Review: Quiz 1

January 19, 2022

_____ 1. Which of these Processing statements will cause a circle to be drawn?

A) `ellipse(10, 10, 20, 30);`
B) `ellipse(20, 10, 10, 30);`
C) `ellipse(20, 30, 10, 10);`
D) `ellipse(10, 20, 30, 10);`
E) None of these

- Think about the parameters for the ellipse() function…what has to be true for a circle to be drawn?
- The first two parameters define the center point of the ellipse. The third and fourth parameters define the width and height. For the ellipse to be a circle, these parameters need to be the same.
- The only choice that works is C.

_____ 2. The best description of the rectangle that would be drawn by the following code fragment is which answer? 255 is the color number for white, 153 is the color number for gray, and 0 is the color number for black.

```
stroke(0);
fill(153);
stroke(255);
rect(100, 200, 50, 25);
```

A) a black rectangle with a white border
B) a gray rectangle with a white border
C) a gray rectangle with a black border
D) a white rectangle with a gray border
E) None of these is an accurate description

- Think about what is happening line by line.
  - The first line sets the stroke (border) to black
  - The second line sets the fill to gray
  - The third line sets the stroke (border) to white – hmm! What will that do?
  - The fourth line draws a rectangle with upper left corner at (100, 200), a width of 50 and a height of 25
- We know the rectangle is gray because of the fill color. So is it B or C?
  - Processing will execute each line of code in order; setting the stroke to black first and then changing it to white before drawing the shape will result in a white border. The answer is B.

1. What would be printed by the following Processing code: /* 4 points */

```
int yards = 1;
int feet = 2;
int inches = 10;
print("1 yard, 2 feet, 10 inches = ");
inches = inches + 12*feet + 36*yards;
print(inches);
println(" inches.");
```

- We are declaring and initializing 3 integer variables: yards = 1, feet = 2, and inches = 10
- We are printing out "1 yard, 2 feet, 10 inches = "
- We are setting inches = the *former value* of inches (10) + 12 * the value of feet (2) + 36 * the value of yards (1). The variable inches now has a new value associated with it:

$$10 + 12*2 + 36*1 = 70$$

- We now print the value of inches (70). This will print on the same line!
- We now print out the word " inches." Anything printed after this would be on a new line because we used the println() function.

1 yard, 2 feet, 10 inches = 70 inches.

2. Suppose x has the value 25 and y has the value 49. Give statements that would produce the following output: /* 3 points */

```
25          49
```

- We know we need to use a print function…but which one(s)?
- 25 and 49 are NOT printing on different lines, but there is space between them
- We need to print the variables and the space in between, which can be accomplished multiple ways

1.

```
print(x, “              ”, y);
```

2.

```
print(x);
print(“          ”);
print(y);
```

3.

```
print(x);
print(“      “);
println(y);
```

3. Suppose x has the value 25 and y has the value 49. Give statements that would produce the following output: /* 3 points */

```
25
49
```

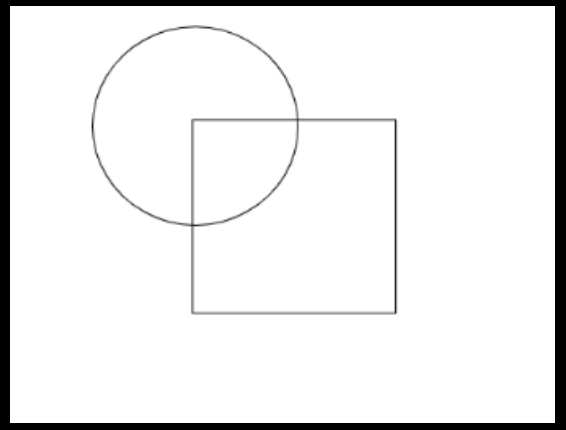
- How is this question different from the previous one?
  - The value of y is printed on a new line
- We must use println() for this to work correctly

```
println(x);
println(y);
```

4. Draw a picture of the shapes that would be drawn by this Processing code: /* 4 points */

```
size(400, 400);
noFill();
rect(100, 100, 100, 100);
ellipse(100, 100, 100, 100);
```

- Let's look line by line and see what's happening:
  - A 400x400 sketch is created
  - We are asking Processing not to fill in the shapes
  - We draw a rectangle with upper left corner at (100, 100) with width=100 and height=100
  - We draw an ellipse centered at (100, 100) with width=100 and height=100 (so it's a circle)
    - Note that the *center*, not the *corner*, of the circle is at the top left of the rectangle!



5. Fill in the blanks in this program so that it converts the given number of pounds into ounces correctly and prints the answer in the Console. /* 4 points */ Hint: there are 16 ounces in a pound.

```
int pounds = 10;
int ounces;

print("10 pounds = ");


_____________________________ ;


print( ounces );


println("_____________________");
```

- What do we still need to do?
  - The first blank comes before `print (ounces);` but we don't have a value for ounces. So we'd better calculate it there. Since there are 16 ounces in a pound, we can say

    `ounces = 16 * pounds;` **OR** `ounces = pounds * 16;`

  - So far, Processing will print "10 pounds = 160". We have the value of the variable ounces, but not the word "ounces," so that must be what goes inside println();

    `println("ounces");`

  - Final version on next slide ⇩

5. Fill in the blanks in this program so that it converts the given number of pounds into ounces correctly and prints the answer in the Console. /* 4 points */ Hint: there are 16 ounces in a pound.

```
int pounds = 10;
int ounces;

print("10 pounds = ");


 ounces  =  pounds * 16  ;


print( ounces );


println("  ounces  ");
```

6. Complete this Processing statement so that it calculates the average of the values stored in variables `mouseX` and `circleCenterX`.. /* 3 points */

```
average = _____ ;
```

- We had a similar question on Lab 1. What do we need to do regarding *operator precedence* (programming lingo for order of operations) so that the average is calculated correctly?
- We need to be sure to use parentheses so the *addition* of mouseX and circleCenterX is done before the *division* by 2.

6. Complete this Processing statement so that it calculates the average of the values stored in variables `mouseX` and `circleCenterX`.. /* 3 points */

```
average = ( mouseX + circleCenterX ) / 2 ;
```