QUEENS COLLEGE Department of Computer Science
CSCI 212 First Midterm Exam Fall 2019 10.03.19

Instructor: Alex Ryba

09.05am – 09.55am, Thursday, October 3, 2019

1. Write complete Java code for the `class Circle`. You should use exactly two instance variables `radius` and `center` that have types `double` and `Point`. You should not implement the `class Point`, but you should use its methods as specified by the following UML diagram.

Point
double x,y
double distanceTo(Point p) double angleWith(Point p, Point q)

Your `class Circle` must have exactly four methods. The methods are a constructor (with 2 parameters) and methods called area, circumference and isOutside. The area and circumference are calculated with the formulas πr^2 and $2\pi r$, where r is the radius. (The value of π is written `Math.PI` in Java.) The method `isOutside` tests whether a point is outside the circle (this happens when its distance from the center is larger than the radius).

Answer:

```
public class Circle {
   private double radius;
   private Point center;

   public Circle(double myRadius, Point myCenter) {
      radius = myRadius;
      center = myCenter;
   }

   public double area() {
      return Math.PI * radius * radius;
   }

   public double circumference() {
      return 2* Math.PI * radius;
   }

   public boolean isOutside(Point x) {
      return center.distanceTo(x) > radius;
   }

}
```

2. The blocks that implement 3 functions in the following class have been omitted. Complete the code for the functions in the spaces left for the 3 answers. You can and should apply any useful methods of the standard Java classes `Scanner, File, String`.

```
public class Functions {

   //  (a)  return the 3rd character of a string or return
   //   the blank space character if the string length is less than 3.
   static char thirdCharacter(String w) {
      // this block to be completed
   }

   //  (b)  return the 3rd word in a line of text or the empty string
   //    if the line has fewer than 3 words.
   static String thirdWord(String s) {
      // this block to be completed
   }

   // (c) return the 3rd line of the file whose name is given as the parameter
   //   or return the empty String if there is no such file or it is too short.
   static String thirdLine(String fileName) throws FileNotFoundException {
      // this block to be completed
   }
}
```

Answer:

(a) Complete the function with title line:

```
static char thirdCharacter(String w) {
   if (w.length() < 3) return ' ';
   return w.charAt(3);
}
```

(b) Complete the function with title line:

```
static String thirdWord(String s) {
   String array[] = s.split("\\s", -1);
   if (array.length < 3) return "";
   return array[2];
}
```

(c) Complete the function with title line:

```
static String thirdLine(String fileName) throws FileNotFoundException {
   File f = new File(fileName);
   if (!f.exists()) return "";
   Scanner s = new Scanner(f);
   String line = "";
   for (int i = 0; i < 3; i++) {
      if (!s.hasNextLine()) return "";
      line = s.nextLine();
   }
   return line;
}
```

3. Consider the following Java program.

```
public class Output {
   public static void main(String args[]) {
      DecimalFormat f1, f2, f3;
      f1 = new DecimalFormat("#.##");
      f2 = f1;
      f3 = new DecimalFormat("#.###");
      Double x = 5.6781;
      Double y = x;
      Double z = 1.2340;
      System.out.println(x);                // line (a)
      System.out.println(f1.format(z));     // line (b)
      f1 = f3;
      System.out.println(f2.format(y));     // line (c)
      y = z;
      System.out.println(x);                // line (d)
      System.out.println(f3.format(x) + y); // line (e)
      Double a[] = {x, y, z};
      System.out.println(a[0]);             // line (f)
      a[1] = 1.0;
      a[2] = 2.0;
      x = a[1];
      System.out.println(y);                // line (g)
      System.out.println(a[0] + "," + a[1] + "," + a[2]);   // line (h)
      System.out.println(a[0] + a[1] + a[2]);   // line (i)
      System.out.println(z);                // line (j)
   }
}
```

(a) What is the ouput from the instruction beginning on line (a)?

Answer:

```
5.6781
```

(b) What is the ouput from the instruction beginning on line (b)?

Answer:

```
1.23
```

(c) What is the ouput from the instruction beginning on line (c)?

Answer:

```
5.68
```

(d) What is the ouput from the instruction beginning on line (d)?

Answer:

```
5.6781
```

(e) What is the ouput from the instruction beginning on line (e)?
Answer:

```
5.6781.234
```

(f) What is the ouput from the instruction beginning on line (f)?
Answer:

```
5.6781
```

(g) What is the ouput from the instruction beginning on line (g)?
Answer:

```
1.234
```

(h) What is the ouput from the instruction beginning on line (h)?
Answer:

```
5.6781,1.0,2.0
```

(i) What is the ouput from the instruction beginning on line (i)?
Answer:

```
8.6781
```

(j) What is the ouput from the instruction beginning on line (j)?
Answer:

```
1.234
```

4. Consider the following Java code. Exactly 3 of the numbered lines have syntax errors. Identify the 3 different errors by giving the numbers of the incorrect lines. For each error also give a corrected version of the line. Any line without a number is correct as written and is not to be changed. Make sure that your corrections lead to the indicated output. (You can give the 3 answers in any order.)

```
package problems;

public class Errors {
   int x, y;
   static int z;

   public Errors() {
      x = 1;         // line 1
      y = 2;         // line 2
      z = 3;         // line 3
   }

   public static void fix(Errors e) {
      x = z;         // line 4
   }

   public int fix(Errors e, Errors f) {
      x = e.x;       // line 5
      y = f.y;       // line 6
      return e.z;    // line 7
   }

   public static void main(String args[]) {
      System.out.println(z);       // line 8
      // The output from the last line should be 0
      Errors e = new Errors();     // line 9
      e = fix(e);                  // line 10
      fix(e,  e);                  // line 11
      System.out.println(z);       // line 12
      // The output from the last line should be 3
   }
}
```

Answer:

(a) There is an error in line number: 4

A corrected version of the line is as follows: e.x = z;

(b) There is an error in line number: 10

A corrected version of the line is as follows: fix(e);

(c) There is an error in line number: 11

A corrected version of the line is as follows: e.fix(e, e);