



<b>DUBLIN CITY UNIVERSITY</b>
-------------------------------

**SAMPLE EXAMINATIONS 2010/2011**

<b>MODULE:</b>	EE402 Object-oriented Programming for Engineers
<b>COURSE:</b>	MTC – M.Eng. in Telecommunications Eng. MEN – M.Eng. in Electronic Systems IPME - Individual Postgrad. Modules-Electronics MEQ - Masters Engineering Qualifier Course DME – Digital Media Engineering ICE – Information Communications Engineering IFPTE1 – PG Int. Foundation Prog. Telecom Eng.
<b>YEAR:</b>	4/C
<b>EXAMINERS:</b>	Prof. P. Ashburn (External Examiner) Dr. David Linton (External Examiner) Dr. Derek Molloy (DCU - Phone: 5355)
<b>TIME ALLOWED:</b>	3 Hours
<b>INSTRUCTIONS:</b>	Answer question 1 <u>and</u> any three of the remaining four questions All questions carry equal marks

---

**Please do not turn over this page until you are instructed to do so**

- The use of programmable or text storing calculators is expressly forbidden.
- Please note that where a candidate answers more than the required number of questions, the examiner will mark all questions attempted and then select the highest scoring ones

This is a computer based examination:

- Please use the answer book and the supplied USB key to complete your answers. For each question you attempt partly or completely electronically, please refer to it in the paper answer book.
- On the USB key & network, please use your exam number as the root directory and use separate subdirectories for each question attempted e.g. Q1 etc.
- You are responsible for ensuring that you have copied all the files that form your answers onto the USB key and network drive. Please double check.
- Any additional files required for the exam are on the network drive. The location will be announced at the beginning of the examination.

## QUESTION 1

[TOTAL MARKS: 25]

1 Answer the following questions (keep your answers concise). Each short question carries approximately 2 marks.

- a) In Java, why is the `super()` call always the first line of the child's constructor? How does this compare to C++?
- b) Why are all methods in the Java `Math` class static?
- c) Why do pointers require a dereference type in C++?
- d) What is the main difference between a C++ *class* and a C++ *struct*?
- e) What is the main difference between abstract classes in C++ and abstract classes in Java?
- f) Describe the use of the *break* and *continue* keywords in C++.
- g) Why can abstract classes not have static abstract methods?
- h) Briefly explain the difference between *modal* and *modeless* dialog boxes in Java.
- i) Describe the use of the `final` keyword on a method in Java.
- j) Why do we use assertions in Java?
- k) What are the advantages of using generics in Java?
- l) Explain the `try`, `catch` and `finally` keywords as they relate to Java exceptions.
- m) What is a C++ functor?

[End of Question 1]

---

**QUESTION 2****[TOTAL MARKS: 25]****2(a)****[4 Marks]**

Describe the use of the *void* pointer in C++ through the use of a source code example. How can we modify a value that is pointed to by such a pointer?

**2(b)****[8 Marks]**

Write method implementations for the Person class definition below:

```
#include<iostream>
#include<string>
using namespace std;

class Person
{
    private:
        string name;
        friend void clearName(Person &);
    public:
        Person(string);
        Person(const Person &);
        virtual bool operator == (Person);
        virtual void display();
};
```

**2(c)****[8 Marks]**

Write a child class “Student” for the class described in 2(b) that should differ slightly from the Person class. The child class should also contain an overriding display() method. Write a main() function that tests all of the methods described in the Person class and your Student class.

**2(d)****[5 Marks]**

What is dynamic binding? Add some code to the main() function in 2(c) to demonstrate the concept of dynamic binding using the Person and Student classes.

[End of Question 2]

---

**QUESTION 3****[TOTAL MARKS: 25]****3(a)****[7 Marks]**

Describe the following C++ STL categories: Containers, Iterators and Algorithms. Concisely describe four examples for each category.

