

ID Checked _____

CSC143 Exam I

Name: _____ Date: _____

1. One of your colleagues is trying to create a new class that represents a point in three-dimensional space. Here's the code so far:

```
public class Point3D {
    private double xCoord, yCoord, zCoord;

    public double getX() { return xCoord; }
    public double getY() { return yCoord; }
    public double getZ() { return zCoord; }

    public String toString() { return "("+xCoord+", "+yCoord+
        ", "+zCoord+")"; }

    // test point
    public static void main(String[] args) {
        // create a point and set components
        Point3D pt3d = new Point3D();
        xCoord = Math.sqrt(3);
        yCoord = -12;
        zCoord = 88;
        // print the point
        System.out.println(pt3d);
    }
}
```

This code will not compile. What is the problem? How can it be fixed?
Remember: The compiler doesn't care about style.

2. Give the output for the application that appears on page 5. You may show it in multiple columns.

3. Compare and contrast the types `MouseListener` and `MouseAdapter` in `java.awt.event`. (List similarities and differences.)
4. What is the effect of the following declaration?

```
import java.awt.*;
```
5. Write a Java application that prints “Hello World” to standard output. Include JavaDoc comments.

6. Create an application that displays the shown window:

The window dimension is 200 x 150. There are three visible components in the window: two buttons and one label. The label displays centered, 36-pt, bold Arial text. (Adjusting the label is optional, not required.) Clicking “up” changes the value displayed at the label, increasing it by one. Clicking “dn” changes the value, decreasing it by one. For extra credit, make a click on the label cause the value to revert to 100, the initial value. No need for comments in this code. (With extra-credit, this can be implemented in 50 lines, including “import” and “}” lines.)



Based on the code on page 6, for each statement in the four groups of statements, indicate the output or error generated if the statement were run in the debugger of a Java IDE. Be sure to show the output produced by earlier statements in the group, even if a later one causes an error. If there is no output, write the word “none”. See example on page 6. Assume the debugger is reset before running each group of statements.

7. Group 1.

```
Berry b = new Berry();

b.snap(2.5 < 3);

System.out.println(b.pop(5));
```

8. Group 2.

```
Berry z = new Cherry();

int p = z.pop(5);

z.snap(p = 8);
```

9. Group 3.

```
Apple y = new Berry();

System.out.println("crackle: " + y.crackle(4));

y.snap(4.5);
```

10. Group 4.

```
Cherry n = new Berry();

int r = n.pop(7);

System.out.println(n.crackle(r + 1));
```

Use the following code to answer question 2 on page 1.

```
public class ReadCode {

    private int i, j, k;

    public void a() {
        System.out.println("a()");
        int[] n = {1, 3, 5, 2, 4};
        i = 2;

        System.out.println(n[n[3]]);
        System.out.println(i=n[1]);
        System.out.println(n[i++]);
        System.out.println(i+n[i]);
        System.out.println(i---n[i]);
    }

    public void b() {
        System.out.println("b()");
        double j = 1;
        i = 1;

        while(i < 10) {
            j += i / 3;
            System.out.println(i + " - " + j);
            i += 2;
        }
        System.out.println(i + " - " + j);
    }

    public void c() {
        System.out.println("c()");
        i = j = k = 7;

        for(int i = 2; i < 11; i += 3) {
            j -= 2;
            k += j;
            System.out.println(i + " - " + j + " - " + k);
        }
        System.out.println(i + " - " + j + " - " + k);
    }

    public static void main(String[] args) {
        ReadCode here = new ReadCode();
        here.a();
        here.b();
        here.c();
    }
}
```

Consider the following definitions.

```
public abstract class Apple {
    public Apple() {
        System.out.println("Apple constructor");
    }
    public void snap(boolean i) {
        System.out.println("i = " + i);
    }
    public abstract int crackle(int m);
    public double pop(double x) {
        System.out.println("x = " + x);
        return x * 2;
    }
}
```

```
public class Berry extends Apple {
    public Berry() {
        System.out.println("Berry constructor");
    }
    public void snap(double d) {
        System.out.println("d = " + d);
        System.out.println("pop: " + pop(d));
    }
    public int crackle(int w) {
        System.out.println("w = " + w);
        snap(w);
        return w + 5;
    }
    public int pop(int k) {
        System.out.println("k = " + k);
        return k + 3;
    }
}
```

```
public class Cherry extends Berry {
    public Cherry() {
        System.out.println("Cherry constructor");
    }
    public void snap(int s) {
        System.out.println("s = " + s);
        System.out.println("pop: " + pop(s) + 3);
    }
    public int pop(int r) {
        System.out.println("r = " + r);
        return super.pop(r) - 1;
    }
}
```

Example Group:

```
int x = 7;
```

none

```
System.out.println(3 * x);
```

21

```
System.out.println(5 + 3 / 0);
```

Di vi de by zero error