Jarrale Butts

April 23, 2023

Southern New Hampshire University (CS-330)

Design Decisions

When developing my 3D scene in OpenGL, I made a deliberate and intentional choice to create something that was both appealing and what I thought was going to be fun. These choices were based on the goal of creating a realistic and visually appealing scene that effectively represented my Mac studio, black panther plushy, Samsung monitor, and my armillary. In addition, I considered the functionality required for the scene to allow for user navigation. One of the main objects I chose for the scene was a cube shape to represent my Mac studio. I chose this object because it is a primitive shape. Additionally, the cube shape allowed me to easily add texture mapping and lighting effects to make the object look more realistic. For the monitor and screen, I used two 3D rectangles. These shapes allowed me to create a more realistic representation of a monitor and screen, as they are common shapes for computer displays. To represent the black panther plushy, I used six spheres and two pyramids. These objects allowed me to create a 3D representation of the plushy that was both visually appealing and recognizable. Lastly, I was able to create the armillary using two compressed spheres at the bottom, a pyramid in the middle, and a sphere on top. I struggled throughout this project, especially in making the sphere and the torus, which I was never able to create.

To enable user navigation in the scene, I implemented several input devices. The WASD keys allow the user to move the camera forward, backward, left, and right. The QE keys allow the user to control upward and downward movement. The mouse cursor enables the user to change the orientation of the camera so that it can look up and down or right and left. Finally, the mouse scroll allows the user to zoom in and out of the scene. I set up the virtual camera to respond to these input devices using OpenGL’s built-in functions and matrix transformations.

Overall, the development choices I made for my 3D scene were deliberate and aimed at creating a realistic and visually appealing scene. The functionality required wasn't completed in the end as I couldn't figure out how to allow panning and orbiting smoothly without resetting the scene with the "F" key.

To enhance the modularity and organization of my code, I adopted a deliberate approach to grouping similar code. Specifically, I placed related code segments adjacent to each other, which made it easier to identify and work with related code. In addition, I ensured that there was ample spacing between blocks of code to prevent them from running into each other, which could lead to syntax errors. Throughout my code, I identified instances of reusable code that could be leveraged to accelerate the process of creating new objects. One excellent example of this approach was during the draw function within the while loop of my code. By copying and pasting the draw function of similar objects, I could quickly create a new object. Following this, I would define a separate modelMatrix and use translation, rotation, and scaling operations to adjust the object's position, orientation, and size, respectively. By adopting these modular and reusable code strategies, I was able to simplify the process of creating new objects in my OpenGL scene, while also ensuring that the code remained clean and organized.