**3(b)****[7 Marks]**

Write some C++ code that stores values of any standard type (e.g. int, float, double etc.) in a STL vector, uses a STL algorithm to sort the data and then displays the values using a STL `for_each` algorithm. Note: the output function should work for any standard type.

**3(c)****[11 Marks]**

Write a section of Java code that demonstrates the following features of the Java Vector class (`java.util.Vector`) in the following order:

- Add objects of any class to the Vector.
- Demonstrate the use of the *capacity()* method and show how the capacity changes in a vector when the current capacity is exceeded.
- Use the *contains()* method.
- Remove the last element from the vector.
- Use the *setElementAt()* method.
- Use the *lastIndexOf()* method.
- Display the remaining contents of the Vector.

Attach the output of your code to your answer.

[End of Question 3]

---

**QUESTION 4****[TOTAL MARKS: 25]****4(a)****[5 Marks]**

Discuss the differences and relationships between: a Window; a Frame; a Dialog Box; and a Panel in Java. Java Graphical User Interface (GUI) applications are event driven – what does this mean?

**4(b)****[10 Marks]**

Write the Java code to create the Java Swing Application that is displayed in Figure 4.1. The application should allow a user to enter text in a text field that will be displayed in the text area when the send button is pressed. If the clear button is pressed the text that was being entered in the text field is cleared. This must be written as a Swing application and the application should terminate when the X button is pressed.

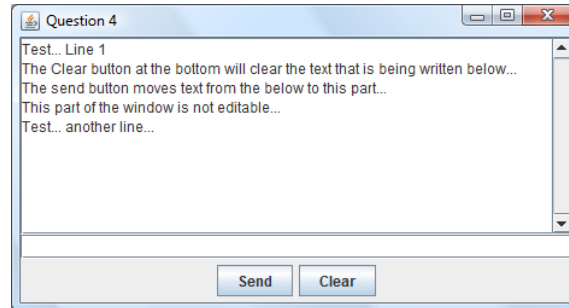


Figure 4.1

4(c)

[10 Marks]

Modify the Java Swing application written in part (a) to allow any number of windows to be opened at the same time. When a message is typed in any window's text field and send is pressed, the message will be displayed in the text area of all windows. The number of windows could be passed from the command line interface. See Figure 4.2 below. Note: This is not a client/server application as the windows will all be on the same JVM.

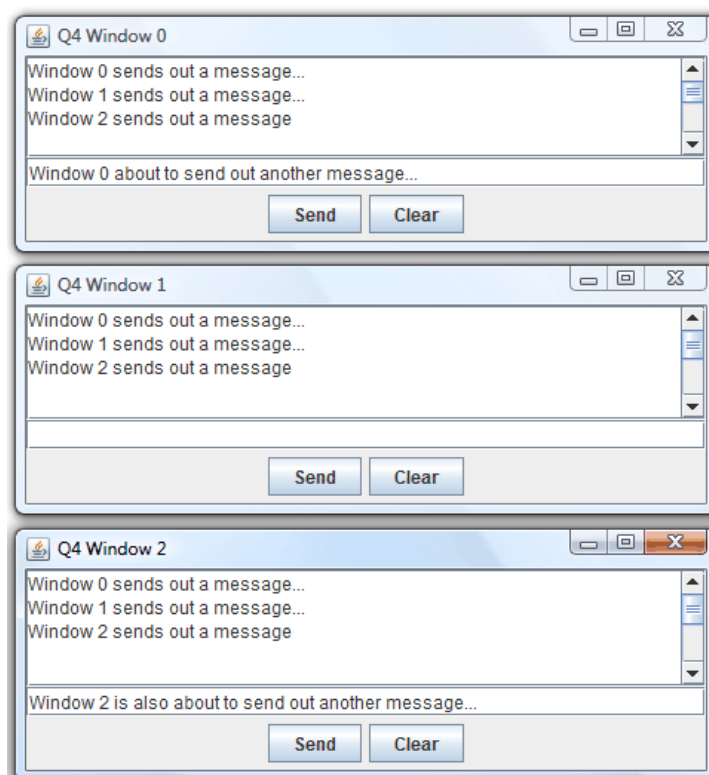


Figure 4.2

[End of Question 4]

**QUESTION 5****[TOTAL MARKS: 25]****5(a)****[25 Marks]**

Write a client/server application that allows a GUI client to send a message to the server that is converted to lower case and sent back to the client. There should be a suitable graphical user interface. See Figure 5.1 for an example client interface and Figure 5.2 for an example running server. The server should be made fully threaded.

