**Computer Graphics ClassAssignment2 Report**

**Prepared By: Wan (2019007901)**

Implemented Requirements

A. Manipulating the camera the same way in ClassAssignment1 using the same code. 18x18 reference grid is also included.

B. i. When user does a drag and drop action, the program will run in single mesh rendering mode.

B. ii. The viewer will only render one mesh at a time. New dropped .obj file will be replaced by the previous rendered object.

B.iii. Only read the vertex position, normal and faces to render the mesh. The meshes are rendered by only using glDrawArrays().

B.iv. When opening an .obj file, the file name, total faces, and total faces with 3 vertices, 4 vertices and more than 4 vertices will be printed on console.

C.i. When user press ‘h’ key, program will run in animating hierarchical model rendering mode.

C.ii. The hierarchical model used in this program includes 7 .obj files located in ‘HierarchicalModelFiles’ directory. Relative paths are used to specify these files in the directory to be read by the program.

C.iii None of the sample .obj files given by professor are used for the hierarchical rendering mode.

C.iv OpenGL matrix stack are used to animate the hierarchical model using glPopMatrix() and glPushMatrix().

C.v This hierarchical model has 3 levels and each internal node and root has 2 child nodes. All nodes are visible and rendered in the program.

C.vi. All child parts are moving relatively to their parent parts. The model is automatically animated without any input from keyboard and mouse.

D.i. There are multiple light sources which will be explained in detail in later section.

D.ii. Wireframe mode can be toggled by pressing ‘z’ key.

EXTRA CRED.B. Meshes consisting of different polygon number are splitted into triangles using triangulation algorithm. Source: <https://stackoverflow.com/questions/23723993/converting-quadriladerals-in-an-obj-file-into-triangles>.

Hierarchical Model Animation Video

<https://www.youtube.com/watch?v=wEsQtRX2BXk>

Lighting Configuration

Number of light sources: 3

Location of light sources: Light0 (-10,-10,-10), Light1 (10,10,10), Light2 (-10,10,10)

Type of light: Light0 (point), Light1 (point), Light2 (directional)

Model Hierarchy

