Autumn Examinations 2011

Exam Code(s)	3IF1, 3BP1
Exam(s)	Third Year Information Technology Third Year Electronic and Computer Engineering
Module Code(s)	CT326
Module(s)	Programming III
Paper No.	1
External Examiner(s)	Prof. M. O’Boyle
Internal Examiner(s)	Prof. G. Lyons Dr. J. Duggan Dr. D. Chambers

Instructions: Answer any 4 questions.
All questions carry equal marks.

Duration	3 hrs
No. of Pages	5
Department(s)	Information Technology
Requirements	None

1: Create a class called **Complex** for performing arithmetic with complex numbers. Complex numbers have the form:

realPart + imaginaryPart * *i* where *i* is the square root of -1.

a: Use floating-point variables to represent the **private** data of the class. Provide a constructor method that enables an object of this class to be fully initialized. Also provide a no-argument conctructor with default values in case no initial values are provided. 5 MARKS

b: Provide a **public** method to add two **Complex** numbers: the real parts are added together and the imaginary parts are added together to create the result. This method should return a new **Complex** object initialized with the result e.g. if **c1** and **c2** are objects of type **Complex**, calling **c3 = c1.add(c2)** would add the value of **c2** to **c1** and then return a new object initialized with the result. The original values of **c1** and **c2** would not change. 6 MARKS

c: Provide a **public** method for subtraction of two **Complex** numbers: the real part of the right operand is subtracted from the real part of the left operand, the imaginary part of the right operand is subtracted from the imaginary part of the left operand. In the same way as for (b), this method should also return a new **Complex** method initialized with the result e.g. if **c1** and **c2** are objects of type **Complex**, calling **c3 = c1.subtract(c2)** would subtract the value of **c2** from **c1** and then return a new object initialized with the result. The original values of **c1** and **c2** would not change 6 MARKS

d: Provide a **public** method for printing **Complex** numbers in the form **(a+b*i*)** where **a** is the real part and **b** is the imaginary part. 4 MARKS

e: Write a short driver program to test your class. 4 MARKS

2.a: Suppose that you've written a program that displays two messages, as follows:

```
      public class NotI18N {
         static public void main(String[] args) {
            System.out.println("Hello.");
            System.out.println("How are you?");
     } }
```

You then decide that this program needs to display the same or similar messages for people living in France and Germany. Outline the steps needed to properely internationalise this program i.e. the hardcoded English language messages should be removed and replaced with a more flexible mechanism that will facilitate additional language support in the future. 10 MARKS

b: Write a simple Employee class that includes an id number, a name, and salary details and a suitable constructor method. Then write a Java program that uses an ArrayList to store a collection of Employee objects. Also, write the code for a Comparator class i.e. a class that implements the Comparator interface, that can be used to compare two Employee objects based on their id number. Finally, use the version of the Collections.sort() method that allows you to pass your own Comparator object to sort the list of Employee objects. 15 MARKS

3.a: Discuss briefly the differences between a process and a thread. Show (using simple code examples) how threads may be created (and started) using the following mechanisms:

(i) Application class implements the Runnable interface.
(ii) Application class extends the Thread class. 5 MARKS

b: What is meant by the term *deadlock*? Using the example of the *Dining Philosophers Problem* (covered in class), discuss how deadlock might occur in this case and propose a solution to overcome the problem. 10 MARKS

c: Outline the design and code implementation of the Java class for an object that will be used as a buffer to hold an integer value. The value may be updated randomly by one or more Producer threads, provided that it has already been consumed by one of a number of Consumer threads. Each value produced must be consumed at exactly once and there may be multiple producer and consumer threads executing (and attempting to access the buffer) concurrently.
10 MARKS

4. Develop a simple Java based payroll system that can calculate the weekly pay due for different categories of employees. The system should be implemented using the following design guidelines:

a: Implement an *abstract* base class called Employee that is used to hold and access basic information about an employee e.g. name, address, etc. This class should also define an *abstract* method called earnings() that returns the weekly pay for each type of employee. The class should include a suitable constructor and accessor methods to retrieve information about the employee.
5 MARKS

b: Implement a class called Manager, derived from Employee. A manager is paid a fixed weekly salary. The class should include a suitable constructor and should also implement the earnings() method. 5 MARKS

c: Implement a class called HourlyWorker, derived from Employee. An hourly worker is paid a fixed wager per hour, so in any given week they will be paind for the number of hours worked in the past week. The class should include a constructor and implement the earnings() method. 5 MARKS

d: Implement a class called CommissionWorker, derived from Employee. A commission worker is paid a base salary per week and an additional bonus based on the number of items sold during the past week. The class should include a constructor and earnings() method. 5 MARKS

e: Write a short driver program that creates an object for each of the employee sub-classes, it then calls the earnings() method for each object and displays the results. 5 MARKS

5.a: What types of Sockets are supported in the Java networking package and which type of Socket would you recommend for a VOIP type application and a File Transfer type application? Write a simple Java program that uses Datagram type sockets to exchange simple text messages i.e. one side sends a single String value to the other and it then receives another String value back as a response.
10 MARKS

b: Write a network Server program in Java where the Server waits for incoming client connections using stream type sockets. Once a Client connects it sends a single integer value to the server; the server then responds with another single integer value back to the Client. The connection is then terminated. The server should use a separate thread of execution for each new client connection and all interaction between the Server and the Client should be done within this thread. The answer should include full source code for the server application.
15 MARKS

6. Assume that a Sports Club wishes to store details about its members. Design and implement a Java application to support this requirement. The application should be able to print out and manage information about the members of the club.
The following guidelines should be used to construct the application:

a: A Java class, called Member, should be defined to store and manage member details. The class should include methods for updating member details and querying their subscription status i.e. are they fully paid up club members. Each member of the club should also have a unique membership id number, this number is automatically assigned when the member object is created.
10 MARKS

b: Define another Java class, called SportsClub, that will be used to manage club membership and access details about individual members. Member objects added to the SportsClub should be stored using a suitable collection object. SportsClub should include methods for adding new members, removing members, getting a list of current members and accessing information about an individual member (based on their id number). 10 MARKS

c: Write a short driver program, in a class called ClubManager, that creates an instance of SportsClub and uses its methods to add, lookup and remove club members. 5 MARKS