Self Assessment
Emilia Emcke

1. **Evaluate how much time was spent learning syntax structure and programming concepts.**

Throughout the semester I felt frustrated because I didn't feel like I was getting a hold of object oriented programming or programming using vectors. It took me a really long time to wrap my mind around the way that works. I made it a priority that I would use this project as a way to understand vectors as well as OOP. I used both in this project! I spent hours watching Shiffman's videos, had many sessions and meetings with the TA, to get some extra assistance, and practiced doing small projects on my own to try and understand the syntax and the structure of object oriented programming. I struggled initially with understanding how the syntax works and how constructors and classes are built. I struggled with finding ways to reuse my code and manipulate what I have already created. I feel like through my research I have come to understand both things pretty well!

2. **Success and frustrations with P5 and Processing**

I feel that I have succeeded in understanding both P5 and processing and knowing the differences in syntax between the two. I do realize that when I use processing for a while, and then switch to P5 I forget certain things. I get frustrated sometimes when I use P5 that my code gets stuck a lot more often. I feel like with P5 my code will run even when something isn't working, and then I get frustrated trying to decipher what the actual issue was.

3. **Compare OOP versus Procedural Programming**

The difference been OOP and Procedural programming is that in procedural programming you use data in your code. Whereas for OOP you use objects, through which you can continuously re-use and recycle some of your code. I haven't quite understood how to do do procedural programming.

4. **Specifically considering your final project: What programming concepts solidified in your final project? What did you learn with reference to programming? Did you have a break through?**

I had a huge break through! I think I had two in this semester, when I began to understand for loops, if statements and how beneficial they can be. I had a second break through recently when I began understanding OOP and vectors! These two latter concepts where really solicited in my final project. I used vectors to determine the movement in the line drawing when the drawings are animated. This gives it a really cool effect that makes the drawing come to life in a way that still makes it look like a drawing, but gives it some dimension.

5. **Were you able to resolve your own bugs? What tricks did you learn? Did you do any debugging?**

I started off with some pretty basic problems: understanding PGraphics, figuring out how to have the user draw without redrawing the image of the user. The hardest hurdle for me was figuring out how to save a group of shapes together into a sort of family and turning that group of shapes into an object that I can reference and manipulate. I also struggled with deleting the group of shapes when looking to draw the next scene.

6. **How do you think you'll move forward with programming? will you keep doing it ? How does this relate to other classes you are either taking or wish to take?**

I want to keep learning about coding and see in what ways I can use it in other areas. I think its been really fun for me to think in terms of code. I notice myself throughout the day observing things and sometimes even thinking in pseudo code (very strange). I'm really glad that I've gotten an insight into the way coding works. I hope that I can get better at it through more classes and also spending **some** time by myself thinking of a project I want to work on in my free time! I'm not sure how unrealistic this is but I think it would be cool to make myself a website one day. I don't know if this is an insane goal but maybe this is something I can start working on over the summer.

Documentation of your final project:

- Take several screen shots of application running. Infact, shoot a short video of the screen. Get good, crisp, well lit, clear shots.

- copy and pasted 2-3 screen shots of code into Sublime that you are most proud of.

```
if (playAnimation && shapeGroup.getChildCount()>0) {
   int i = 0;
   for (PShape shapePart : shapeGroup.getChildren()) {  // Per each group of lines
    if (playAnimation && shapePart.getChildCount()>0) {
      for (PShape shape : shapePart.getChildren()) {  // Per each line
        shape.beginShape();
        shape.strokeWeight((sin((float)(i+weightAnimation)/10) + 1.5) * 5); // adjust strokeweight
        shape.endShape();
        weightAnimation += 0.005;
```

I am specifically proud of this code because it took me such a long time and so many trial and errors to get the syntax of this concept correctly. This took me so much research and also phone calls with my cousin who understood this concept a bit better than I did!

- concept paragraph answering the following: What is your piece about? What's the concept? Is it interactive, responsive or time based?

My project is an interactive narrative. The idea came to me after seeing a picture of a TV show that I used to watch as a kid. I remember thinking that the concept was magical. The main character, Harold, would use his purple crayon to draw whatever he wanted. And this drawing then came to life and became something that he could interact with. I wondered if I could re-create this feeling through my interactive project. Because of this I decided to call my piece draw to life! I wanted to use my code to bring child like drawings to life.