Name: . . . . . .

ID Number: . . . . . .

COMP102: Test 2 A

Model Solutions

5 September, 2006

Note: Answers were included for some of the parts of questions 1 and 2 by mistake.
Test 2 B contained alternative questions for this part of the test.

Instructions

- Time allowed: **90 minutes** ($1\frac{1}{2}$ hours).
- Answer **all** the questions.
- There are 90 marks in total.
- Write your answers in the boxes in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- There is some Java documentation at the end of the test paper.
- This test will contribute 20% of your final grade, if it helps your grade.
- Non-electronic translation dictionaries and calculators without a full set of alphabet keys are permitted.

Questions	**Marks**	
1. Basic Java	[33]	
2. Conditionals	[22]	
3. Using a DrawingCanvas	[10]	
4. Loops with Files	[10]	
5. Objects and Fields	[15]	
	TOTAL:	

Student ID: .....

Question 1. Basic Java [33 marks]

(a) [3 marks] What will the following fragment of Java print out?

```
double size = 4.86;
size = size / 2;
System.out.println("size = " + size);

if ( size > 1 )
    size = size + 1.0;
else
    size = Math.min(-4.86, size);
System.out.printf("size = %.1f\n", size);
```

```
2.43
Size = 3.4
```

(b) [4 marks] Consider the following drawShape method.

```
public void drawShape(int n){
    canvas.setForeground(Color.red);
    if (n<=1 || n == 4)
        canvas.setForeground(Color.blue);
    if (n > 2 && n < 5)
        canvas.setForeground(Color.green);
    canvas.fillRect(10,10,100,100);
}
```

(i) [2 marks] What colour rectangle will be drawn if drawShape(1) is called?

```
blue
```

(ii) [2 marks] What colour rectangle will be drawn if drawShape(4) is called?

```
green
```

(Question 1 continued on next page)

Student ID: .....

(Question 1 continued)

(c) [4 marks] What will the following fragment of Java print?

```
int k = 7;
while (k < 10){
    System.out.println("k = " + k);
    k++;
}
```

```
k = 7
k = 8
k = 9
```

(d) [4 marks] Write a frament of Java using a **for** statement that behaves the same as the fragment in **(c)** above.

```
for(int j = 5; j<10; j++){
  System.out.println("num: " + j);
}
```

(e) [4 marks] Write a method called triple which has one *int* parameter and returns a value of type *int*. triple should return the value that is three times the value of its parameter.

```
public int triple (int n){
  return n*3;
}
//or
public int triple (int n){
  int ans = n*3;
  return ans;
}
```

Student ID: .....

(f) [7 marks] The following evenNumbers method has one integer parameter, n. Complete the method so that it prints out all the even numbers from 2 up to and including n, using either a **for** loop or a **while** loop.

```
    public void evenNumbers(int n){
        for ( int j=2; j<=n; j=j+2){
            System.out.println(j);
        }
    }
// or
    public void evenNumbers(int n){
        int j=2;
        while (j<=n){
            System.out.println(j);
            j=j+2;
        }
    }
// or
    public void evenNumbers(int n){
        int j=1;
        while (2*j<=n){
            System.out.println(2*j);
            j++;
        }
    }
```

(g) [7 marks] Suppose the file "data.txt" contains the text:

```
5 fish 4 chips and 3 patties
```

What will the following fragment of Java print?

```
try{
    Scanner ss = new Scanner (new File ("data.txt"));
    while ( ss.hasNextInt( ) ){
        int n = ss.nextInt( ) * 2;
        String w = ss.next( );
        System.out.println(n + " x " + w);
    }
    ss.close( );
    System.out.println("done");
}
catch(Exception e){System.out.println("File error: "+ e);}
```

```
10 x fish
8 x chips
done
```

Student ID: .....

Question 2. Conditionals [22 marks]

The middle method below is intended to return the middle value of three integers — the value that would be second if the values were put in order. For example, middle(8, 17, 2) should return 8.

```
/* This version of middle has errors */
public int middle(int a, int b, int c){
    if (a<b && b < c)
        return b;
    else if (a<b && c<b)
        return c;
    else
        return a;
}
```

The middle method above has errors.

