

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

**Semester II Examinations, 2004/2005**

Exam Code(s)	<u>1IT1</u>
Exam(s)	<u>Master's Degree in Information Technology</u>
Module Code(s)	<u>CT509</u>
Module(s)	<u>Programming</u>
Paper No.	<u>1</u>
Repeat Paper	<u>N</u> Special Paper <u>N</u>
External Examiner(s)	<u>Prof. D. Bell</u>
Internal Examiner(s)	<u>Prof. G. Lyons</u>
	<u>Dr M. Madden</u>

**Instructions:**

Answer any four questions.  
 All questions carry equal marks.

Duration	<u>3 hrs</u>
No. of Answer books	<u>1</u>

**Requirements:**

Handout	<u>✓</u>
MCQ	<u></u>
Statistical Tables	<u></u>
Graph Paper	<u></u>
Log Graph Paper	<u></u>
Other Material	<u></u>

No. of Pages	<u>3</u>
Department(s)	<u>Information Technology</u>

You are part of a team working for Athenry Aircon Automation to develop software for the new 2005 version of their air conditioning system, the **AAircon05**. The system includes a thermometer, heater, cooler and control software. You are required to provide Java implementations of the following software components:

- (a) A Java **interface** called **Thermometer** that specifies three methods: (1) get the air temperature, returning a floating-point number; (2) set the units to Celsius; (3) set the units to Fahrenheit (4)
- (b) An **interface** called **Heater** specifying methods to turn on and off the heater, and an **interface** called **Cooler** specifying methods to turn on and off the cooler (5)
- (d) A class called **AAircon05**, composed with private members of references to a **Thermometer**, a **Cooler** and a **Heater**. It has a constructor that initiates the **Thermometer**, **Heater** and **Cooler** references with objects supplied by the constructor's caller, and a public method **control** (see below). It also has a private member **isActive** for storing whether the controller is active or not, and associated public methods to activate and deactivate the controller. Finally, it has a private member for storing the target temperature (a floating-point number) and associated methods for setting and getting the target temperature. (10)
- (e) The **control** method, which runs in an infinite loop, operating as follows:
  - If the controller is not activated, it does nothing
  - Otherwise, it gets the current temperature and compares it to the target temperature. If they are within 1 degree of each other, it turns both the heater and cooler off. If the current temperature is lower, it turns the heater on and the cooler off, and does the opposite if the current temperature is higher. (6)

2. Discuss in detail all of the following four Java topics, using code examples for illustration where appropriate:

- (1) File management with the **File** class
- (2) Data conversion between strings, integers and floating point-numbers
- (3) Keywords **break** and **continue**
- (4) Working with multi-dimensional arrays (25)

3. (a) What advice would you give a programmer beginning object-oriented design about how to identify objects? (5)

(b) Outline the principal differences between Object Oriented Programming and Procedural Programming. Discuss what you consider to be the principal advantages of Object Oriented Programming over Procedural Programming. (7)

(b) Galway-Mayo Institute for the Bewildered (GMIB) requires your assistance with developing a student record system. Below is a description they have provided:

*For all students, we will store the name, ID, programme name and year. However, nobody will be 'just' a student – they are categorised as Undergrads, Taught Postgrads and Research Postgrads. For undergrads, we will store a grade point average (GPA), which is: a number in the range 0.0 – 5.0, as well as the details listed above. For all postgrads, we will store a project title and a project GPA, as well as the generic student details listed above. Research postgrads will additionally have a monthly stipend, whereas taught postgrads will have a coursework GPA.*

Based on this description, draw a diagram showing an appropriate class hierarchy for the student data. Using the diagram, develop a **partial** implementation of classes required. You must include all member variables identified in the description, and constructors to initialise them to arbitrary values (e.g Name: "NoName", ID: 9999999, etc.), but you do not need to include any other methods. (13)

4. Write a Java program to read and process a text file. It must conform to the following requirements:
- (a) The name of the text file is entered as a command line argument to the program. If no name appears on the command line, the program displays an error message and exits. Likewise, if the file does not exist, the program displays a message and exits. (5)
  - (b) The file is opened and each word in it is read and stored in an ArrayList. The file is closed after use. (Words are considered to be any text that is separated by spaces, tabs or newlines.) (5)
  - (c) The minimum and maximum word lengths are calculated in a separate method and the answers returned to the main method, which displays a message with the total number of words, minimum length and maximum length. (5)
  - (d) In another method, the number of occurrences of the words "the" and "a" (ignoring their case) are counted and the answer returned to the main method, where it is displayed. (5)
  - (e) The program must include appropriate code to handle any exceptions that may arise. (5)
5. (a) Discuss the structure of code for a Java applet, including the purpose of each of the main applet methods. Include an explanation of what happens when a HTML page is loaded containing a link to an applet. (6)
- (b) What is event-driven programming? Demonstrate with an example how event classes and event listeners are used in Java. (7)
- (c) Write the code for a Java applet that draws five concentric circles, as follows:
- The circles are centred in the applet region
  - The largest circle fits exactly within the applet region (i.e. its diameter is either the width or the height, whichever is smaller)
  - The radius of each successive circle is smaller than the last one by an amount specified by a parameter GAP that is contained in the HTML file
- Also provide a minimal HTML file to display your applet within a region of size 500 x 400, specifying a value of 25 for the GAP parameter. (12)
6. (a) Java supports lightweight object persistence. Explain what this is. Using code examples, explain how object serialisation is performed in Java. (9)
- (b) For the following line of Java code, indicate in what order all parts are evaluated, and why:
- ```
result = -a / Math.abs(b/c*d) + (++e);
```
- (7)
- (c) The three key features of the Java Collections Framework are its interfaces, implementations and algorithms. Discuss this, providing examples of each of these features. (9)