SEMESTER ONE EXAMINATIONS 2003

MODULE: ***Object Oriented Programming – EE553***

COURSE: ***M.Eng./Grad. Dip. in Electronic Systems***
M.Eng./Grad. Dip. in Telecommunications Engineering
RAEC – Remote Access to Continuing Eng. Education

YEAR: ***Postgraduate (Year 5)***

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.**

This booklet contains 5 pages, including the cover sheet.

DO NOT TURN OVER THIS PAGE UNTIL YOU ARE INSTRUCTED TO DO SO

Question 1.

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

(i) How is *scope resolution* performed in C++?

(ii) Why are *destructors* required in C++?

(iii) Explain how Java avoids the difficulties associated with *multiple inheritance*?

(iv) Explain the use of the *Object class* in Java.

(v) In C++ what is a *static local variable* and why would it be used?

(vi) What does the term *overloading* mean?

(vii) In the following piece of code explain what occurs. What is the value of x after execution (i.e. after the last line)? Why?

```
int x=6, *p;
p = &x;
*p+=++x;
```

[14 marks]

(b) What is an **abstract class**? Why would you create an abstract class?

[4 marks]

(c) Discuss **constructors** in C++. Can they be overloaded? Why can they not be virtual? In what order are they called when inheritance takes place? What is the copy constructor and how can it provide specific functionality?

[7 marks]

Question 2.

(a) Explain the use of **friend functions** in C++. Why are they a useful feature? What difficulties could arise with the use of friend functions?

[8 marks]

(b) You will find a section of code on the disk. It contains the outline definition for the **Person**, **Student**, **Staff**, **Lecturer** and **Postgraduate** classes.

- Add constructors to these classes
- Add display() methods to these classes
- Create an array of **Person** objects and add constructed **Lecturer** and **Postgraduate** objects to the array (hint: use pointers and dynamic typing).
- Loop through the array and call the display() methods of the objects – where the correct display() method for (the dynamic type) should be called.

[17 marks]

Question 3.

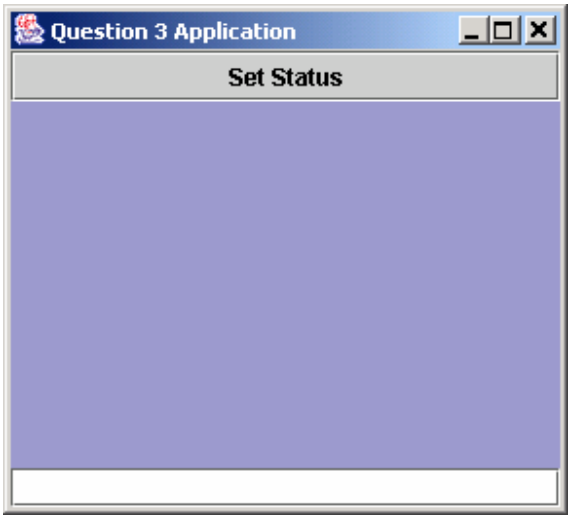
(a) What is **Remote Method Invocation** (RMI) and how is it used in Java? Explain the terms *skeletons* and *stubs*. What are the limitations of RMI?

[9 marks]

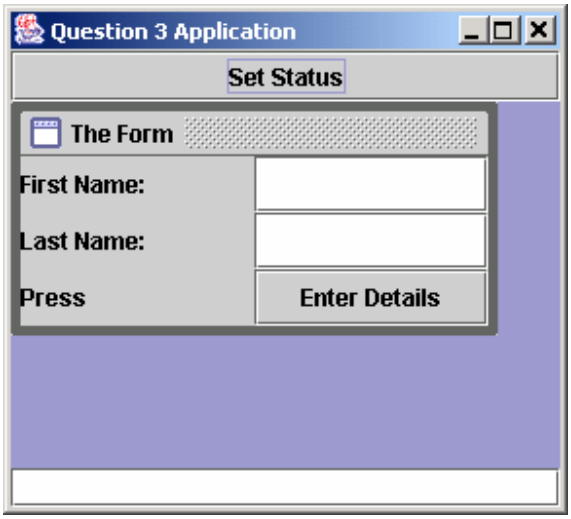
(b) Write a Java application that looks like the application below. The application:

- Should open up with the format as shown in (a)
- When the "Set Status" button is pressed the internal frame should appear as in (b)
- When the details are entered in the fields and "Enter Details" is pressed as in (c)
- The Status at the bottom should display the data entered in the fields as in (d)
- If the "Set Status" button is pressed again then it should begin again.

