Thursday 26 April 2018
9.30 am – 11.30 am
(Duration: 2 hours)

DEGREES OF MSc Information Technology, Software Development and IT Cyber Security

PROGRAMMING

(Answer all 6 questions.)

This examination paper is worth a total of 75 marks

The use of a calculator is not permitted in this examination

INSTRUCTIONS TO INVIGILATORS

Please collect all exam question papers and exam answer scripts and retain for school to collect. Candidates must not remove exam question papers.

This page is intentionally left blank

1. Consider the following lines of code:

```
Console bigConsole = new Console(100,100); // This is the big
    console
Console smallConsole = new Console(10,10); // This is the small
    console

bigConsole.println("Boltzmann");
smallConsole.println("Einstein");

bigConsole = smallConsole;

bigConsole.println("Maxwell");

String name = "Feynman";
```

(a) Please say whether the name "`Boltzmann`" is printed on the small console or on the big console and explain your answer.

[2 marks]

(b) Please say whether the name "`Einstein`" is printed on the small console or on the big console and explain your answer.

[2 marks]

(c) Please say whether the name "`Maxwell`" is printed on the small console or on the big console and explain your answer.

[3 marks]

(d) Is it possible to print the string `name` on the big console? If yes, how? If no, why?

[3 marks]

2. Write a method `findPrimeNumbers` that prints on the screen all the prime numbers between 1 and 100 (included). A prime number is an integer number that can be divided exactly only by `1` and by itself: For example, `7` is a prime number because can be divided exactly only by `1` and by `7`, while `8` is not a prime number because it can be divided exactly by `1`, by `2`, by `4` and by `8`.

Marks will be awarded as follows (10 in total):

- 1 marks: method header;
- 4 marks: part of the method that checks whether a number is prime or not;
- 4 marks: part of the method that aims at finding all prime numbers between 1 and 100;
- 1 marks: general structure of the method.

Marks will not be deducted for trivial syntax errors.

3. A Java program includes the following definitions:

```
int k,l;
boolean test,a,b,c;
String alpha,beta;
```

(a) Consider the following statement, written after k and l have been assigned a value, in the same program:

```
test = k > l ? false : true ;
```

What is the value of test if the value of k is greater than the value of l? And what is the value of test if k and l have been assigned the same value?

[2 marks]

(b) Consider the following statement:

```
test = a && b && c;
```

Write the value of test for all possible values of a, b and c (there are eight possible combinations).

[3 marks]

(c) Explain the difference between the following three operations:

```
k / l;
k % l;
k / (double)l;
```

[3 marks]

(d) Explain the difference between the following two operations:

```
k + l;
alpha + beta;
```

[2 marks]

4. Imagine you need to write a programme to add up all integers (whole numbers) between `0` and `N` (inclusive; i.e. `0+1+2+3+…+N`). Write a Java programme that does this using **recursion**. Note that you can do recursion by repeatedly calling a method, or repeatedly creating new class instances and calling a method – either are permissible. Include a main method that computes and prints the output for `N = 10`.

Marks will be awarded as follows (10 in total):

- 4 marks: correct use of the concept of recursion;
- 2 marks: correct logic within method (checking for stopping);
- 2 marks: correct calling (i.e. calling method within method, or making new object within method);
- 2 marks: correct main method.

Marks will not be deducted for trivial syntax errors.

5. You are making a system to store and process student matric numbers and their grades in Programming IT. The matrics and grades are stored in a text file called grades.txt with the following format:

```
0905543 A2
2034518 C1
2237786 D1
...
```

i.e. the matric number, followed by some whitespace and then the grade. You can assume that there will always be a single whitespace and only one entry per line and no more than 200 students in the file.

(a) Write code to load the entries from the file and create and populate two arrays (one for matric and one for grade). Justify your choice of **type** for the arrays. Your code can all be in a `main` method (no need to design any classes).

[5 marks]

(b) You have been asked to search through the two arrays to find the matric corresponding to the highest grade. If more than one student shares the highest grade, your code should output the **first** student with this grade in the array. As a first step, write the code for the following helper method that returns true if `g1` is a better or equal grade than `g2`. Grades are a letter followed by a number where `A` is the highest letter and the lower the number the better. For example, `A2` is better than `C1` and `B1` is better than `B2`. Hint: the `String.charAt(int pos)` method returns the `char` at a particular position in a string.

[4 marks]

(c) Write some code to add to your `main` method in (a) that will use the helper method in (b) to search through the array and then print the matric and grade corresponding to the highest grade (note: you can assume that there will always be at least two students).

[5 marks]

(d) State one reason why using two arrays like this to hold data is not sensible.

[1 marks]

(e) An alternative solution involves creating an object that will store the data for a single student. Write the code that designs such a class (called `StudentGrade`), with suitable getting methods and a `toString` method to display a student's grade in the format:

```
<matric> = <grade>
```

You can assume that once a student object has been created, its matric and grade will not change.

[6 marks]

(f) The `Comparable` interface in Java allows all sorts of objects to be sorted. For a class to implement the `Comparable` interface, it must implement the `compareTo` method. Fill in the code in "`position A`" in the following code so that it returns

1 if the current object has a better **or equal** grade than the other object. Use the helper method defined in part (b):

```
public int compareTo(StudentGrade other) {

     // position A

}
```

[3 marks]

(g) Write some code that creates an array of your StudentGrade objects, populating it from the values stored in the arrays you created in part (a). Sort the array (using Arrays.sort()) and print out the ordered matrics and grades using the object's toString method.

[6 marks]

Marks will not be deducted for trivial syntax errors.

6. The following code describes a very simple GUI in which the text in a `JTextField` (`inputField`) is copied into a `JLabel` (`outputLabel`) when the user clicks on the "`copy`" button:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class SimpleGUI extends JFrame implements ActionListener
{
      private JTextField inputField;
      private JTextField outputField;
      private JButton copyButton;
      public SimpleGUI() {
            this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            this.setSize(300,300);

            JPanel contentPanel = new JPanel();
            this.add(contentPanel);


            inputField = new JTextField("some text");
            contentPanel.add(inputField);
            outputField = new JTextField(10);
            contentPanel.add(outputField);
            copyButton = new JButton("copy");
            // part A

            contentPanel.add(copyButton);

            this.setVisible(true);
      }

      public void actionPerformed(ActionEvent e) {
            // part B
      }

      public static void main(String[] args) {
            new SimpleGUI();
      }
}
```

(a) Write the code missing at "`part A`" that will make the system respond to the user clicking on `copy` button.

[2 marks]

(b) Write the code missing at "`part B`" that will perform the copying actions when the button is pressed. The code should check the source of the event and wipe the text in `inputField` once copied.

[3 marks]

Marks will not be deducted for trivial syntax errors.