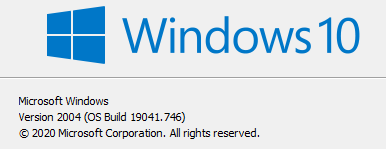
COMP 3520 Final Project

Joshua Gehl

105025519

Mar 10 / 2021

**OS Version and VS Version**: 



**Libraries**: opengl32.lib, glew32s.lib, glfw3.lib, Assimp(installed through NuGet)

**Header Libraries:** glm, stb\_image.h

I could not compile and link using the command line because I wasn’t able to get the MinGW version of the glew32 library to build correctly for my system so I used VS2019 instead because it was a little easier to set up.

**Instructions to run in VS2019**:

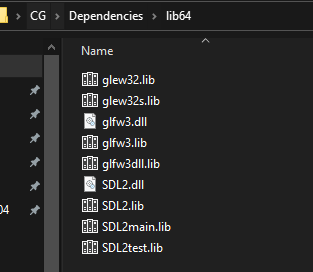
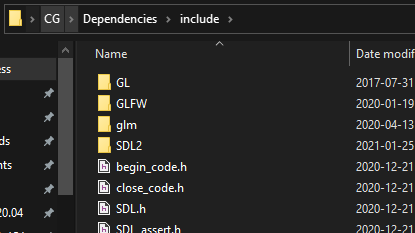
1. Create new Empty C++ Project
2. RIght click Source Files on the right side SolutionsExplorer
3. Add main.cpp to it
4. Make sure the images for the textures are in the folder on your machine as main.cpp (you don’t have to add them to VS, they just have to be in the same folder as main)
5. On top next to ‘Local Windows Debugger’, if it says x86 or x32, change it to x64
6. On the SolutionsExplorer on the right side, right click the project name
7. Select Properties -> VC++ Directories
8. Add the paths to the include/ and lib64/ to ‘Include Directories’ and ‘Library Directories’. Picture below of the contents of these folders
9. The next setting is under Linker -> Input
10. Add this to the end of Additional Dependencies:

;opengl32.lib;glew32s.lib;glfw3.lib;

1. Click Tools > NuGet Package Manager > Manage Packages For This Solution
2. Install the package: Assimp\_native\_4.1\_v142
3. Run program by clicking ‘Local Windows Debugger’

**If you cannot run my program, please contact me and I can show it to you over screen share.**

**Pictures of include and lib64 directories**



**Brief Explanation**

My project is split into multiple classes, I have classes for Window, Shader, Texture, Camera, Mesh, LightMest, Light, and ClockMesh. I explained each of these individually in the comments of my program. I create Mesh objects and call their render() function to draw them to the scene. I have a floor, ceiling, walls, table, chairs, clock, and 3 lights, as well as some other objects in the cardboard box on the table.

**Controls**

|  |  |
| --- | --- |
| **MOUSE** | Look around |
| **ESC** | Toggle pause / unpause |
| **1, 2, 3** | Toggle light sources |
| **4, 5, 6, 7** | Change Camera Position |
| **ENTER** | Unlock Camera Movement |
| **W A S D SPACE LSHIFT** | Camera Movement |

**Sample Output**

I made a YouTube video for my sample output with an in-depth explanation; I understand the prof wanted 5 minutes tops but I couldn’t fit the entire explanation in 5 minutes:

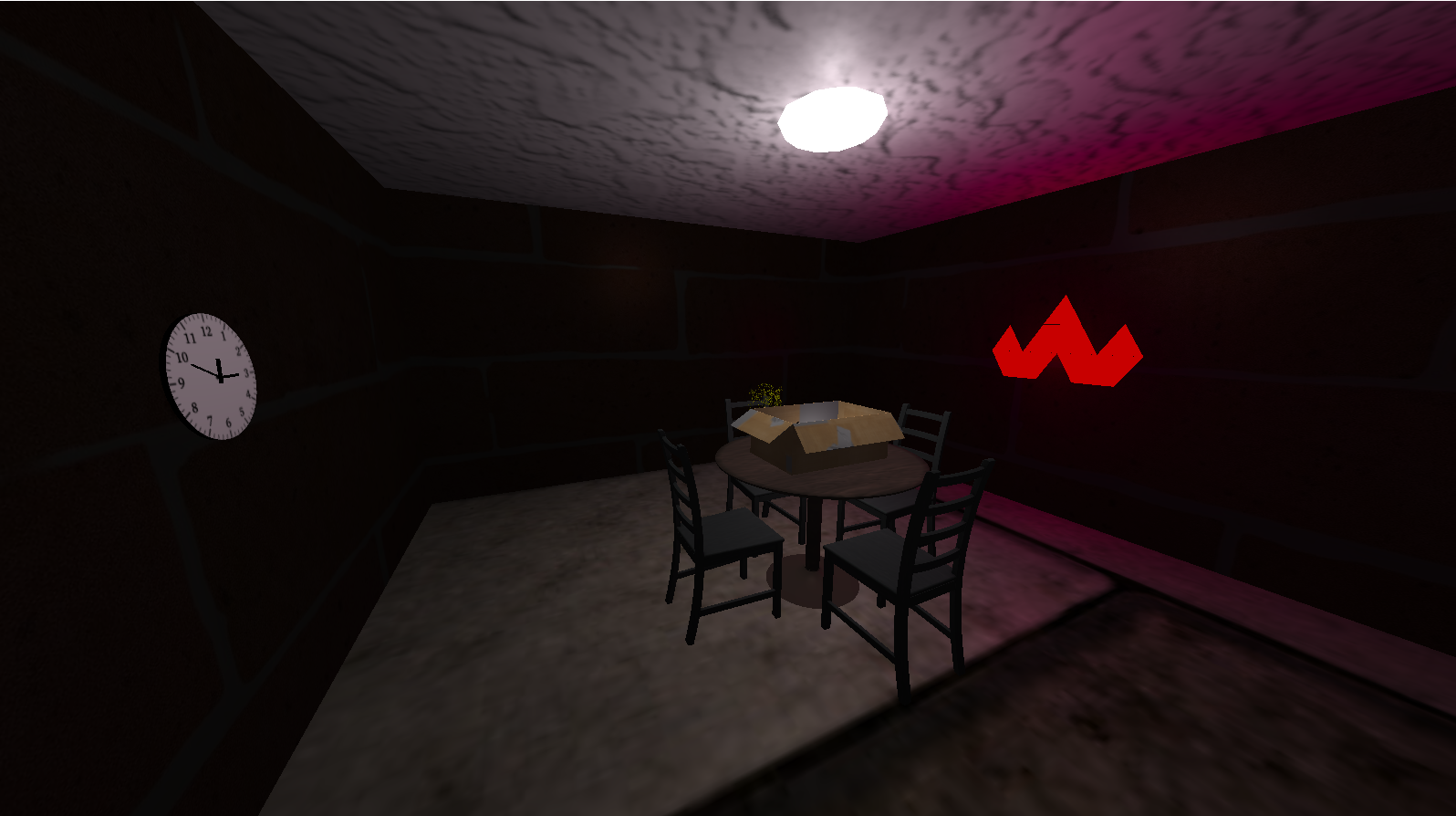
<https://www.youtube.com/watch?v=4dJJfTaGy0I>

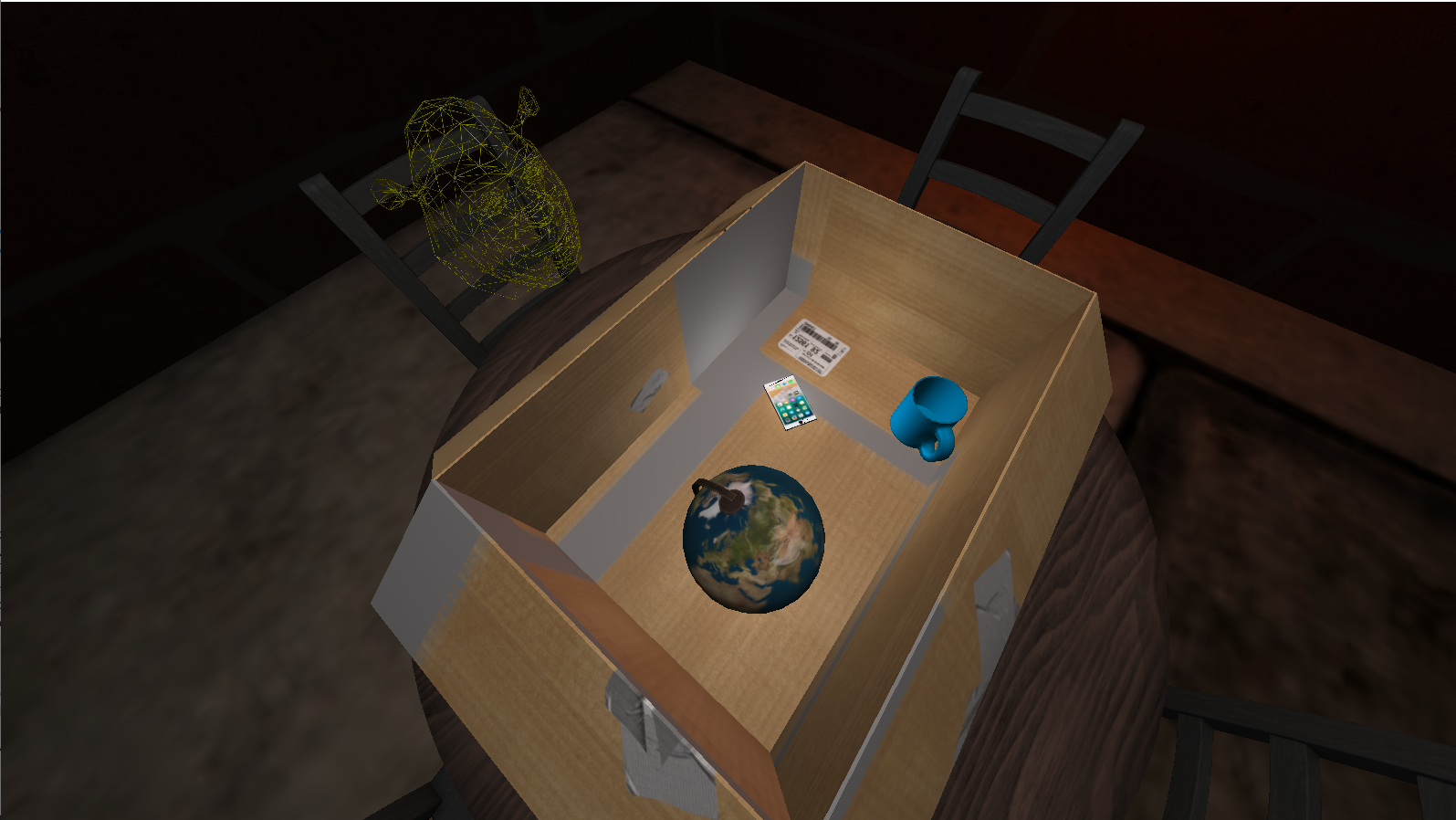
If you wish to see just the demo of the project, here is a link timestamped to when the demo starts:

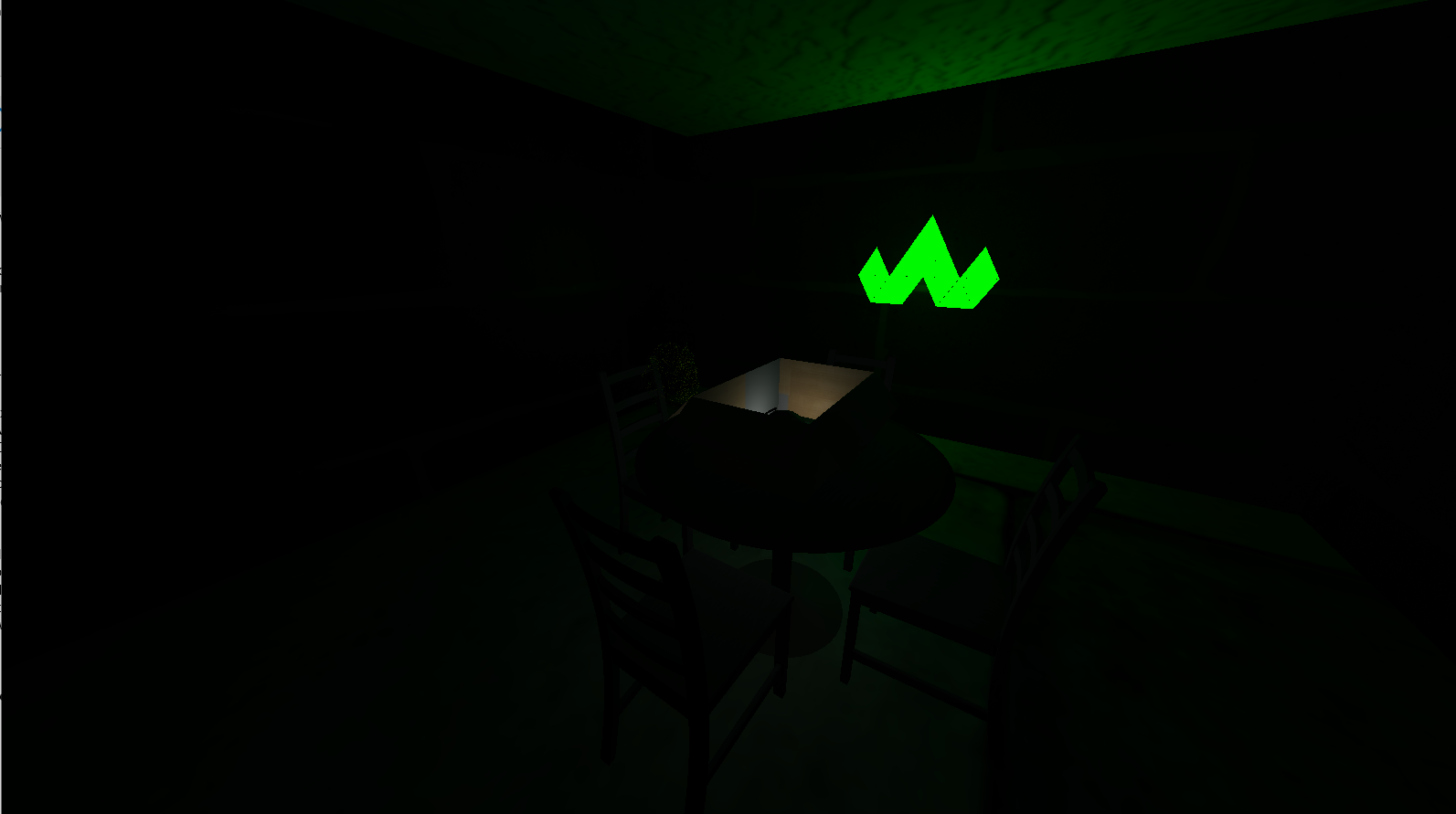
<https://youtu.be/4dJJfTaGy0I?t=479>

I uploaded my src files and assets to BB, I was not able to upload my whole VS project to BB since it was >1GB however I uploaded my entire VS project to my OneDrive in case you can’t get it working as a new project. It can be found here:

**Screenshots**

****

****

****

**Checklist**

1. (10 pts) Replace the items in your box (on the table) with more complex objects. They can represent a mug with a handle, a globe, or any other objects you like.

2. (10 pts) Add some objects on your table, like a pc/laptop or any item you may like. Add some chairs behind the table.

3. (15 pts) Add walls and floor to your room. This will make your table looks like sitting on something.

4. (30 pts) Add at least three distinct point light sources to your scene. The user should be able to turn them on and off individually. The objects should respond to all light sources.

5. (15 pts) Create at least four cameras for your scene. The user should be able to switch between them to look at the scene from different perspectives.

6. (20 pts) Add your working clock from assignment 2 to a clock object on the table, or to a clock object on the wall. It should show current time.

7. (potential 20 points) Optional: Create an object using Bezier surfaces. A cushion for chairs is a good example. You have to do the calculations in your app to get the points.

8. (potential 20 points) Optional: Add texture to your objects. You may also put some artwork on the walls.

9. (potential 20 points) Optional: Load models designed by third party softwares (e.g: Blender), and display them.

10. (potential 40 points) Optional: Add shadows for the objects.