NUI Galway
OÉ Gaillimh

Spring Examinations 2014 / 2015

Exam Code(s)	3BCT, 3BP1
Exam(s)	Third Year Computer Science & Information Technology Third Year Electronic and Computer Engineering
Module Code(s)	CT326
Module(s)	Programming III
Paper No.	1
External Examiner(s)	Dr. J. Power
Internal Examiner(s)	Prof. G. Lyons Dr. M. Madden *Dr. D. Chambers
Instructions:	Answer any 4 questions. All questions carry equal marks.
Duration	2 hrs
No. of Pages	4
Department(s)	Information Technology
Requirements	None

1.a 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 course code. The Student class implements an accessor method called getCourseCode() that returns a String representing the course code for that student e.g. 3BCT. 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. 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 1200 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 have 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

2.a: 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 string object 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. 12 MARKS

b: Write another Java application with the same functionality as outlined in part a 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. This application does not need to implement a reliable data transfer protocol. 13 MARKS

3.a: What is the best way to stop executing threads (assuming they still have not finished their work)? Show using a code example how a thread may be created (and started) using an application class that 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

b: 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 written 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. 15 MARKS

4.a: Write a Java application that inputs a date as a string in the form 17:02:2009 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

b: Suppose that you have written a program that displays two messages, as follows:

```
public class q4b {
   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 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. 7 MARKS

c: Using suitable source code samples, outline the steps required to use a Database within a Java program using the JDBC API. How are transactions supported using this mechamisn? What type of Metadata may be retrieved from a Database using JDBC? 12 MARKS

5: 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 member 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. The Member class should implement the Comparable interface and use the membership id number to define the natural order for these objects.
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. Member objects should be sorted by their id number as they are added to the collection. 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