University of Glasgow

Friday 26 April 2013
2.00 pm – 4.00 pm
(Duration: 2 hours)

DEGREE OF MSc in Information Technology

PROGRAMMING

(Answer all 6 questions.)

This examination paper is worth a total of 75 marks
(each of Questions 1-5 carries 10 marks, whilst Question 6 carries 25 marks)

You must not leave the examination room within the first hour or the last half-hour of the examination.

THE USE OF CALCULATORS 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 intentionally left blank

1. Consider the following program:

```
public static void main(String [] args)
{
   int a, b, c, sum;

   a = 1 ;
   b = 2 ;
   c = 5 ;

    sum = a+b ;
    calculateSum(a,c) ;

    System.out.println(sum) ;
}

private static int calculateSum(int first, int second)
{
   int sum ;
   sum = first+second ;

   return sum ;
}
```

(a) Give the console output, explaining your answer.

[2]

(b) Explain the meaning and function of the qualifiers `private` and `static`.

[2]

Now consider the following method header:

```
private static int calculateCumulative(int start, int end)
```

(c) Complete the body of `calculateCumulative` so that the method returns the sum of all integers between `start` and `end` inclusive. You can assume that the value of `start` is always less than or equal to the value of `end`.

[4]

(d) Modify the body of the helper method `calculateCumulative` to deal with the case where the value of `start` is larger than the value of `end`. Your revised method body should still calculate the sum of all the integers between the two given method parameters (inclusive).

[2]

2. Consider the following program:

```
public static void main(String [] args)
{
   boolean done = false;
   do {
      System.out.println("X") ;
   } while (done) ;
}
```

(a) Give the console output, explaining your answer.

[2]

(b) Explain what happens if the variable `done` is initialized to `true`.

[1]

Now consider the following method:

```
private static void drawLine(int length)
```

(c) Complete the body of `drawLine` so that it prints '`X`' characters exactly `length` times on the same line (with no spaces between consecutive '`X`' characters), followed by a new line. You can assume that `length`>0.

[3]

Now consider the following method:

```
private static void drawLines(int length, int val)
```

(d) Complete the body of `drawLines` so that it that it produces the following console output (that is, 2*n*+1 lines in total, such that *n*+1 lines each contain *m* 'X' characters, and *n* lines are blank, with these different types of lines alternating, where *n* is the value of `val` and *m* is the value of `length`):

```
XXXXXXXX

XXXXXXXX

XXXXXXXX

...

XXXXXXXX

XXXXXXXX
```

You can assume that `length`>0 and `val`>0.

[4]

3. Consider the following method:

```
private static double calcDouble(double a, double b, int n)
{
   double y ;
   y = a*n+b ;
   return(y) ;
}
```

(a) Explain why variable `y` is of type `double`.

[1]

Now consider the following method header (assume that `a>0` and `b<10`):

```
private static int calcInt(double a, double b)
```

(b) Complete the body of `calcInt` so that it returns the smallest integer `n` such that `a*n+b > 10`.

[4]

Now consider the following method header (assume that `a>0` and `b<10`):

```
private static void writeLine(double a, double b)
```

(c) Complete the body of `writeLine` so that it uses the value of `n` calculated by the method `calcInt` from Part (b) as input to a `switch` statement. The `switch` must print to the console "`low`" if `n<=3`, "`medium`" if `n>3` and `n<=7`, and "`high`" otherwise.

[5]

4. Consider the following classes:

```
public class Square implements Shape
{  private double sideLength;

   public double getArea()
   {  return sideLength * sideLength;
   }
   public double getPerimeter()
   {  return 4 * sideLength;
   }
}

public class Circle implements Shape
{  private double radius;

   public double getArea()
   {  return Math.PI * radius * radius;
   }
   public double getPerimeter()
   {  return 2 * Math.PI * radius;
   }
}
```

(a) What type of Java entity is `Shape`?

[1]

(b) Give code for `Shape`.

[3]

(c) Write code for a class `Triangle` representing an equilateral triangle (i.e., a triangle in which all sides have the same length) that implements `Shape`.

Hints: **(i)**: use the same instance variable as in `Square`.

(ii): the area of an equilateral triangle is $\sqrt{3}n^2/4$, where n is the length of one side.

(iii): `Math.sqrt(n)` gives the square root of `n`, for `n` of type `double`.

