Final Project Reflection

The scenery that I have selected for my final project in an OpenGL environment is a basketball court. When first designing this court, I intended to make it a simple concrete court with a white backboard and net, a gray pole and an orange rim. However, once I learned how to use textures correctly and add my own, this very quickly changed. I realized the potential of OpenGL and how well 3D textured objects can be displayed in it. I have been watching the show “House of the Dragon” on HBO lately, and this sort of became my theme for the court.

First, we had to make the ground plane. This is a standard procedure for basically anything in OpenGL and was used for my court. Next, I used a long cylinder to create the pole for the basketball hoop and positioned it towards the back side of the court as a real one would. After this, I made the backboard with a box. I used a box instead of another ground plane, as I felt it would be easier to get the correct thickness of the backboard. Then, I created a torus and positioned it on the front, lower side of the backboard to act as a rim. Lastly, I tried to create another torus to act as a net under the rim, but doing so gave many issues with the texture differentials between the two tori. Therefore, I had to use a box as the net, but it looked better than I thought it would in the end.

To start, I used the template of what we have been using in the class. I started with the maincode.cpp and created functionalities and adjustment options regarding lighting, shader handling, and the objects in the scene. From there, the scenemanger.cpp was used for the texture loading, creation and rendering of the objects, and buffers/arrays. The scenemanager.h class was used for setting up and creating the functions used in the scenemanager.cpp. Lastly, the viewmanager.cpp and .h were used to establish the standard viewpoint and projection adjustments.

Users can freely roam around this basketball hoop using projection matrices. It can serve as a full 3D environment allowing for further expansion of it. A user can move around the environment by using the standard “A,W, D, S” functionalities for horizontal movement. Vertical movement can be performed by using the “Q, E” keys to go up and down. The speed of the user can be adjusted also, simply by moving the scroll wheel back and forth. This environment acts and serves as a great baseline 3D environment for a game. This is also seen with the textures that I chose for each object. Since the theme is related to dragons I made the court a black, scaly look to resemble their hide. For the pole, I used a stainless-steel texture. For the backboard I chose to do a very orange-sky look as these skies are often seen in environments in sci-fi. The rim also matched the orange as well, blending in well with the backboard. Lastly, for the net I added a texture that resembled a real basketball net’s appearance. Doing all of this really brought the scene together and made it pretty good.

The custom functions that I used in my code are well-written and easily adjustable. If another object was to be added, simply create the functions in scenemanager.h, apply the creation, rendering, and texturing of it in the scenemanger.cpp, and lastly create the object transformations and adjustments in the maincode.cpp. Camera and projection setting can also be found in the main code. Creating the project like this allows for easy modulations and changes. This makes it reusable for something such as a small video game. The developers could simply build from here.

Things that could be added could include a skybox for the surrounding environment, another hoop on the opposing side or changes in court size along with it, as well as people and a basketball to play. After this, changes in the maincode.cpp, camera.cpp, and some small ones in the scenemanagers could possibly create a functioning mini basketball game relatively quick. This is what I aimed for when creating this, as I will probably continue to work on this project further than this class. I really have enjoyed creating this, even though it has also taken a lot of time and error. However, that’s how the best things usually are.