Joy Kalinowsky

CS 330

7-1 Reflection

I decided to use simple shapes to represent the objects in my image so I could recreate the scene with just beginner level 3D programming skills. I decided to choose a sphere to represent the branches and flowers of the cherry blossom tree on the right to recreate the general shape of its flowering branches. I also chose a sphere to represent the hill on the right that the cherry blossom tree is located on to recreate its downward slope. I decided to use a cylinder to represent the trunk of the cherry blossom on the right up to the point where its obscured by cherry blossoms to get as close to the shape of a real tree trunk as possible. I decided to choose tori to represent the line of cherry trees on the left side of the image, the river in the center of the image, and the river bank on the left side of the image. I used this shape in particular to represent all these objects because, in the original image, these objects appear to "wrap" around the foreground. I believed that by using the torus shape to represent these objects, I could recreate this wrap-around effect of the cherry blossom trees, the river, and the river bank in my 3D scene.

I set up the controls for my virtual camera to enable the users to control the virtual camera for my 3D scene and to navigate in a variety of directions. The user is able to move forwards with the W key, backwards with the S key, to the left with the A key, and to the right with the D key. They are also able to move upwards with the Q key and downwards with the E key. They can use the P key to switch to the viewport view and the O key to switch to the orthographic view. They can move the mouse to look around the scene. The supported input devices that users can perform these key presses and mouse movements on are a keyboard and a mouse/touchpad.

There are several custom functions in my program that I am using to make my code more modular and organized. The RenderScene function is used to render the 3D shapes of the environment. For each shape, there is a section of this function dedicated to rendering it. All of the 3D rendering sections are constructed in the same way, just with slight variations in terms of what shapes are rendered, the color/texture of these shapes, the position of these shapes, and the size of these shapes. This makes the 3D shape rendering "base" section easily reusable with certain variables changed. There is also a custom function named SetupSceneLights that is dedicated to rendering the lighting of the environment. The custom function LoadSceneTextures is used for loading and binding the textures of the objects. Finally, there is the PrepareScene function, which loads the shape meshes and textures used in the program into memory.