



## **Spring Examinations 2011 / 2012**

**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. M. Madden  
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. The following Java code provides the outline of a simple bank account class:

```
import java.io.*;
public class Account implements Comparable<Account>,
Serializable {
    protected int accnum;
    protected HolderDetails holder;
    protected List<Transaction> transactions;
    protected float balance;

    // Add a suitable constructor here

    // Add methods to make deposits / withdrawals

    // Method to print out account transaction summary

    // Add suitable attribute accessor methods

    // Add method to implement the Comparable interface
}
```

- a: Complete the implementation of the Account class, providing a suitable constructor, attribute accessor methods, methods for making a deposit or withdrawal, a method to print out a transaction summary related to a range of Dates and an implementation method for the Comparable interface.  
7 MARKS
- b: Provide implementations for the HolderDetails class and the Transaction class. The HolderDetails class is used to store personal details about the account holder. The Transaction class contains details about past transactions including the type of transaction, the amount and the Date.  
7 MARKS
- c: Define and implement a new class, called CurrentAccount, derived from Account, that allows withdrawals to proceed up to some overdraft limit. Note that the base Account class shown has no overdraft facility.  
5 MARKS
- d: The attributes of class Account are defined as *protected*. What is the implication of this definition?  
3 MARKS
- e: What does the statement *implements Serializable* mean? What are the implications of this statement?  
3 MARKS

- 2.a: Describe the general structure and purpose of the IO Streams classes provided in the Java programming environment. Also briefly describe the mechanism to support random file access in Java? 4 MARKS
- b: Write a Java application that inputs a date as a string in the form 17-07-2010  
The program should use an object of class *StringTokenizer* to extract the various components of the date string as tokens. The program should then convert the day, month and year to int values and display them. 6 MARKS
- c: Write a simple Student class that includes an id number, a name, and course details and a suitable constructor method. Then write a Java program that uses an ArrayList to store a collection of Student 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 Student 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 Student objects. 15 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 exactly once and there may be multiple Producer and Consumer threads executing (and attempting to access the buffer) concurrently. 10 MARKS

4.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 Spain. Outline the steps needed to properly 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 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 set up 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.a: What types of Sockets are supported in the Java networking package? Which type of Socket would you recommend for a VOIP type application and a File Transfer type application? **5 MARKS**

- b: Write a Java application that uses Stream type sockets to exchange Java Objects using object serialisation. The client side should connect to the server and send it an Integer Object. The server should print out this value and respond to the client with a text based response encapsulated in a String Object. The client should receive the String Object from the server and print out this response. **10 MARKS**

- c: Write another Java application with the same functionality as outlined above, in part b of this question, but this time using Datagram type sockets. Hint: you can use `ByteArrayOutputStream` and `ByteArrayInputStream` to populate and read the array associated with the `DatagramPacket` object. **10 MARKS**

6. Assume that a Sports Club at the University 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 student details. The class should include methods for updating member details and querying their registration status i.e. are they fully paid up members of the club. 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 name or 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