGHT** 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) Indicate whether the following sentence or statement is **True** or **False**.

(i) Value-returning methods must have parameters.

(ii) The word void is a modifier in Java.

(iii) If a member of a class is public, you cannot access it outside the class.

(iv) If a member of a class is a method, it can (directly) access any member of the class.

(v) Once an object is created, you can access the public members of the class using the dot operator.

(vi) You cannot override the default definition of the method `toString` because it is provided by Java.

(vii) The abstract data type specifies the logical properties without the implementation details.

(viii) A single array can hold components of many different data types.

(ix) Key comparisons are also called item comparisons.

(x) In general, if **L** is a sorted list of size **n**, to determine whether an element is in **L**, the binary search makes at most $\mathbf{2*log_2 n + 2}$ **key (item)** comparisons.

(xi) Inheritance implies an "is-a" relationship.

(xii) In single inheritance, the subclass is derived from a single superclass.

(xiii) If you have created an exception class, you can define other exception classes extending the definition of the exception class you created.

(xiv) The class `Container` is the superclass of all the classes designed to provide a GUI.

(xv) The R in RGB stands for Right.

(xvi) You can create instances of Color by mixing red, green, and blue hues in various proportions.

(xvii) Applets use the init method in place of constructors to initialize various GUI components and data members.

(xviii) A method that calls itself is an iterative method.

(xix) Tracing through indirect recursion is generally more tedious than tracing through direct recursion.

(xx) In reality, if you execute an infinite recursive method on a computer, it will execute forever.

(10/25)

(b) Find any error(s) in each of the following and explain how to correct it (them).

(i)
```
import javax.swing.*   // include swing package
```

(ii)
```
panelObject.GridLayout(8, 8); // set GridLayout
```

(iii)
```
container.setLayout(
   new FlowLayout(FlowLayout.DEFAULT));
```

(iv)
```
container.add(eastButton, EAST);    // BorderLayout
```

(5/25)

(c) Trace and show the ouput of the following programs:

(i)
```
public class Test {
   public static void main(string[] args) {
      int i = 1;
      while (i <= 6) {
         xMethod(i, 2)
         i++;
      }
   }

   public static void xMethod(int i, int num) {
      for (int j = 1; j <= i; j++) {
         system.out.print(num + " ");
         num *= 2;
      }

      System.out.println();
   }
}
```

(ii)
```
public calss Test {
   public static void main(String[] args) {
      int sum = xMethod(5);
      System.out.println("Sum is " + sum);
   }

   public static int xMethod(int n) {
      if (n == 1)
         return 1;
      else
         return n + xMethod(n – 1);
   }
}
```

(4/25)

(d) Trace and show the output of the following program:

(i)
```
public class Test {
   public static void main(String[] args) {
      A a = new A(3);
   }
}

class A extends B {
   public A(int t) {
      System.out.println("A's constructor is invoked");
   }
}

class B {
   public B() {
      System.out.println("B's constructor is invoked");
   }
}
```

(ii) What is wrong with the following code?

```
class Test {
   public static void main(String[] args) {
      try {
         Rational r1 = new Rational(3, 4);
         Rational r2 = new Rational(0, 1);
         Rational x = r1.divide(r2);

         int i = 0;
         int y = 2 / i;
      }
      catch (Exception ex) {
         System.out.println("Rational operation error");
      }
      catch (RuntimeException ex) {
         System.out.println("Integer operation error");
      }
   }
}
```

(iii) What is displayed when the following program is run?

```
class Test {
   public static void main(String[] args) {
      try {
         method();
         System.out.println("After the method call");
      }
      catch (RuntimeException ex) {
         System.out.println("Integer operation error");
      }
      catch (Exception ex) {
         System.out.println("Rational operation error");
      }
   }
      static void method() throws Exception {
         try {
            Rational r1 = new Rational(3, 4);
            Rational r2 = new Rational(0, 1);
            Rational x = r1.divide(r2);

            int i = 0;
            int y = 2 / i;
            System.out.println("Welcome to Java");
      }
      catch (RuntimeException ex) {
         System.out.println("Integer oepration error");
      }
      catch (Exception ex) {
         System.out.println("Rational operation error");
      }
   }
}
```

(6/25)

2. (a) Create a class named `Order` that performs order processing of a single item. The superclass has four fields for customer name, customer number, quantity ordered, and unit price. Include `set` and `get` methods for each of the fields. This class also needs methods to compute the total price (quantity times unit price) and to display the fields. Create a subclass that overrides `computePrice()` by adding a shipping and handling charge of RM4.00. Write a program that uses these classes.

