OLLSCOIL NA hÉIREANN
THE NATIONAL UNIVERSITY OF IRELAND, CORK

COLÁISTE NA hOLLSCOILE, CORCAIGH
UNIVERSITY COLLEGE, CORK

Autumn Exam 2011

Second Year Computer Science

CS2500: Software Development

Dr Carron Shankland
Professor J.A. Bowen
Dr M.R.C. van Dongen

INSTRUCTIONS: Answer all 9 questions for full marks (225). All questions carry equal marks (25). There is no need to write whole classes *except where this is explicitly stated*. There is no need to write `import` statements. Use meaningful identifier names. Pay attention to the layout of your code and make sure your coding style adheres to the Java coding conventions. There is no need to add comments.

3 hours

Question 1: Basics. *(25/225 marks)*

Question 1.a. *(10 marks)*
Provide a `main( )` method that uses the enhanced `for` loop to print its arguments to standard output.

Question 1.b. *(15 marks)*
Provide a method that creates a `Scanner` to read from standard input. The method should read words from standard input and print them. The method should keep on reading until either (1) there is no more input, or (2) the user inputs the word `stop`. If the user inputs the word `stop` then that word should also be printed.

Question 2: Classes. *(25/225 marks)*

Question 2.a. *(10 marks)*
Explain the notions of a *class* and that of an *object*.

Question 2.b. *(5 marks)*
What are *object references*?

Question 2.c. *(10 marks)*
Explain the purpose of *visibility modifiers*. As part of your explanation you should provide an example.

Modifier	Class	Package	subclass	world
public	✓	✓	✓	✗
protected	✓	✓	✓	✗
no modifier	✓	✓	✗	✗
private	✓	✗	✗	✗

Question 3: Basic Types. *(25/225 marks)*

Question 3.a. *(5 marks)*
Name at least three Java's primitive types.

Number, char, boolean
↳ floating point
↳ integer

Question 3.b. *(10 marks)*
What are the main differences between Java primitive types and Java object types.

Question 3.c. *(10 marks)*
Explain the notion of a *wrapper* class.

↳

Question 4: Methods. *(25/225 marks)*

Question 4.a. *(10 marks)*

Explain the pass-by-value mechanism.

Question 4.b. *(15 marks)*

Consider the following method definitions.

```
private static void f( int a ) {
    int b = a + 1;
    g( a, b );
}

private static void g( int b,, int a ) {
    int c = b + a;
    System.out.println( c );
}
```

Show how the method call `f( 2 )` is evaluated using the pass-by-value mechanism. Your explanation should include drawings of the intermediate states of the stack.

Question 5: Class Design. *(25/225 marks)*

Fota Wildlife Park has several `Animals`. Each `Animal` has *eating*, *roaming*, and *noise* behaviour. An `Animal` eats either grass or meat. Currently Fota has a `Hippo`, a `Lion` (`Feline Animal`), and a `Wolf` (`Canine Animal`). *However, they are* expecting *a `Dog`, a `Tiger`, a `Cat`, and many more `Feline` and `Canine Animals`.*

The `Hippo` eats grass and roams alone. The `Feline Animals` roam alone and eat meat. The `Canines` roam in packs and also eat meat.

Model the behaviour of Fota's `Animals` using abstract classes in the higher levels and concrete classes at the final level of the class hierarchy. The noise behaviour should be implemented by printing the name of the `Animal` that makes the noise. Write each class and abstract class on a separate sheet in your answerbook.

Keep in mind that your class design should support the addition of new `Animals` with as little coding effort as possible.

Question 6: Inheritance. *(25/225 marks)*

Question 6.a. *(10 marks)*

What is *inheritance*? Your answer should use and explain the the notions of 'being more specific' and 'being more general'.

Question 6.b. *(5 marks)*

State three advantages of inheritance.

Question 6.c. *(5 marks)*

What is *overriding*?

Question 6.d. *(5 marks)*

What is *overloading*?

Question 7: Recursion. *(25/225 marks)*

Question 7.a. *(10 marks)*

Explain the notion of recursion.

Question 7.b. *(15 marks)*

Using recursion, implement a method that numbers in binary notation. The argument of the method is an `int`. The method prints the value of the argument in binary. You may assume that the argument is non-negative. The method should use recursion to achieve its task. **Important: add comments that explain the algorithm.** You are not allowed to use an iterative solution.

Question 8: Iterators and Generics. *(25/225 marks)*

Question 8.a. *(5 marks)*

Explain the notion of an `Iterator`.

Question 8.b. *(10 marks)*

Provide code to print the members of a given `ArrayList` object. Your code should use an `Iterator`. You are not allowed to use a `for` loop and you are not allowed to refer to the `size( )` of the `ArrayList`.

Question 8.c. *(10 marks)*

Provide a generic class `Pair` for representing pairs of things. The class should provide a constructor, a method for getting the first member of the pair, and a method for setting the second member of the pair. The class should be parameterised over two types: one for the first member and one for the second member of the pair.

Question 9: Exceptions. *(25/225 marks)*

Joe is an avid scientist. In his spare time he conducts experiments with his computer. Some of his experiments involve Joe's own `JoeThermometerConnection` class, the purpose of which is to get temperature readings from his outdoor thermometer. The class provides the following constructor and instance methods.

`JoeThermometerConnection( )`: Make a connection with the thermometer.

`double getTemperature( )`: Get the current temperature in degrees Celsius.

`void close( )`: Close the `JoeThermometerConnection` connection.

Joe's thermometer is faulty, which causes errors.

Using proper exception handling, demonstrate how to use Joe's class to (1) make a connection with the thermometer, (2) read and print the current temperature, and (3) close the connection with the thermometer. The exception handling should provide as much details as possible about the cause of failure, and print the stack trace which led to the failure.