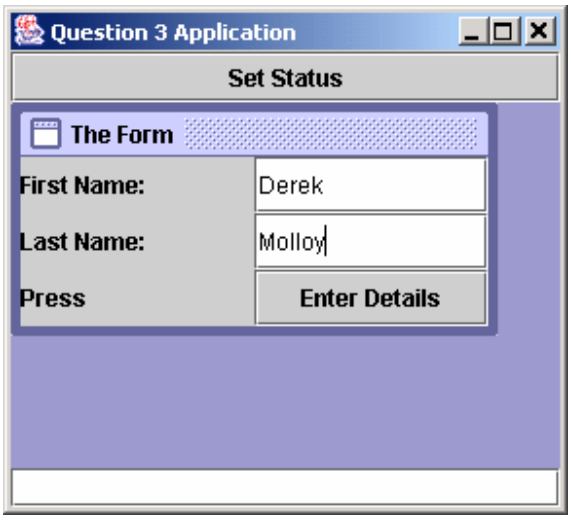
[16 marks]



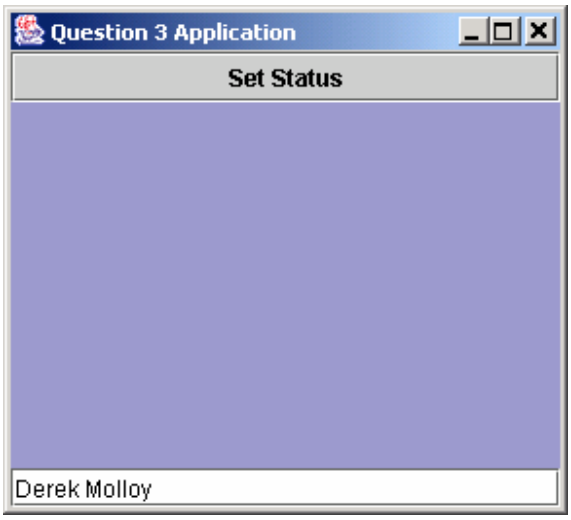
(a)



(b)



(c)



(d)

Question 4.

(a) The java.util.Math contains many mathematical operations. In your opinion, why is there no public constructor for the Math class? Show an example of how you would use the random() method to pick a random whole number between 1 and 100 (inclusive).

[5 marks]

(b) Write a Java application that uses a Java Vector and Stack as follows:

- Create an Account class that stores the account number, account balance, account owner and has a constructor, display(), makeLodgement() and makeWithdrawal() methods.
- In a command line Java application create a Vector object
- Store 3 anonymous Account objects in this vector
- Create a loop, looping through the Vector object and display the details of the Account objects stored in this vector.
- In the same loop "push" these elements onto a Stack object.
- From the java.util.Stack API documentation work out how to extract the elements from the stack and display the Account details.

[15 marks]

(c) In the example in (b) you created an Account object. How would you compare two Account objects to see if they had the same values? Show this with a short segment of code – for example – how could you test that these two objects are equivalent?

```
Account a = new Account("Derek",100,1234);
Account b = new Account("Derek",100,1234);
```

[5 marks]

Question 5.

(a) Explain using an example why you would need to synchronize a segment of code when using Java threads? (Your answer should show a line-by-line step through of a segment of code, explaining why it would not work correctly if the segment of code was not synchronized). If synchronization is a solution to making an application thread safe, then why should we not just synchronize all our code?

[7 marks]

(b) Write a Java client/server application pair, where the client passes a DepositAccount object to the server and the server calculates the interest on the account using the current interest rate available on the server. The account object is then passed back to the client, where the client displays the details including the updated balance. The server should also display the amount of money added through the interest calculation.

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

- **Client.java**,
- **Server.java** and
- **ConnectionHandler.java**

These files are in the directory **question5**. An example client/server output is shown below:

[18 marks]