UNIVERSITY *of* LIMERICK
OLLSCOIL LUIMNIGH

COLLEGE *of* INFORMATICS *and* ELECTRONICS

Department of Computer Science
and Information Systems

End-of-Semester Assessment Paper

Semester: **Semester 1**
Module Code: **CS4135**
Duration of Exam: **2.5 hours**.
Lecturer: **C. Ryan**

Academic Year: **2006/07**
Module Title: **Software Architecture**
% of Total Marks: **70%**

Instructions to the Candidates:
- Answer any 5 questions.
- All questions carry equal marks.

Q1:- Attempt ALL parts.

a) What is meant by Garbage Collection in Java? How does it make programming easier? (3)

B): What are the three top-level elements to appear in a java source file? (3)

c): What happens when the following piece of code is executed in (i) C++ (ii) Java (2+2)

(if you find any error, suggest the way to correct it).

```
int a;

double b = 20.2;

a = b;
```

d):- (i) Define a two dimensional array in Java of the type float. Initialize it to have dimensions of 20 by 20. (Do not specify its contents). (2)

(ii): Are arrays treated like primitive data types in Java? What can we typecast an array into?(2)

Q2:Attempt ALL parts.

(a) Describe the difference between a class variable and an instance variable. Give an example when it is useful to employ a class variable. Separately answer what is the keyword associated with the class variables? (1+1+1)

b) Have a look at the following code.

```
class Book{
   public String getAuthor(){
         return author;
   }
}
class javaBook extends Book{
    String getAuthor(){
        System.out.print("getting it");
        return super.getAuthor();
   }
}
```

(i) Does it correctly inherit the method? Explain. (1+1)

(ii) Justify your answer to part (i) by explaining what restriction does java impose on the access modifiers when inheriting a method. (2)

(c) (i)Would the following piece of code generate "compile time" errors? Explain your answer with reasoning. (1+1)

```
class myclass{
  void method1(){
    method2();
  }
  void method2 () throws RuntimeException{
```

```
    throw new RuntimeException();
  }
}
```

(ii) If there are any compile time errors, correct them or else explain what would happen if the same code is executed. (1)

(d): A superstore advertises itself proudly that they sell everything ranging from the aeroplane to any book of your choice. They need to create an inventory management system. They hire you to write the system for them. Their technical team demands that the wide variety of items may not be related in anyway except that each one of them must be able to return a numeric ID (int).

(i) What two ways does java provide to impose such a constraint? Which will be preferable for better design considering this scenario? Explain. (1+0.5+0.5)

(ii)Consider two classes Book and Car. Write the complete code to fulfil the demands of the whole scenario. (No need to write detailed characteristics of books and cars. Just focus on the code that defines and imposes the constraints). (1+0.5+0.5)

```
class book{
 String title;
}

class car{
 String model;
}
```

Q3: Attempt ALL parts.

a): (i) Let us consider that we have a class AA. Class AA has some private members and methods. We need to write another class that 1) can be instantiated as a stand-alone class and 2) still be able to access the private members of class AA. What kind of class do we need? (1)

(ii) Write a public class AA with two private member variables of the type int. Then write another public class BB that fulfils the criteria described in part (i). In its main method print the values of the private member variables of class AA. (1+1+2)

(iii) Write the names of 4 kinds of nested classes. (2)

b): (i) What is the difference between a container and a component in Swing (from a functional perspective)? Give one example each of a container and a component. (2)

(ii) List the three top-level Containers in Swing and an intermediate container that has flow layout as its default. (2)

(iii) What is meant by a containment hierarchy in swing? What sits at its top/root? (2)

(iv) Which layout manager allows the components to use exactly the same display space at various times? (1)

Q4:Attempt ALL parts.

(a) Write a class Frame1 with three member variables of the types JLabel, JButton and JTextField. Write an instance method setUp(). Write the code of this instance method so that you create a frame with a GridLayout that allows for a single column and 3 rows. The label should have an intial value "initial" when the frame shows up. When you enter some text in the textfield and click the button, the label is updated with the text entered in the text field. Write the event handling code with the help of an anonymous inner class.

You can use the method getText() to get the text entered in the field and use the method setText to set the label of JLabel control.

Ensure that the frame is visible and that clicking its 'x' button (available at the right top of any window) exits the application. Also ensure that the frame fits the size of its components.

Note: that all this functionality (apart from declaring the label, button and textfield) has to be inside the setup Method. You do NOT need to write the call to the event dispatching thread. (9)

(b) (i) Let's say you are writing an application for a Middle Eastern country. The labels have to be in their native language. What property of the swing containers comes to your mind as the most relevant so the components appear properly? (1)

(ii) What is the base class for all the events? In order to get the information about the source of the event, which method is the most commonly available method. (1)

(iii) What is the difference between a low level event and a semantic event? Give an example of a semantic event. (2)

(iv) How to avoid implementing all the unnecessary methods of a listener interface when you are only interested in one method? (1)

Q5:- Attempt ALL parts.

a) What is a software component? Give two well known definitions and state where the
definition comes from. (6)

b) Describe in detail the advantages of using software components? (8)

Q6:- Attempt ALL parts.

a) Describe Kruchten's 4+1 View Model of Software Architecture, detailing what each of the views are and who uses them. (12)

b) Why is the ``+1'' view considered different to the other four? (2)