Robert Lowrey

CS-330

October 14th, 2023

Project One

      For my project I chose to do a charging cube, a Teflon tape roll, a massage roller ball, and a black desk mat. I chose the charging cube object because it is made up of many triangles in order to make the cube. This cube is white with some shadow on the lower vertices to reflect the lighting of the 2D image I submitted in module two. The Teflon tape roll is made up of two cylinders which is comprised of many triangles to create the circular aspect of each layer in the object. The massage roller ball is made up of two cylinders each with different colors (green and white) and a black center to account for the rolling ball. With not being able to see any part of the program once it's ran, it was very hard to complete this project. I was doing everything blind, so I really hope it came out as anticipated.

      A user can navigate the camera through the scene by either the mouse or keyboard input. The mouse can move the camera and change the direction that the camera is looking; However, the user cannot look past 89 degrees in the up and down direction. The mouse input can also control the movement speed of the camera with the mouse scroller. If you scroll up on the mouse, the camera will move faster around the scene and if you scroll down, the camera will move slower around the scene. Apart from the mouse input we can also navigate around the world scene by the W, A, S, D, Q, E, and P keys. The W key is used for moving positive (forward) along the z-axis. The A key is used to move negatively (left) on the x-axis. The S key is used to move negatively (backwards) on the z-axis. The D key is used to move in the positive direction (right) on the x-axis. If the user chose to input Q on the keyboard, it will move the camera in the positive direction (upwards) on the y-axis. The E key will move the camera in the negative direction (downwards) on the y-axis. Lastly, if the user inputs the letter P on the keyboard it will change the view perspective from orthographic (2D) to perspective (3D), and then the user will just hit the P key again to change it back to the previous view.

      One custom function in my program is for the light one source vertex shader. This will help keep track of one light source while the second light has a different function to keep track of the color. This function will house the light with the blue colored aspect. More custom functions that I used are for the different inputs of the system. One function is for the mouse input and the other for the keyboard input. Another function that I made was the texturing function that allowed two objects to be textured as per the specification document that was given to us. These are just some of custom functions that I used in order to keep my project modularized and organized.