DEPARTMENT OF PHYSICS & ASTRONOMY
3459 EXAM-1
14:00 - 17:00 : February 25th 2010

Please read the exam guidelines, rules, instructions and marking criteria at `http://moodle.ucl.ac.uk/mod/wiki/view.php?id=13963&page=Mid-term+exam` (linked from the *Exams and coursework* page).

This exam is worth 25% of your final mark for the course and is made up of two parts:

- 15 multiple-choice questions, worth 7.5% of your final mark;
- a programming exercise, worth 17.5% of your final mark.

You should endeavour to spend no more than 30 minutes on the multiple-choice section.

Both the answers to the multiple-choice questions (in a file called `mc.txt`) and Java source code of your solution to the programming exercise should be uploaded using Moodle under the section headed “Exam 1”.

DEPARTMENT OF PHYSICS & ASTRONOMY
3459 EXAM-1
PROGRAMMING EXERCISE

You will write Java classes and methods to read data from a URL, analyse the data and present the results.

Monitoring stations at various positions around the world have been measuring the atmospheric concentration of a specific chemical, which we will call X. The results are available in the text file at this URL:

`http://www.hep.ucl.ac.uk/undergrad/3459/exam-data/x.txt`

The file is arranged in sections, one for each monitoring station. Each section begins with a line of the form

```
Location: -48.437904  -168.208547
```

where the numbers are respectively the latitude and longitude of the station in degrees. Subsequent lines in each section contain measurements made at this station in the form

```
2003 02 03 13:22:03 5.267538
```

where the fields are respectively: year; month (1–12); day of month; time (hours:minutes:seconds) in 24-hour format; concentration of the chemical X in ppb (parts per billion).

You should write a program using appropriate classes and methods to read the data from the URL, store them in suitable collection objects, and carry out the following tasks:-

- For each location calculate the mean concentration of X.
- Print the location of the monitoring station with the largest concentration in each hemisphere: the northern hemisphere is defined by positive latitude, the southern hemisphere by negative latitude.
- Print the mean of
  - all measurements made before the year 2000;
  - all measurements made in 2000 and later years.

DEPARTMENT OF PHYSICS & ASTRONOMY
3459 EXAM-1
MULTIPLE CHOICE

You should endeavour to spend no more than 30 minutes on the multiple-choice section.

DO NOT WASTE TIME CODING THE QUESTIONS TO GET THE ANSWERS

You should enter your answers to the multiple-choice questions into a text file called `mc.txt` created using a text editor such as WordPad. The file should have the following format:

```
YOUR NAME
01 a
02 b
03 c
...
14 b
15 a
```

There is exactly one correct answer to each question.

Q1 What does `static` mean in the following code extract?

```
public class Test {
  public static int numLoops = 10;
    // ...
}
```

(a) The value of the variable `numLoops` cannot be changed
(b) The value of the variable `numLoops` can only be changed by a method belonging to the class `Test`.
(c) There is a separate variable called `numLoops` belonging to each object of type `Test`.
(d) There is only one variable called `numLoops` belonging to the class `Test`.

Q2 Why might this code not compile?

```
public class TestClass {
  public void action() { }
  public static void main(String[] args) {
    action();
  }
}
```

(a) The `main` method does not take the correct argument type.
(b) The `action` method does not contain a `return` statement.
(c) The `main` method is static and so cannot call the non-static method `action` without first creating an instance of `TestClass`.
(d) It is valid code and will compile successfully.

Q3 How many lines of output will the following line print to the screen?

```
int a=0;
do {
  System.out.println("test");
  a++;
}
while (a<5);
```

(a) 0
(b) 4
(c) 5
(d) 6

Q4 Which of the following methods will NOT compile when incorporated into a class?

