Assignment 6

Complete the following programming exercises in **Java**. Submit your answers as a set of **.java** files, each file properly formatted/indented. Files of the wrong format (e.g. .zip, .doc. .txt) may not be graded. As indicated below, not all questions are worth the same marks.

This assignment is due 6 pm Wednesday 1st April. No late submissions accepted.

Q1. Write a class in Java called Point2D. The class stores a point's x, y coordinates as private data members and must have the following methods:

- A constructor that creates a new point with given x and y coordinate values
- A method Point2D midPoint(Point2D p) which returns a new point object which is the midpoint of the line segment between point p and this point
- A String toString() method which overrides the toString() method of the *Object* class (see https://docs.oracle.com/javase/8/docs/api/java/lang/Object.html) and returns a string representation of the point.

Your method should function correctly with the main() method below, which should print out the string "(3.5,2.0)"

```
public static void main(String args[]) {
   Point2D p1 = new Point2D(1,1);
   Point2D p2 = new Point2D(6,3);
   System.out.println(p1.midPoint(p2));
}
```

[20 marks]

Q2. Add a static method to your Point2D class static Point2D farthestFromOrigin(Point2D[] points) which returns a reference to the point in the array that is farthest to the origin.

Explain why this method should be static.

[10 marks]

Q3. Write a method in Java static int charCount(String s, char c) which counts the number of occurrences of the character c in the string s. You may use any of the Java String methods, documented at https://docs.oracle.com/javase/8/docs/api/java/lang/String.html, to implement your method. Write your method in a Java file CharCount.java.

[10 marks]

Q4. Included below is a set of five C++ files which together implement a bank application. The files define four classes: an abstract Account class, a CurrentAccount and LoanAccount class, that both inherit from Account, and a Bank class. The Bank_App.cpp file contains test code to create a Bank, add accounts to it and the call methods on the Bank.

Rewrite this entire C++ application as a set of Java classes. Replicate the structure and functionality of the original code as exactly as possible. The test code in Bank_App.cpp should be replicated in a Java class called Bank_App and, when run, the code should output exactly the following:

```
Account type: Current Account
==============================
account number: 1001 has balance: 300 Euro
   - Overdraft limit: 200

Account type: Loan Account
==============================
account number: 2002 has balance: 250000 Euro
   - Loan term: 360 months

The total amount of outstanding loans is = 550000 euros.
```

[50 marks] (The code for each class is worth 10 marks)