[3]

Consider the following class, which is passed `Circle` objects one by one via the method `addCircle`. The total area of all the circles passed is recorded, together with the `Circle` object having the largest circumference. These attributes are then returned via the accessor methods `getTotalArea` and `getMaxPerimeter`.

```
public class CircleStats {
{  private double totalArea;
   private Circle maxPerimeterObj;

   public CircleStats()
      totalArea = 0;
      maxPerimeterObj = null;
   }
```

```
   public void addCircle (Circle c)
   {  totalArea += c.getArea();
      if (maxPerimeterObj == null ||
         c.getPerimeter() > maxPerimeterObj.getPerimeter())
         maxPerimeterObj = c;
   }

   public double getTotalArea()
   {  return totalArea;
   }

   public Circle getMaxPerimeterObj() {
   {  return maxPerimeterObj;
   }
}
```

(d) Indicate how to adapt `CircleStats` so that it can be passed objects of type `Square` and `Triangle` in addition to `Circle` objects.

[1]

(e) What is the advantage of the adaptation to the code that you made in Part (d)? What importance does `Shape` have in this respect?

[2]

5. Let `word` be a string of length `len`. A substring of `word` is a sequence of contiguous characters of `word`. The purpose of this question is to implement a recursive method for finding all substrings of `word`. For example, the substrings of the string `"rum"` are the following seven strings:

`"r"`, `"ru"`, `"rum"`, `"u"`, `"um"`, `"m"`, `""`.

The following is an outline of the class `SubstringGenerator` containing a recursive method `getSubstrings` for generating all the substrings of word.

```
public class SubstringGenerator
{
  private String word;
  private int len;

  public SubstringGenerator(String s)
  { word = s;
    len = word.length();
  }

  public String [] getSubstrings()
  { if (len==0)
    { // Part (a)
      return substrings;
    }
    else
    { String [] substringsFirst = new String [len];
      // Part (b)
      // Part (c)
      SubstringGenerator restClass =
                              new SubstringGenerator(rest);
      // Part (d)
      int numRest = substringsRest.length;
      String [] substrings = new String[len + numRest];
      // Part (e)
      return substrings;
    }
  }
}
```

Overall, the method works on the basis of enumerating all substrings of `word` that start with the first character (there are `len` of them) and then enumerating all the substrings of `word` that are obtained by removing the first character.

(a) Give code (a single line is sufficient) at the point in the code labelled by the comment "Part (a)", in order to complete the base case of the recursion.

[1]

(b) Give code (two lines are sufficient) at the point in the code labelled by the comment "Part (b)". These should populate `substringsFirst` with all substrings of `word` beginning with the first character of `word`.

[3]

(c) Give code (a single line is sufficient) at the point in the code labelled by the comment "Part (c)". This should assign to a string `rest` the contents of `word` obtained by removing the first character of `word`.

[1]

(d) Give code (a single line is sufficient) at the point in the code labelled by the comment "Part (d)". This should assign all substrings of `rest` to the string array `substringsRest`.

[1]

(e) Give code (two lines are sufficient) in the code labelled by the comment "Part (e)". This should assign all substrings contained in `substringsFirst` and `substringsRest` to the string array `substrings`.

[4]

6. Suppose that you are required to design a program with a small Graphical User Interface (GUI) that is capable of carrying out the following actions:

- allow the user to input an integer, one at a time;
- display the minimum, maximum and average of the integers entered so far;
- allow the user to clear any displayed numbers, and clear the stored values of the minimum, maximum and average numbers;
- allow the user to quit the program.

(a) Give a diagram showing a suitable layout for the GUI that will enable it to support the functionality described above. Annotate your diagram to show which JComponents appear in your GUI.

[8]

(b) State which layout manager object should be constructed in order to support the GUI layout illustrated in your diagram in Part (a).

[1]

For the next three parts of this question, consider a general GUI-based program.

(c) In the Model–View–Controller (MVC) architecture for a program involving a GUI, describe briefly the roles of each of the Model, View and Controller classes.

[3]

(d) Suppose that a program involving a GUI implements the MVC architecture by having separate classes for each of the model, view and controller. Draw a diagram that indicates the relationships between each class. Your diagram should include four boxes – one for each of the main, model, view and controller classes – and arrows between the boxes, where an arrow from box A to box B means that any of the following happen:

- class A uses methods from class B;
- class A creates an instance of class B;
- class A stores an object of class B as an instance variable.

[3]

(e) For each of the model, view and controller classes, explain the relationship(s) between that class and the other two, with reference to the arrows that you drew as part of your answer to Part (d). You do not need to mention the main class here.

[5]

Now consider again the GUI-based program for finding the minimum, maximum and average of a set of integers as covered by Parts (a) and (b) of this question. Suppose that this program is to be implemented using an MVC architecture.

(f) Explain in outline the roles of the model, view and controller class within the context of this example program in particular. (Note that your answer to Part (f) should focus on the specifics of this precise program, whereas your answer to Part (c) focused on a general program with an MVC architecture.)

[5]