(a) `private int funcA() {return 1.0;}`
(b) `public void funcB(int j) {}`
(c) `private static void funcC() {}`
(d) `private double funcD() {return 1.0;}`

Q5. What does the following line do?

```
int a[] = new int[10];
```

(a) It creates an array that can contain one integer, and assigns to this integer the value 10.
(b) It creates an array that can hold 10 integers, referred to as `a[1]` ... `a[10]`.
(c) It creates an array that can hold 10 integers, referred to as `a[0]` ... `a[9]`.
(d) It creates an array that can hold 11 integers, referred to as `a[0]` ... `a[10]`.

Q6. What will happen if you attempt to compile and run the following code fragment?

```
int a = Integer.parseInt("10.5");
System.out.println(a);
```

(a) `10.5` being printed to the screen
(b) `10` being printed to the screen
(c) A compilation error
(d) The program crashing and a `NumberFormatException` being thrown

Q7 Which of the following lines will successfully create a `Vector` object and add a `String` element?

(a) `Vector<String> v = new Vector<String>(); v.add("2");`
(b) `Vector<String> v = new Vector<String>(); v.add(2);`
(c) `Vector<String> v = new String[3]; v.add("two");`
(d) `Vector<String> v = {"two"};`

Q8. Which of the following would not be a valid line of code (i.e. would result in a compilation error) if inserted after the `"//here"` line?

```
public class Test {
  public static void first() {System.out.println("first");}
  public void second() {System.out.println("second");}
  public static void main(String[] args) {
    Test t = new Test();
    // here
  }
}
```

(a) `first();`
(b) `second();`
(c) `t.second();`
(d) `Test t2 = t;`

Q9 If the following code fragment were executed, what would be printed to the screen?

```
public class NameBadge {
  private String name;
  public void setName(String fullName) {name = fullName;}
  public String getName() {return name;}
  // ...
  public static void main(String[] args) {
    NameBadge a = new NameBadge();
    NameBadge b = a;
    a.setName("Danny Dog");
    b.setName("Peppa Pig");
    System.out.println(a.getName());
  }
}
```

(a) `a`
(b) `b`
(c) `Danny Dog`
(d) `Peppa Pig`

Q10 What type of object is created by this line of Java code?

```
HashMap<String,Integer> h = new HashMap<String,Integer>();
```

(a) A map that can use strings as keys to access integer numbers.
(b) A map that can use integer numbers as keys to access strings.
(c) An array that can hold a mixture of strings and integer numbers.
(d) A string of digits representing an integer number.

Q11 Under what circumstance will the word “Finally!” be printed by the following program?

```
public class TestFramework {
  public static void main(String[] args) {
    try {
      test();
    } catch (Exception e) {
      System.out.println("Exception!");
    } finally {
      System.out.println("Finally");
    }
  }
  public static void test() throws Exception {
    // ...
  }
}
```

(a) Only if the method `test` throws an exception.
(b) Only if the method `test` does not throw an exception.
(c) Always.
(d) Never.

Q12 What will be printed to the screen by the following code fragment?

```
int i=0;
Scanner s = new Scanner("exp(0) = 1");
while (s.hasNext()) {s.next(); i++;}
System.out.println(i);
```

(a) `1`
(b) `2`
(c) `3`
(d) `4`

Q13 What will the following line print to the screen?

```
System.out.println("10"+1);
```

(a) `1`
(b) `10`
(c) `11`
(d) `101`

Q14 What will be printed to the screen by the following program?

```
public class Test {
  private int c = 0;
  public static void main(String[] args) {
    Test t = new Test();
    t.c = 1;
    System.out.println(t.c);
  }
}
```

(a) 0
(b) 1
(c) c
(d) An exception will be thrown because the code attempts to change the value of a private member variable.

Q15 How could you test the `Counter` objects `a` and `b` in this program to verify that they contain the same value of `count`?

```
public class Counter {
  private int count;
  public Counter(int val) {count=val;}
  public static void main(String[] args) {
    Counter a = new Counter(3);
    Counter b = new Counter(3);
    // Test here!
  }
}
```

(a) Use `if (a=b)` ...
(b) Use `if (a==b)` ...
(c) Use the Java built-in `equals` method.
(d) Define your own `equals` method for the class.

END OF PAPER