Melanie Probst

CS-330 Project Two Reflection

In my scene, I have four Pusheen the Cat figures. For each figure, excluding the one situated on its side, I reused the same set of vertices that I created myself. I basically created a series of trapezoids and rectangles using the vertices in OpenGL to create the shape of the cat. I also made two pyramids for the ears. In order to apply the face texture to the figures, I singled out the top front trapezoid from the rest of the figure’s shapes. Then, I separately constructed the faces and translated them into the correct spot to complete the figure. One Pusheen is on a podium, holding balloons. I used a cylinder for the podium and cylinders for the balloon strings. I used spheres for the balloons. I applied colored textures to these shapes. The remainder of the Pusheen figures are sitting on the ground. The figure to the left of the “Balloon Pusheen” is holding a donut. I created the donut with two torus shapes. I decided to use two torus shapes so I could apply a golden donut texture and a pink icing texture. I couldn’t find a texture that had star sprinkles like the donut in the scene, so I created a star shaped plane and made the sprinkles myself. I applied color textures to each star and laid them against the donut shape. The smallest figure in the middle is holding an ice cream cone. To recreate the ice cream cone, I created a white sphere for the ice cream and a cone shape using vertices. I applied a waffle cone texture to my cone. For the Pusheen on the right, the one that is laying down, I took the code for the other three figures and modified it to be longer horizontally. I added two extra quadrilaterals in order to preserve the shape of the face, but simply modified specific vertices to be farther to the right.

The user can navigate my scene by using WASD keys to move left, right, forwards and backwards. The Q and E keys allow the user to navigate up and down. The user can control the direction in which the camera looks by using their mouse. The user is able to view the scene in either a perspective view or an orthographic view.

I have a few custom functions (setCoords and createObject) that allow me to create a torus. This is useful because it allows me to create as many torus shapes as I need. I have custom functions that allow me to pass the shaders I use in my program in order to implement the functionality required to allow the user to switch between orthographic and perspective views. There is a perspective and orthographic function for each shader. Finally, there is a function that allows me to load the textures I use in the program. The function can be called for each texture I use and it saves space in the code, which improves readability.