(a) [3 marks] What will middle(8, 17, 2) actually return?

2

(b) [3 marks] What will middle(17, 8, 2) return?

17

(c) [6 marks] Show two further calls to middle using the arguments 2, 8, 17 in different orders, one of which will return the correct answer and the other will return an incorrect answer.

Correct: middle(2, 8, 17) or middle(2, 17, 8) or middle(8, 2, 17)

Incorrect: middle(17, 2, 8)

(Question 2 continued on next page)

Student ID: .....

(Question 2 continued)

(d) [10 marks] Write a correct version of the middle method.

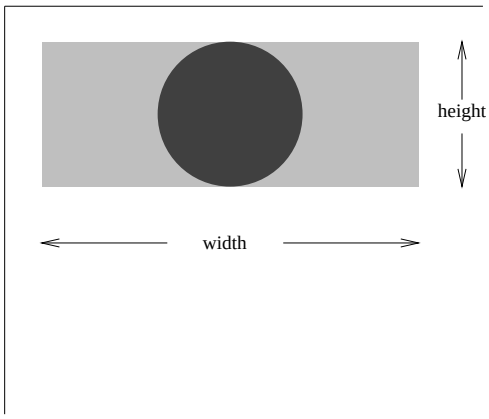
```
\\lots of options. Here are some
    public int middle(int a, int b, int c){
        if ( b < a ){
            int temp = b;
            b = a;
            a = temp;
        }
        if ( b < c )
            return b;
        else if (a < c)
            return c;
        else
            return a;
    }

    public int middle(int a, int b, int c){
        if ( a < b ){
            if ( b < c )
                return b;
            else if (a < c)
                return c;
            else
                return a;
        }
        else if (a < c)
            return a;
        else if (b < c)
            return c;
        else
            return b;
    }

    public int middle(int a, int b, int c){
        if ( (b <= a && a <= c) || (c <= a && a <= c) )
            return b;
        else if ( (a <= b && b <= c) || (c <= b && b <= a) )
            return b;
        else
            return c;
    }
// Be careful to handle the case when two or three of the arguments are equal
```

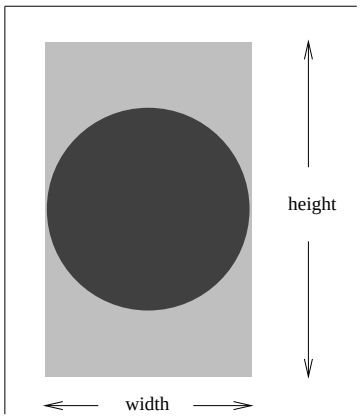
Student ID: .....

Question 3. Using the DrawingCanvas [10 marks]

Complete the following drawIcon method so that it draws a gray rectangle with a black circle inside it. The drawIcon method has two parameters that are the width and the height of the rectangle and a third parameter giving the canvas to draw on.

The top left corner of the rectangle should be placed at (20,20). The circle should be centered in the rectangle and should just touch the long sides of the rectangle. For example, depending on the width and height, the drawIcon method might draw one of the following.





There is documentation on the DrawingCanvas and Color classes at the end of the test paper.

```
  public void drawIcon(int width, int height, DrawingCanvas canvas){
      canvas.setForeground(Color.gray);
      canvas.fillRect(20,20, width, height);
      canvas.setForeground(Color.black);
      if (width>height)
          canvas.fillOval(20+width/2−height/2, 20, height, height);
      else
          canvas.fillOval(20, 20+height/2−width/2, width, width);
  }
//or
  public void drawIcon(int width, int height, DrawingCanvas canvas){
      canvas.setForeground(Color.gray);
      canvas.fillRect(20,20, width, height);
      canvas.setForeground(Color.black);
      int diam = Math.min(width, height);
      int left = 20 + width/2 − diam/2;
      int top = 20 + height/2 − diam/2;
      canvas.fillOval( left , top, diam, diam);
  }


// Note that the width, the height, and the canvas are all passed as arguments;
// they should not be redefined or inputted from the user.
// Note also that the position of a shape is specified by the left and top, not the center
```

Student ID: .....

