Critically analyze/evaluate how much time was spent learning syntax & structure, programming concepts vs. actually programming, and how does this reflect on the final quality of your end result.

As I expressed in my previous evaluation, I need to dig my hands into code to be able to learn and understand what is happening. So for me, learning syntax, structure, and programming was a mix of reading tons of documentation and trying to implement the code myself, whether in online code sandboxes or directly into my project. At this point in my coding education, I have begun to try to come up and implement complex solutions purely based on accrued knowledge, and the moments where I am able to do so without double-checking syntax online or referring to references are so satisfying! It really does feel like learning a foreign language, there comes a point where you are able to formulate and use your own sentences that get across your point, even if they might not be the most correct sentence ever spoken. Still though, I generally had at least ten tabs of references open. My final project definitely feels like a mix of tried-and-true Stin solutions, such as counters combined with if-statements as well as event listeners—and novel techniques such as increasing the modularity of my functions and utilizing the custom dialect of a library (p5.riso)! I pushed myself very hard with this project and am proud of the quality of my final result!

• ***Comment on your successes and frustrations with Processing and P5.js.***

Gosh, it is somewhat hard to remember Processing now…I remember that getting into the groove with imaging canvas coordinates to make complex skills was a very hard skill to learn. But now it feels pretty comfortable to me! In my final for my other coding class, I found it was easier to write custom shapes by creating custom functions than to design an svg in an art editor and importing it into my project…which is pretty shocking for me, the lifelong illustrator. But manipulating the code of a custom shape function is so much easier than manipulating the complicated code of an svg designed in a GUI editor.

As for p5.js, I really like it, and I am excited to continue exploring all of the insane libraries people have created! I find it so beautiful how open-source so much of the coding community is. Although, I have found that some of the documentation of some p5 libraries to be lacking and you have to be able to read the libraries themselves decently well to understand what is going on and how to utilize them to solve my specific problems. Fortunately, p5.riso's documentation and example projects were fantastic, but still to use the information stored in the RISOCOLORS array, I naturally had to look in the library itself, at the array and see how the information was stored (in rgb, under the property 'color') to be able to fetch it in my code.

• ***Compare and contrast OOP versus Procedural Programming. How are they similiar? How are they different?*** I am looking for you to explain this in your own words, what these concepts are. I am NOT looking for you to tell me how you used these in your project.

I think of procedural programming as modular programming–creating a series of separate variables and custom functions which work together, typically in fairly linear sequences, to achieve a desired set of steps. Sort of like gears fitted together in a machine. Object-oriented programming on the other hand is less linear. When I think of OOP I

immediately think of both objects and classes, and see their use case for when complicated variables need to persist over longer periods of time. These variables share some information and are often similar but not identical, and also need to be able to store new information separately and react to things on their own.

• ***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?***

Something I was really happy to be able to do was combine my knowledge of HTML/CSS with p5.js in this project. I have been taking Drawing on the Web alongside Creative Coding this semester, so I knew I wanted to combine techniques learned from both courses in my final projects. I want to continue down the path of learning web dev for both 'practical' purposes as well as for experimental and innovative visual/written storytelling. So it was great to be able to apply p5.js for more standalone web experiences! I felt like all the techniques I had learned were coming together to make something that I was truly passionate about, which was very fulfilling.

• ***Specifically considering your final project: Were you able to resolve your own bugs? What tricks did you learn in the process to help? Did you do any debugging?***

I often had to turn to references to solve bugs, but I definitely solved quite a few myself, and more often than not debugging was a series of steps that mixed both whacking my head against the code as well as crawling down documentation for answers. For example, I was so confused—when I made the reverse page turn animation—why the technique I had used for the forward page turn wasn't working. Then I realized that I had used element.remove to make the pages disappear. And so, I searched online "what is the opposite of element.remove", and subsequently tried body.appendChild(element) to make the elements exist again. This did not work as expected and I switched to the technique I was already using a ton throughout my code–element.style.display–to control the appearance and disappearance of all the visual elements throughout my project. One trick I learned very well in this debugging process is to check how old forum posts are, a lot has changed in the past ten years.

• What was your intended milestone? Did you make it? Did another one pop up? Tell me about this. Tell me how you resolved it.

I consider the working prototype I presented to the class to be my milestone. It was really transformative for me to be able to make an application that changes raster images, that changes my own illustrations, dynamically. Illustration is one of my truest passions in life, and after decades of creating static images, I am so excited about the potential of this! Being able to add interaction and deeper ways for the user to engage with my drawings is so profound to me! It still blows my mind to see the different color layers change the entire image–the way colors multiply is really amazing. Also the aesthetic of pixel dithering is so beautiful, and I was so excited to play with that for the first time. Now I want to throw all my illustrations into this code to see how they transform.

• What are you *most* proud of, with reference to your final project?

What I just wrote about above is what I am most proud of, I have been slowly building up to being able to create my own innovative web-based novel that combines both written and visual art, and I am excited about maximalism on the web and breaking out of the legacy of Bauhaus and corporate standards of design (even if understanding universal design is important and incredibly useful), and I feel like I have made some huge breakthroughs to eventually realizing that goal. p5.js is an amazing tool that I will be sure to use in the future and I am grateful to have learned it this semester!

As for the technical achievement I am most proud of–probably making the color dropdown menus. Going into the p5.riso library and figuring out how to iterate through that array to populate a dropdown menu, and then to make the color fill match all the color names, and then have clicking on that color name actually change the correct layer? I am very proud to have accomplished that!

• ***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 programming in tandem with design, and programming for realizing artistic visions! I am planning on taking Front-End Web next semester! Perhaps I should learn a little back-end at some point, but front-end is certainly my area of interest.