

## **CMPS 5J – Program 3**

**Winter 2018**

**Due: Wednesday January 31 @ 11:59pm**

### **Objective**

The objective of this assignment is for you to make your previous (possibly static) sketch become an animation. If your previous program involved some sort of animation already, then build on it. Your prior program should have had some "interaction" which for most was probably drawing something at the cursor location. For this assignment your program will need to change over time, without any user interaction (although user interaction is also good). This will be done with the use of one or more user defined variables that get modified as the program runs.

In addition you will learn how to create more complex and interesting images, without your program being long and complex. You will do this by making use of two additional standard Processing functions, `translate()` and `rotate()`.

You should also try and include at least one conditional statement in this program.

Finally, you will get further practice with using variables to create an easy to understand, easy to modify program.

### **Program Specification**

You may extend your program 2 drawing or start an entirely new drawing. If you do extend program 2, be sure and read "Giving Credit Where Credit Is Due." (<https://users.soe.ucsc.edu/~dustinadams/CMPS5J/ecommons/credit.htm>) Creating an entirely new drawing is recommended. With the addition of `translate()`, `rotate()`, if-statement, and variables, you should now be able to create more complex and interesting sketches/programs, without having the program itself be complex and confusing.

Your program should have very few "magic numbers." The examples in the first 3 chapters of the text are filled with magic numbers - literals like 119 and 160 (see example 2-1 from the text). These make the program hard to modify and hard to understand. Although the textbook author has not eliminated all magic numbers in the early chapters, he has removed many. For example, the use of `zoogX` and `zoogY` in example 4-8.

Your program must include several examples of variables that control the placement or size of one or more objects so that simply changing the value of the variable moves or resizes the object (be sure and test this - remember the car example from class when I first tried to change the size). At least one of those user defined variables (i.e. not the system variables such as `mouseX`) must change with time so that your static sketch becomes an animation.

Ideally a program should have no magic numbers. For example, again in example 4-8 from the text you could declare two additional variables to eliminate the magic numbers 16 and 32:

```
int eyeWidth = 16;
```

```
int eyeHeight = 32;
```

Or better yet:

```
int eyeWidth = 16;
```

```
int eyeHeight = 2*eyeWidth;
```

Your program must also include at least one if-statement and use `translate()` and `rotate()` to make it more visually appealing.

**What to turn in**

Turn in a pde file called **program3.pde** to Canvas.

**Grading rubric**

- 1.0 Did they submit a .pde file?
- 1.0 Does the program include an opening block that includes a description of the program?
- 1.0 Is the code tidy and well organized?
- 2.0 Does the sketch run without errors when you click play?
- 2.0 Does their program include at least one if-statement?
- 1.0 Does their program include variables?
- 1.0 Does their program include the use of translate?
- 1.0 Does their program include the use of rotate?