Question 4. Loops with files [10 marks]

Complete the averageWordLength method below so that it returns the average length of the words in a file. You may assume that that the file contains nothing but words and numbers. The method should only count the words, not the numbers (integers or doubles). The parameter of the method specifies the name of the file. The method should close the file after reading all the words.

There is documentation on the String and Scanner classes at the end of the test paper.

```
public double averageWordLength(String fileName){
    double chars = 0;
    int count = 0;
    try{
        Scanner sc = new Scanner(new File(fileName));
        while (sc.hasNext( )){
            if (sc.hasNextInt() || sc.hasNextDouble())
                sc.next ();
            else{
                count++;
                chars = chars + sc.next (). length ();
            }
        }
        sc.close( );
    }
    catch(Exception e){}
    return chars/count;
}
// or
public double averageWordLength(String fileName){
    int chars = 0;
    int count = 0;
    try{
        Scanner sc = new Scanner(new File(fileName));
        while (sc.hasNext( )){
            if (sc.hasNextDouble()) sc.next();   // (checks for any number, including integers)
            else{
                count++;
                chars += sc.next (). length ();
            }
        }
        sc.close( );
    }
    catch(Exception e){}
    return ((double)chars)/count;
}
// be careful to use double arithmetic, not integer arithmetic to compute the average.
// Note that the file name is the value in fileName, not the String "fileName"
```

Student ID: .....

Question 5. Objects and Fields [15 marks]

This question concerns a program for a visual simulation of rocks being dropped to the ground from different heights. The program contains two classes, RockSimulator and Rock.

Part of RockSimulator is shown below. The simulate method contains a loop that repeatedly constructs a new Rock object, and then simulates it dropping by calling methods on the object.

The Rock class on the facing page represents the state of a falling rock, including its height above the ground and its current speed. At each time step after it is dropped, the height of a rock will decrease, and the speed will increase. The amount the height decreases will depend on the current speed.

```
public class RockSimulator{
    private DrawingCanvas canvas;

    public RockSimulator( ){
        :
        :
    }

    /* Keeps dropping rocks from random heights */
    public void simulate( ){
        while (true) {                                    // drop rocks forever
            Rock rock = new Rock(Math.random( )*400); // make new rock at a random height
            while ( !rock.atGround( ) ){                  // simulate it falling to the ground
                rock.step( );
                this.canvas.clear( );
                rock.draw(this.canvas);

                try{ Thread.sleep(10); }catch(Exception e){} // wait for 0.01 seconds
            }
        }
    }
}
```

(Question 5 continued on next page)

Student ID: .....

Complete the definition of the Rock class:

- Declare two fields to store the state information.
- Complete the constructor, to intialise the state of the rock to have a speed of 0, and have height above ground specified by the argument to the constructor.
- Complete the atGround method to return true if the rock has hit the ground, and false if the rock is still above the ground.
- Complete the step method to change the state by one 10 millisecond time step: the height should decrease by the speed$\times$0.01 and the speed should increase by 0.098.

```
//(Answer lines marked with an asterisk )
class Rock{
    // fields for the current speed and height above ground.

*   private double height;  // height above the ground in meters
*   private double speed;   // speed downwards in meters per second

    /* Constructor: sets initial speed to 0 and initial height to the given value */
    public Rock(double h){
*       this.height = h;
*       this.speed = 0;
    }
    /* atGround returns true only if the rock has reached ground level . */
    public boolean atGround( ){
*       return (this.height <= 0);

//or (though this is yucky)
*       if (this.height <= 0) return true;
*       else                  return false;
    }

    /* Decreases the height by (speed x 0.01), and increases the speed by 0.098 */
    public void step( ){
*       int y = 400-(int)this.height;
*       canvas.fillOval(100, y, 10, 10);
    }

    /* Draws the rock at the current height */
    public void draw(DrawingCanvas canvas){
        int y = 400-(int)this.height;
        canvas.fillOval(100, y, 10, 10);
    }
}
// "this." is optional everywhere here.
// Note that the height is the height of the rock abouve the ground,
//  not the position on the screen - the latter is calculated in draw
```

****