**Note:** You have been provided with 3 files to help you with your solution: Client.java, Server.java and ConnectionHandler.java on the examination network drive. Please build your solution using these files.

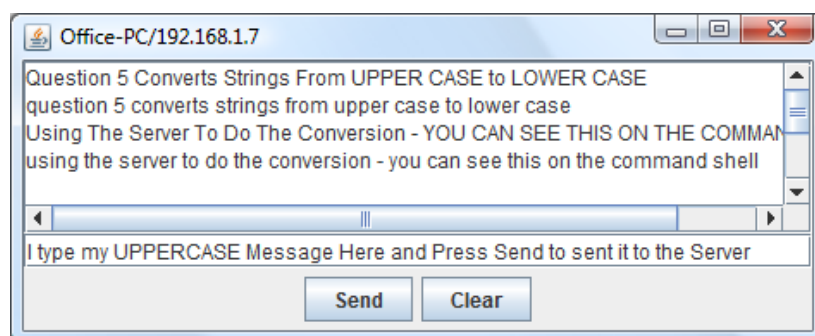


Figure 5.1 Client GUI, where the text is entered and server results are displayed.

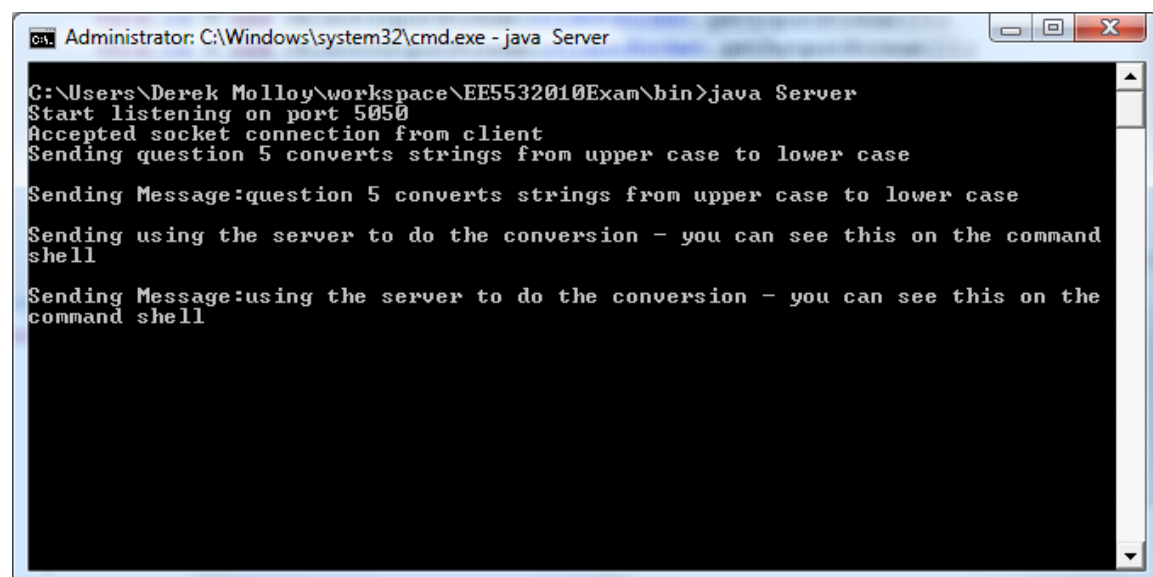


Figure 5.2 Server, where the text is processed

[End of Question 5]

[END OF EXAM]