CS118

THE UNIVERSITY OF WARWICK

First Year Examinations: April 2014

Programming for Computer Scientists

Time allowed: 2 hours

Attempt **ALL** questions.
Each question carries 20 marks.

Most answers require a single statement or a short piece of Java code to be written. Do not give complete programs or supply irrelevant declarations or input-output statements unless asked for explicitly. Number questions clearly but do NOT start each answer on a new page.

Read the instructions on the answer book carefully and make sure that the particulars required are entered on each answer book.

Continued

---

1. **Primitive Data Types**

   (a) Describe what is meant by variable *declaration* and *assignment* with reference to Java's primitive data types. Give valid examples of variable declaration and valid examples of variable assignment. [4]

   (b) Provide answers to the questions given below.

      i. Is `String` one of Java's eight primitive data types? If it is not, state why this is the case.

      ii. Is it possible to convert a numeric value to another numeric type? What are the benefits of doing so, and should the programmer be aware of any potential pitfalls?

      iii. When a variable is declared, what initial value does it contain?

   [6]

   (c) i. Write a short program that, given an 8-bit binary number (stored in an array), converts this number to an unsigned, positive, base-10 integer value. Given the input `00001111` your code should return `15`; given the input `10101010` your code should return the value `170`.

      ii. Modify your code so that it also outputs the unsigned, positive, base-10 integer value represented by flipping the bits. In the case of the input `00001111`, as well as outputting `15`, your code should also output `240` (i.e. the value of `11110000`).

      iii. Describe how you would modify your code so that the most significant bit (in the input) was to act as a sign bit; that is, the left-most bit determines whether the number is positive (when it is 0) or negative (when it is 1).

   [10]

Continued

---

2. **Control statements**

(a) Under what conditions will the statements `p1`, `p2`, `p3` and `p4` be executed?

```
if (a<0) { p1 } else if (b!=0) { p2 }
else if (a==b) { p3 } else if (a==0) { p4 }
```

[4]

(b) Rewrite the following program statement

```
if (a<b) if (b<0) { p1 }
else if (a==b) if (b==0) { p2 } else if (b>0) { p3 }
```

in the simplest form you can. [2]

(c) Give the output of the following program.

```
1   class TooMany {
2       static void howMany (int k) {
3          switch (k)  {
4           case 1 : System.out.print("one");
5           case 2 : System.out.print("too");
6           case 3 : System.out.println("many");
7           }
8       }
9
10     public static void main(String[] args) {
11          howMany(3);
12          howMany(2);
13          howMany(1);
14     }
15  }
```

[3]

State the difference to the output if a break statement were added to the end of:

(i) line 4;

(ii) lines 4 and 5;

(iii) lines 4, 5 and 6.

[3]

Continued

(d) The Speaking Clock is a national telephone service in the UK that has been around since 1936. Today it receives around 135 million calls a year from the general public and from companies, such as British Rail for example, who wish to know the exact time to the nearest second. The Speaking Clock receives an accurate time feed and converts this to spoken English. For example, the feed 21.36:25 is converted to

> "At the third stroke the time will be nine thirty six and twenty five seconds"

Write a program that takes the time feed as an integer (i.e. 213625, in the above example) and from this is able to produce the correct text that can then be fed to a speech synthesiser for the telephone announcement. [8]

---

3. **Arrays**

   (a) Write a method `MaxMin` that finds the maximum and minimum numbers in an array. You should use an accumulating parameter to store the results. [6]

   (b) You are asked by a customer to re-implement their `BubbleSort` method. Bubble sort is a simple $O(n^2)$ sorting algorithm that works by repeatedly stepping through an array to be sorted comparing adjacent items and swapping them if they are in the wrong order. The customer specifies that the resulting array should be in ascending order. This means that after the first pass through the array the largest element should have bubbled its way to the top; the same procedure should be applied to the remaining $n - 1$ elements. [8]

   To assist you, an example is provided below:

   Bubble sorting the list $9, 8, 4, 6, 3$

   First pass:

   compare pair $(9, 8)$; swap as $9 > 8$; result $8, 9, 4, 6, 3$
   compare pair $(9, 4)$; swap as $9 > 4$; result $8, 4, 9, 6, 3$
   compare pair $(9, 6)$; swap as $9 > 6$; result $8, 4, 6, 9, 3$
   compare pair $(9, 3)$; swap as $9 > 3$; result $8, 4, 6, 3, 9$

   Second pass:

   repeat process for list $8, 4, 6, 3$

   Etc.

   (c) Describe the terms *monomorphic* and *polymorphic* in the context of arrays. Using some Java code as an example, demonstrate how it is possible to implement seemingly polymorphic arrays in Java. [6]

Continued

4. **Classes**

   (a) Write an abstract class called `ClassOne` that contains two methods, `methodA` and `methodB`, one of which is implemented and one of which is not. [3]

   (b) Write a concrete class called `ClassTwo` that extends `ClassOne`. Do not use method overriding in your implementation. [2]

   (c) Write an implementation of `ClassTwo` that does use method overriding. [2]

   (d) Write an interface called `InterfaceOne` that contains type signatures for two methods called `methodC` and `methodD`. [2]

   (e) Now write a new concrete class called `ClassThree` that is based on `ClassOne` and `InterfaceOne`. [4]

   (f) Explain the term multiple inheritance. You should use the definitions above to illustrate your answer. [2]

   (g) Using the following class definition

```
public class C  {
   public  static int a;
   private static int b;
   public  int x;
   private int y;
}
```

   explain how you would access the values of `a`, `b`, `x` and `y` from another class. If further code is needed in class `C` then you should document this, being careful to explain the selection of any modifiers that are used. [5]

5. **Arguments, error handling and OOP**

   (a) What is the difference between passing by value and by reference? Which of the two mechanisms is used for passing arrays and why? [4]

   (b) Describe Java's approach to exception handling. You should include in your answer:

   (i) a description of the supporting classes in the Java API,
   (ii) the three fundamental components of error handling in Java,
   (iii) illustrative code in which three type of exceptions can occur,
   (iv) an explanation of the `finally` construct.

   [8]

   (c) Object oriented programming is based on a number of defining principles. List what they are and give a brief explanation of each. [8]

End