Reflection on the Semester

1. How much time was spent learning syntax/structure, programming concepts, versus programming?
Looking back on this semester, I've realized how much time I spent learning programming fundamentals like syntax, structure, and key concepts versus actually writing code. At first, I felt completely overwhelmed trying to figure out how everything fit together. For example, truly understanding how different parts of the program, like arrays or loops, interacted with one another took time to grasp. Honestly, it was a slow start. What helped me get over the initial hurdles was watching YouTube videos. Seeing programming concepts broken down step-by-step and applied to real examples made things so much easier to understand. Following along and seeing immediate results on my screen gave me the confidence to keep going.

2. Successes and frustrations with Processing and P5.js
P5.js was both exciting and frustrating. Its visual capabilities made it fun to create interactive projects, but debugging issues and figuring out how to implement certain features wasn't always straightforward. Sometimes, the documentation wasn't as clear as I hoped, and I had to rely on external tutorials or trial and error to get things working. A success was when I finally got my animations to flow smoothly and work seamlessly with user input. The frustration came from small glitches—like alignment issues or unpredictable behaviors—which took a lot of time to fix.

3. Comparing Object-Oriented Programming (OOP) to Procedural Programming
Procedural programming is like following a detailed step-by-step recipe. It's straightforward and works well for smaller, linear tasks. In contrast, Object-Oriented Programming (OOP) focuses on creating objects that combine data and behaviors. While procedural programming is simple to start with, OOP is more scalable and modular for larger projects. Initially, I found OOP confusing because it felt abstract, but as I used it more, I saw how grouping related data and functionality into objects made my code more organized and easier to manage.

4. Programming concepts solidified in the final project
This project helped solidify key concepts I struggled with earlier in the semester. Arrays and loops, which seemed intimidating at first, became second nature by the time I was managing hundreds of particles. I also became much more comfortable using P5.js-specific functions like dist() for collision detection and push()/pop() for isolating transformations. Debugging was also a big part of the learning process—I finally understood how to isolate issues and test smaller parts of my code to identify errors efficiently.

5. Debugging and resolving bugs
Debugging was one of the most time-consuming parts of the project but also the most educational. When things didn't work—like animations glitching or variables not resetting properly—I relied heavily on tools like console.log() and broke down my code into smaller testable pieces. Watching tutorials on debugging techniques helped a lot, too. I learned to embrace debugging as a normal part of the process, and there's something incredibly satisfying about finally solving a problem that seemed impossible at first.

6. Milestones and unexpected challenges
My original milestone was to create a visually engaging and interactive experience with an animated circle and functional Morse code gameplay. While I achieved this, one unexpected challenge was refining the cloth simulation. It needed to stand out visually but not distract from the main gameplay. Balancing its placement and behavior took more trial and error than I anticipated. Adjustments to its spacing and interactions helped me achieve the polished look I was aiming for.

7. What are you most proud of in your final project?
I'm most proud of how cohesive the final project feels. The animated circle, the Morse code mechanics, and the cloth simulation all tie together thematically and visually. Seeing everything come together after so many revisions and debugging sessions was incredibly rewarding.

8. How will you move forward with programming?
Moving forward, I'm excited to keep building on what I've learned this semester. I've signed up for a coding course over winter break outside of NYU to continue improving my skills. Next semester, I plan to take more programming classes as part of my coursework. Programming is becoming a bigger part of my education, and I'm excited to explore how I can apply it to problem-solving, design, and future projects. This semester showed me how challenging but rewarding programming can be, and I'm excited to see where it takes me next.