SOLUTIONS

Problem 1 [10 points] General Concepts

Answer the following questions. Be concise and clear.

Ia [1 point] What is 0.3 ÷ 0.346? Give a numerical result that is accurate to at least two decimal places. If you get stuck, show your work.



Ib [1 point] Let *i* be the imaginary number $i = \sqrt{-1}$, x = 2 + 3i, and y = 1 - 2i. What is xy? (2+3i)(1-2i) = (2)(1) + (3i - 4i) + (3i)(-2i) = 2 - i + (6)(-1)(-1) = 8 - i in MATLAB try this:: >> (2+3i)*(1-2i)

- *Ic* [1 point] *OOP* stands for <u>Object-Oriented Programming</u>.
- *1d* [1 point] As opposed to MATLAB, Java is <u>strongly</u> typed.
- *le* [2 points] Explain the principle of *type promotion*. Give a brief example.

types that are related – when used in an expression, the "lower" types are converted to the "higher" types example) 1 / 2.0 = 0.5 without the ".0", Java would use integer division

If [2 points] In terms of a programming language, what is *scope*?

the location in which a variable, method, class is accessible (looking for at least 2) also OK: where the variable can be "seen"/accessed

lg [2 points] Distinguish between a *class* and *object*.

class: blueprint, template: the code used to create an object object: the unique entity used to represent something when running a program

Problem 2 [15 points] Java Language

Fill in the boxes for the following Java code fragments. Consider each problem as a separate Java code fragment. If the code will produce an error or exception during compilation or execution, write **error** as the output.

problem	code fragment	output
2a [1 point]	System.out.println("hi!"); // we've given the answer	hi!
2b [1 point]	System.out.println(9/5);	1
<i>2c</i> [1 point]	<pre>int x = 1; boolean b = (boolean) x; System.out.println(b);</pre>	error
2d [1 point]	<pre>int x = 4; int y = x; x = 2; System.out.println(y);</pre>	4
2e [1 point]	<pre>System.out.println(""+1+2+3);</pre>	123
<i>2f</i> [1 point]	<pre>int x = 1; System.out.println(1+(x++)); System.out.println(x);</pre>	2 2
<i>2g</i> [2 points]	<pre>int x = 2; if (x = 3) System.out.println(x);</pre>	error
<i>2h</i> [3 points]	<pre>int sum = 0; int x; for (x = 4 ; x > -2 ; x) sum += x;</pre>	7
<i>2i</i> [3 points]	<pre>public class Thing { public static boolean d; public static void main(String[] args) { int a; { char b; { int c; } } char c; System.out.println(d); } }</pre>	false
<i>2j</i> [1 point]	<pre>class _ { \$_; _() { System.out.println(_\$_); } } public class { public static void main(String[] _) { new _(); } }</pre>	null

Problem 3 [35 points] Control structures

Problem: Suppose that a device generates a sequence of a random number (1 to 40, inclusive) of random bits. Each bit (0 or 1) has an equal probability of being generated. Complete program **Problem3** that simulates the device by generating a sequence of bits and reporting the following information:

- The entire sequence of bits from left to right.
- The largest number of successively generated 1s.

Sample Session: Suppose the simulation generated a sequence of eight bits in this order, from left to right: 0, 0, 1, 1, 1, 1, 0, 1. For this example, the program would print this report:

```
Sequence: 00111101
Maximum sequence of 1s: 4
```

Specifications:

- Do not use strings, arrays, or objects in your solution, which will be solely contained in one Main Class (Problem3).
- Do not count or report the maximum number of 0s.
- You must use fields **MAXBITS**, **sequence**, and **maxOnes**. Do not write or use additional fields.
- You will need to complete methods **randBit**, **randInt**, and **runSim**, which are specified below. Remember that **Math.random()** returns a random *double* between 0 and 1, including only 0. Do not write methods that have not already been defined in this problem.

```
public class Problem3 {
   public static final int MAXBITS = randInt(1,40); // # of bits in sim
   public static String sequence = "";
                                                     // seq of bits generated in sim
   public static int maxOnes;
                                                     // max number of 1s in sim
   // Run program:
   public static void main(String[] args) {
      runSim(); // run one simulation to set sequence and maxOnes
      System.out.println("Sequence: " + sequence );
      System.out.println("Maximum sequence of 1s: " + maxOnes);
   }
   // Return a random integer (LOW <= integer <= HIGH):</pre>
   public static int randInt(int low, int high) {
      if (low > high) System.exit(0);
      return (int) (Math.random()*(high-low+1))+(int)low ;
   }
               1
                       1
                                       1
                                                     1
   // Return a random bit (0 or 1) using method randInt:
   public static int randBit() {
      return randInt(0,1);
   }
                1
                       1
```

