



DUBLIN CITY UNIVERSITY

Semester 1 Examination 2000/2001

COURSE: *M.Eng./Grad.Dip in Electronic Systems
RAE - RACeE*

YEAR: Postgraduate

SUBJECT / MODULE: EE553 – Object Oriented Programming

EXAMINERS: Dr. Derek Molloy (DCU Extension 5355)

TIME ALLOWED: 3 hours

INSTRUCTIONS: Answer **FOUR** questions.
All questions carry equal marks.

- **Before you start – put your name and id-number on the supplied disks!**
- Please use the answer books and the supplied disks to complete your answers to this exam.
- On the disk, please use separate directories for each question attempted, called question1, question2, etc.
- For each question you attempt, please reference your files on the disk related to that question in your answer book.
- **You are responsible for insuring that you have copied all the files that form your answers onto the disk.**

PLEASE DO NOT TURN OVER THIS PAGE UNTIL YOU ARE TOLD TO DO SO

This booklet contains 5 pages, including the cover sheet.

Question 1.

(a) Answer the following short questions. Keep your answers concise.

- (i) Compare the concept of *friendly* in Java to the concept of *a friend* in C++.
- (ii) Explain the use of *casting*. Give an example of when you would use a cast.
- (iii) Explain the concept of the Object class in Java.
- (iv) What is a *static method* and why would it be used?
- (v) What is meant by the term *encapsulation*?
- (vi) Explain the difference between *lightweight* and *heavyweight* components.
- (vii) Consider the code: `int x[] = new int[25];` Explain why each of the following statements are either true or false:
 - a. `x[24]` is 0
 - b. `x[24]` is undefined
 - c. `x[25]` is 0
 - d. `x[0]` is null
 - e. `x.length` is 25

[14 marks]

(b) What are **Exceptions** in Java? Why and when are they used? Give an application example of how you might use your own exceptions?

[6 marks]

(c) The Java language uses **automated garbage collection**. Why is this a useful facility? Outline how it works and give three ways that it may be invoked.

[5 marks]

Question 2.

- (a) Compare and contrast **applets** to **applications**. When would one choose to use applets over applications? What are the limitations of applets? Describe the lifecycle of applets and applications. Can an application be an applet? [9 marks]
- (b) Write a Java applet that loads an image from the local hard disk and displays it wherever the mouse is pressed. If the mouse is dragged then the image should be displayed at the location of the mouse. You should use the Canvas (java.awt.Canvas) class for your display. *Note: An image is included on the drive for your convenience.* See Figure 1 for the example output. [16 marks]

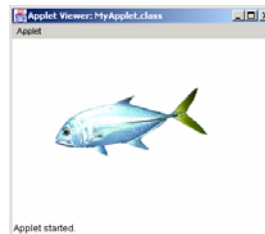


Figure 1. The example output of the Mouse moving application (See Question 2 part b)

Question 3.

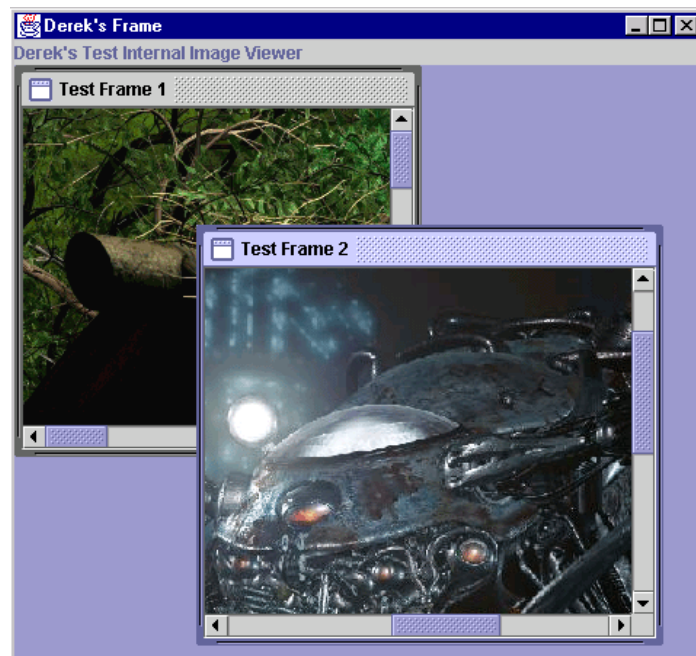
- (a) **Constructors** and **destructors** used in C++. Why are they used and what is the C++ syntax for their use? [5 marks]
- (b) **Polymorphism** is a feature of most OOP languages. Describe polymorphism with reference to over-loading and over-riding. Address the following issues: [11 marks]
- a. Over-loading versus over-riding.
 - b. Over-loading issues that arise when a child over-rides one method of many that are inherited from a parent class.
 - c. How do access modifiers affect over-riding?
- (c) Write a section of C++ code that demonstrates **polymorphism**, including an example of over-loading, over-riding and scope resolution. [9 marks]

Question 4.

- (a) **Remote Method Invocation (RMI)** is a very powerful aspect of the Java programming language. Explain how communication takes place. What are stubs, skeletons and the registry? What are the advantages and disadvantages of RMI over regular client-server communication?

[10 marks]

- (b) Write a section of Java code that uses the Java **Swing** set to create the following **application**. The application should have internal image frames, with a dialog box to open new internal frames.



[15 marks]

Question 5.

- (a) What is JDBC and how is it used in Java? Explain the terms Driver, Driver Manager, Statement (all 3 types), Result Set and Connection.

[10 marks]

- (b) Write a Java client/server application, where the client sends a string to server, the server then either (depending on the command sent):

- **Encrypts** the string and returns the encrypted string back.
- Or **decrypts** the string, sending the decrypted version back.

You should write the code for both commands (encrypt and decrypt).

Note: The encryption algorithm should be very simple, even as easy as adding one to each character, or subtracting the letter from 'z' etc. Consider only letters between 'a' and 'z', all lowercase.

You have been supplied with three sets of code to handle the basic aspects of this application. These are called

1. **Client.java**,
2. **Server.java** and
3. **ConnectionHandler.java**

These files are in the directory **question5**.

[15 marks]