John Barnyk

Mr. Richards

Graduation Project

12 December 2021

Log Check #1

There was a variety of tasks committed for this log check period. Most were either set up tasks or learning tasks. I feel I am currently on a good pace, with the expectation of a rough prototype for the next log check. I am proud of what I have been able to accomplish this far.

First, I knew I wanted to use GitHub to hold and distribute my project. I had learned about this site after years of programming and video games. GitHub is a site that host “repositories”, or a project, often relating to programming. To upload, edit, or do anything at all to said files, one must use an API called “Git”, which is not unique to GitHub. To actually edit files, on must “clone” the files already in the repository, or repo for short, to his computer. Then one can edit the local versions of those files. Once those edits are made, one must “stage” a “commit”. A commit is simply a group of changes, whether they be edits, deletes, or creations. Staging means simply collecting these changes into a commit. Once a commit is staged, you can then you can push it, or put the changes in place on the main files.

This process may appear to be more work than its worth, but it is better in the long run as it catalogs all changes ever made. It also allows me to post it online publicly to allow anyone else to use my code.

I also learned the Ursina game engine. This is a framework for producing games using python. It is akin to a code only Unity or Unreal Engine. By using the engine, I can focus on writing rendering code, the purpose of my project, without having to code a myriad of other minor things to make work. I would need to write a world coordinate system, an object handler system, a camera system, a system to read GLSL code (shader code) and interpret it, and finally a window system. All these things are handled by Ursina. This way I do not need to waste my time on reinventing the mighty wheel.

One thing I did need to write was a system to procedurally create a mesh from the list describing the location of where the blocks are. I needed this so I could efficiently make a mesh of the world. While it does not make much difference right now, down the line it could become be quite problematic when scaled up. To do this, I first generated all vertices that contact each block. I then remove every vertex that is redundant or is inside blocks. I then create a list of all the vertices. Then finally I create a list of triangles using those triangles.

The final thing I did was start on my documentation in the form of my “Holy Hand Docs”. Here I have been documenting different types of information from how python works, how Ursina works, and even how my program functions. This makes it much easier for others to understand my code should they need to.

During this next log check, I will be able to begin writing the actual shader script, as well as doing all the background math behind it. From there, it will be a matter of applying it in an efficient manner.

Word Count: 576

Mentor’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_

Mentor’s Comment: