Module 7: Final Project

# **Comp Graphics and Visualization**

Version 1.2

Ronny Z. Valtonen

UI/UX Design and Development



For my final project I decided to render the above image in 3D. I used a tapered cylinder to re-create the trees scattered around the scene, and using a sphere at the top calculated by the half the height of the tree multiplied by the radius of the sphere, it applies the sphere at the correct height to mimic the smooth top of the trees. This solution is not perfect, however, with the limited number of triangles, this seems like the best solution. I used grass/shrub/leaf textures to mimic the look of the hedges and trees as well using a gravel texture to recreate the look of the walking path. I used a couple directional lights to attempt to create the lighting seen in the image, and I did my best to implement shadow mapping, but that did not go very well so I had to scrap the idea after 4 hours of that. Shadow mapping as referenced in LearnOpenGL would have allowed the creation of small shadows due to the trees and bushes blocking the light source as lighting doesn’t bend.

Within this 3D scene, the user can operate the view by using the WASD keyboard keys to move around as well as Q and E to move the camera up and down. Using the mouse the user can change the view point of the camera and using the scroll wheel can change the speed at which the camera moves. Finally, there is an option to change from perspective view to orthographic view using P and O keys respectively.

The user is also able to connect a Xbox controller, the MainCode will detect if one is present using GLFW and poll the states of each button and joystick. The right joystick controls the location of the camera, the right joystick controls the view direction, L2 and R2 controls the height of the camera. The B button acts as the escape key.

Next I also change the phong lighting to blinn-phong as it reproduces specular reflections much better.

Lastly, to make my code more modular, I had to create the same trees over and over again. Instead of copy pasting the code, I created a helper function that calculates where the sphere needs to be and the size of the tree. So I could create another tree with a different height: the sphere and the tapered cylinder will cohesively be placed correctly instead of manually placing both objects over and over again. Using a dictionary to read the location of each tree, the helper function calculates and places them all accordingly.

References

*LearnOpenGL - Shadow Mapping*. (n.d.). https://learnopengl.com/Advanced-Lighting/Shadows/Shadow-Mapping

*Sonar Systems. (2015, December 15). OpenGL Tutorial 19 - GLFW Joystick/Gamepad Input [Video]. YouTube.* [*https://www.youtube.com/watch?v=mx0IUbxAQ34*](https://www.youtube.com/watch?v=mx0IUbxAQ34)

*LearnOpenGL - CubeMaps. (n.d.). https://learnopengl.com/Advanced-OpenGL/Cubemaps*

*LearnOpenGL - Advanced Lighting. (n.d.). https://learnopengl.com/Advanced-Lighting/Advanced-Lighting*