Roy Romero

June 19, 2022

CS-330

Final Project Reflection

To start with this was probably the most difficulty I’ve had with a class. My brain isn’t creative and doesn’t think in the 3rd dimension. So having to re-create a scene and use new software I’ve never been exposed to before was challenging. Speaking on that matter since this was my first exposure to OpenGL, I knew that no amount of studying, and training would allow me to make hyper realistic images. So, that’s where I came across my first design choice which was to recreate my scene using 16-bit/Minecraft textures. I did this so my scene wouldn’t look “gaudy” for lack of a better term. I feel like utilizing those 16-bit textures gave my scene a more endearing look. The reason I chose my scene was because I saw various primitive shapes in it. I knew that I was most likely going to have trouble with any complex shapes, so I chose a scene in my home that had some simplicity in it. Secondary reason I chose my scene is that I really like my obsidian crystals so being able to recreate them was nice. To program for the required functionality, I just created functions and code that could be reused for every shape I wanted to make. The user can navigate my scene utilizing a combination of the keyboard and the mouse. For the keyboard press, W for forward, S for backwards, A for left direction, D for right direction, Q for upwards, E for downwards and P to switch camera view between orthographic perspective and regular perspective. To increase the camera speed, scroll up on the mouse wheel and to set the speed back to normal scroll the mouse wheel back down. If you hold down the left mouse button you can look around using the camera. To set up control of my camera I created an entire class in which I put my code into. In my camera files I utilized GLFW get key functions to read user input for pressing the keys. For the speed increase/decrease of my mouse wheel I used a mouse call back to read user input. I developed custom functions to create an object, this I can constantly re-use these functions throughout my project. I created a VAO, VBO, and EBO class which essentially reads my vertices and indices that put in and then recreates a shape accordingly. I also customized my function so I could put RGB, texture, and light coordinate values in my vertices array. Creating these custom functions and separate classes really helped me save time creating this project.