Ollscoil na hÉireann, Gaillimh *GX_____*
National University of Ireland, Galway

Spring Examinations, 2007/2008

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. J. Keane
Internal Examiner(s)	Dr. D. Chambers Prof. G. Lyons

Instructions: Answer any 4 questions.
All questions will be marked equally.

Duration	3hrs
No. of Answer Books	1
No. of Pages	5
Department(s)	Information Technology

1. 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

2.a: Describe the general structure of the IO Streams classes provided in the Java programming environment. 5 MARKS

b: Create a standalone Java application that will count and sum up the number of lines in the text file passed as an argument on the command line. The program should create a **FileReader** object and pass this in the constructor of a **LineNumberReader** object to handle the file reading required.
The **LineNumberReader** class has two useful methods (that could be used):

public String readLine() throws IOException; This method reads a line of text. It returns a String containing the contents of the line, not including any line-termination characters, or null if the end of the stream has been reached.

public int getLineNumber(); This method returns the current line number.
10 MARKS

c: Write a Java program that uses an ArrayList to store a collection of Integer 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 Integer objects. Then use the version of the Collections.sort() method that allows you to pass your own Comparator object to sort the list of Integer objects. 10 MARKS

3.a: Discuss briefly the differences between a process and a thread. How should executing threads be stopped (assuming they still haven't finished their work)?
5 MARKS

b: Show (using simple code examples) how threads may be created (and started) using the following mechanisms:
(i) Application class extends the Thread class.
(ii) Application class implements the Runnable interface.
Assume you have a bank account class that may be accessed by more than one thread of execution simultaneously. Show how the various methods of the class may be made thread safe. 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.a: Desribe the functionality provided by the following Java code. Would it have been possible to write similar code using an *Enumeration* instead of an *Iterator*? Explain your answer.

```
import java.util.*;

   static void filter(Collection c) {
      for (Iterator i = c.iterator(); i.hasNext(); )
            if (!cond(i.next()))
               i.remove();
      }
```

10 MARKS

b: Develop a simple GUI-based Java program that may be used to control a washing machine. Use suitable Swing components to allow the washing machine operator to perform the following functions:

1) Switch the machine on.
2) Choose a temperature from a list.
3) Spin speed selection buttons - can be 600, 800 or 122 RPM.
4) Display the current status of the wash cycle.

Show the top-level design of the GUI, including any Panels and related Layout Manager objects that you propose to use. For each of the components you've chosen above, write the code to construct the component, add the component to a container and then setup simple event handling for the component (for those that generate events). The event handlers need only print out a message indicating that they have been called. 15 MARKS

5: 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. 7 MARKS

(Question 5 is continued on the next page)

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 7 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. 3 MARKS

e: Write a short driver program to test your class. 3 MARKS

6.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? 5 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 text string to the server with a simple query – the server then responds with a text based response. 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.
10 MARKS

c: Suppose that you've written a program that displays three messages, as follows:

```
public class NotI18N {

    static public void main(String[] args) {

        System.out.println("Hello.");
        System.out.println("How are you?");
        System.out.println("Goodbye.");
    }
}
```

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