Post-Reflection
Laura Slabbert
CART 253
Section A

At the beginning of this class I had no idea if I was going to be able to complete it or not, not because of a prophetic anticipation of the difficulties I would have in November and December, but because the last two times I had taken a class that involved numbers and logic systems, I had to drop them. I'm very comfortable with seeing myself as someone who just doesn't get this kind of stuff. Having finished the class now, though, a lot of the mystery has been lifted from the practice of programming. Now, when I navigate websites or play video games, I can begin to picture a vague outline of what the skeleton of the program I'm using might look like. The most obvious change was how much easier things became, without me even realizing it. I noticed this through my apprehension at having to add an "alive=true/false" element to variation #3, because I was remembering the struggle I had with it when making the art jam, and then being surprised at how easy it was to do the second time around. That feeling helped me build some much-needed confidence, and will certainly be a point of reference for any future programming struggles I encounter.

This class has impacted my creative practice by shedding new light on many of the preexisting game ideas I have. Whereas before, I would picture the visuals, some level design, and narrative beats, now I can begin to piece together how I might actually start building these ideas. There's a lot I have yet to fully wrap my head around, though, as loops and arrays are still a bit of a mystery to me. However, instead of beating myself up for not yet getting them, as I might have done at the beginning of the class, I now have enough confidence to allow myself to try and fail as many times as it takes for those pieces to finally click into place.

My previous perspective of coding as a creative outlet has also greatly shifted, as I now understand how one can draft code and rough out an idea before worrying about missing brackets and commas. The “Making” module was quite a turning point for me in understanding how the very beginning of the process worked. Now I can appreciate coding not only as something that allows for rough drafts, but also as something to inspire more creativity.

In my communications studies classes, we were taught Marshall McLuhan’s theory that “the medium is the message”, and I can now see how that is true for coding. I think the focus on simple shapes used in the programs we built emphasized this, because when the graphics are simple and there is no sound, the program is what does the heavy lifting for the aforementioned “message”. An example of the medium being the message was the change made in my third variation, where instead of having two separate functions to move the car and the train, in order to reduce repetitive code, both of their commands were written in the same function. This is a mechanically efficient choice, but also futhers the narrative that the two are connected and have no choice but to crash. Artistic works are often at the mercy of their audiences, but through the rules created in the programming, an artist has a level of control over what their audience can and can’t do, and the audience has the potential to interact with the project in a way the artist did not anticipate, creating an element of collaboration between the two that is not an inherent part of other mediums. This inspires me to want to make more flexible programs one day, that allow the audience to push the boundaries of any future banana-torture themed works I may create.

As for my future as a creative coder, I’m not yet at a point where I would tell someone I’m a programmer, or “an artist who codes”, but I absolutely feel that I am on my way there. My understanding of creative code is just how similar it is to other arts, in the sense that you don’t need to reinvent the wheel, but you can put your own spin on it. For example, you can look at

how someone else made a for loop work for their specific project, and adapt it to your own, and from that point you can patch together references until you've built the specific and unique bit of programming your project needs. Before this class, I had mentally limited myself to only ever being able to make 2D pixel games, using something like GameMaker, and although those games take an immense amount of effort and have a lot of potential, they seem easier and therefore matched my assumption of what I was capable of. Over the time I've spent in this class, however, so much of programming has been de-mystified, and I've begun to think about games in the third dimension, and am getting really excited about the stories I'd like to tell. To be specific, I'd love to make something that uses real world bits and pieces that are 3D-scanned into the digital world, like this, as it creates the potential to combine my interest in toys and highly physical mediums and my interest in video games, when I used to think I'd end up pursuing those interests separately. This class has also equipped me to better participate in game jams, which I had done three times before. While I don't feel that I'm at a point where I could program for a game jam, I now have a much better understanding of what my programming teammates will be up to and what they may need from me and any visual assets I may develop for the game. I think my next step as a programmer is to start looking at learning a game engine, and seeing where I can go from there.