(10/25)

(b) A company pays its employees as managers (who receive a fixed weekly salary), hourly workers (who receive a fixed hourly wage for the first 40 hours they work and "time-and-a-half", i.e., 1.5 times their hourly wage, for overtime hours worked), commission workers (who receive RM250 plus 5.7% of their gross weekly sales) or pieceworkers (who receive a fixed amount of money per item for each of the items they produce – each pieceworker in this company works on only one type of item).

Write an application to compute the weekly pay for each employee. Each type of employee has its own pay code: Managers have paycode 1, hourly workers have code 2, commission workers have code 3 and pieceworkers have code 4. Use a `switch` to compute each employee's pay based on that employee's paycode. Use a `CardLayout` to display the appropriate GUI components that allow the user to enter the facts your program needs to calculate each employee's pay based on that employee's paycode.

(15/25)

3. (a) In the V7 car engine, the car battery ignites spark plug, the spark plug burns the fuel inside the piston. The explosion moves the piston. The down stroke of the piston rotates the wheel, and the upstroke of the piston expels the gas via valve. Draw a class diagram in the design stage to reflect the situation mentioned above.

(7/25)

(b) Draw an use case diagram for the e-Learning system that student can submit their assignments and check their course work. They can also download slides. In this system, lecturers can upload slides, post announcement and results of the students.

(6/25)

(c) Model is used to represent the real world. Compare the object model with the dynamic model. Explain each type of them.

(6/25)

(d) Draw a class diagram in the analysis stage to link the following classes using aggregation, inheritance and association when appropriate.

School of Computer Science, Research Group, Lab, Room, Office, Staff, Lecturer, Secretary, Student.

(6/25)

4. (a) Draw a sequence Diagram based on the statements given below. In this case, we identified the Card Reader, ATM Screen, Joe's Account, and Cash Dispenser objects.

- Actor Joe inserts his card into the card reader, which accepts the card.
- The card reader object reads the card number.
- The card reader object tells the ATM screen object to initialize the ATM screen.
- The card reader object tells the Joe's Account object to open his account.
- The ATM Screen object prompts Actor Joe for his PIN.
- Actor Joe gives his PIN to the ATM Screen object.
- The ATM Screen object tells the Joe's Account object to verify his PIN.
- The ATM Screen object prompts the Actor Joe for his transaction.
- Actor Joe tells the ATM Screen object that he wants to withdraw money
- The ATM Screen object prompts the Actor Joe for his transaction amount.
- Actor Joe tells the ATM Screen object that the transaction amount is RM20.
- The ATM Screen object tells the Joe's Account object to withdraw RM20 from his account.
- The Joe's Account object verifies that Joe has RM20 to withdraw.
- The Joe's Account object deducts RM20.
- The Joe's Account object tells the Cash Dispenser object to provide RM20 cash.
- The Joe's Account object tells the Cash Dispenser object to provide a receipt.
- The Joe's Account object tells the Card Reader object to eject Joe's card.

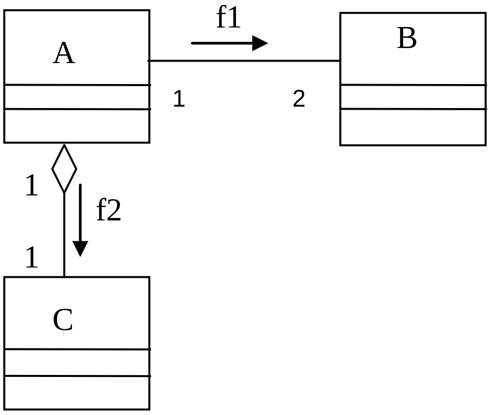
(6/25)

(b) Draw a state transition diagram for a bank account based on below statements:

- If the account is open and a withdrawal is made that brings the balance below zero, the account's state changes to overdrawn.
- If the account is overdrawn and a deposit is made that brings the balance to zero or above, the account automatically reopen.
- If the account is overdrawn and the balance is checked and found to be less than zero for more than 30 days, then the account will be closed.
- If the account is open and the customer requests its closure, then the account will be closed.

(6/25)

(c) Write a simple java program to implement the class diagram below:



(8/25)

(d) How to perform a unit test in an object oriented program? Give **one (1)** example.

(5/25)

- oooOooo -