// Problem3 continues on next page

```
// Run the simulation, which updates the sequence of bits (sequence)
// and determines the max sequence of 1s (maxOnes):
public static void runSim() {
   int oneCounter = ____ ; // count of 1s so far
1
   // Generate and process each bit from count = 1 to MAXBITS:
   for ( <u>int count=1</u> ; <u>count <= MAXBITS</u> ; <u>count++</u> ) {
               1
                           1
                               1
                                    1
                                               1
       // generate the current bit:
 3
       int nextBit = randBit();
       // increment count if current bit is 1, else reset count:
 7
       if (nextBit==1) oneCounter++;
                                                     2: if and condition
       else oneCounter = 0;
                                                     2: if clause
                                                     3: else clause
       // update maxOnes if current sequence is greater:
 7
       if (oneCounter > maxOnes) maxOnes=oneCounter;
                                                      2: if
                                                     2: condition
       // update sequence of bits:
                                                     3: if clause
 3
       sequence+=nextBit;
                                               2: comments
                                               1: syntax
                                               -3: shadowing fields (using field
                                                   names for diff. var.)
                                               -4: algorithm fails for sequence
                                                   that ends with a 1
```

} // end for

} // Method main

} // Class Problem3

Problem 4 [40 points] OOP

Problem: You need to complete two classes **Triangle** and **Point** that driver class **Problem4** uses to compare the position of **Triangles** that are supplied as points p1, p2, and p3 in terms of (x, y) coordinates. As shown in the following figure, **Triangle** A is to the left of **Triangle** B. Assume that the **Triangles**' bases will always be aligned with the x axis and that the vertices are specified in this order: p1, p2, p3. Assume p2 is always to the left of p1.



Use principles of good object-oriented programming (especially information hiding for fields!) when using and/or completing the following classes:

- **Problem4**: Method **main** creates two **Triangles**, outputs the area of the **Triangle** created first, and determines which **Triangle** is leftmost. We have written the complete code for you.
- **Point**: This class represents a vertex coordinate (**x**, **y**).
- Triangle: This class represents a triangle with three coordinates (p1, p2, p3), which are objects of class Point.
 - Method **leftmost** compares the current **Triangle** with another **Triangle** by comparing the leftmost horizontal coordinates of each **Triangle**. Return either **Triangle** if neither is leftmost.
 - Method **area** computes the area of the current **Triangle** using this formula: $area = (base \times height)/2$.
 - Other methods: You need to complete the toString method and the constructor. We have provided accessors.

Example session:

```
Area: 2.0
Triangle: A
```

```
public class Problem4 {
     public static void main(String[] args) {
        Triangle t1 = new Triangle("A", new Point(1,2), new Point(0,0), new Point(2,0));
        Triangle t2 = new Triangle("B", new Point(2,3), new Point(1,0), new Point(4,0));
        System.out.println("Area: "+t1.area());
        System.out.println(t2.leftmost(t1));
     }
  } // Class Problem4
  class Point {
     private double x; // horizontal coord
1
     private double y; // vertical coord
1
     // Create a new Point:
        public Point(double a, double b) { x = a; y = b; }
     // Accessors:
        public double getX() { return x; }
        public double getY() { return y; }
  } // Class Point
```

```
class Triangle {
    private String name; // name of Triangle
1
    private Point p1; // (x,y) of p1 (top coord)
1
1
                       // (x,y) of p2 (left, bottom coord)
    private Point p2;
1
    Private Point p3; // (x,y) of p3 (right, bottom coord)
    // Construct a new Triangle called NAME from coords P1, P2, P3:
       public Triangle(String name, Point p1, Point p2, Point p3) {
     1
          this.name = name ;
     1
          this.pl = p1;
     1
          \underline{\text{this.p2}} = p2;
     1
          \underline{\text{this.p3}} = p3;
    }
    // Accessors for coordinates:
       public Point getP1() { return p1; }
       public Point getP2() { return p2; }
       public Point getP3() { return p3; }
    // Compare current Triangle with supplied Triangle (other) and
    // return leftmost Triangle:
       public Triangle leftmost(Triangle other) {
         if (p2.getX() <= other.getP2().getX()) // other.p2.getX() also OK
            return this;
        return t;
                                                                     1: if
    // Compute and return the area of the current Triangle:
                                                                     2: p2.getX
       public double area() {
                                                                     2: <=, correct order
                                         1
                                               1
                                                      1
                                                                     3: other.getP2
          double base
                         = Math.abs( <u>p3.getX()-p2.getX()</u> );
                                                                     3: .getX
                                                                     1: return
                                                                     3: this
          double height = Math.abs( pl.getY()-p2.getY() );
                                                                     1: return
                                               1
                                                      1
                                                                     3: other
                                         1
          return 0.5*base*height ;
                  1
                         1: correct ans
    }
    // Stringify the current Triangle:
       public String toString() {
          return <u>"Triangle: "+name</u>;
                      1
                              1 1
    }
 } // Class Triangle
```

Bonus: [5 points] What is the output for this perfectly legal Java program?

```
class Friend {
  private String Friend;
  private Friend friend;
  public Friend(String Friend) { this.Friend = Friend; }
  public void Friend(Friend friend) { this.friend = friend; friend.friend = this; }
  public String friend() { return friend.Friend; }
}
public class Friends {
  public static void main(String[] friends) {
      { Friend Friend = new Friend("friend.friend"); }
     Friend Friend = new Friend("Friend");
     Friend friend = new Friend("friend");
     Friend.Friend(friend);
      System.out.println(friend.friend());
   }
}
```

```
// Output: Friend
```