Assignment 3 brief Report

I implemented the model viewer by using the lecture examples as samples and collecting all the ideas from lectures examples. My model viewer mainly consists of two parts : debugging part and lighting part. The user can enters the the debugging part by default mode. In debugging mode, the user can check if the edges are drawn correctly , normals are assigned correctly and materials. The way that I implemented is simple. I use the key_callback to check what key is pressed. If it is “D” and the count is only 0, the code line (glPolygonMode will be activated and it will show only the edges of each triangle) .If user presses D again , the count will increase and fragcolour will be assigned with diffuse material color and then will be assigned colors depending on each vertex’s normal.

In lighting Part, most of ideas comes from the sphere light example from lecture. I would like to mention about yellow light rotating part here. To rotate the yellow light, i used the rotation matrix and in vert shader, light position is multiply with rotation matrix .View matrix is the sample same like the modelview example from lecture. I implemented my program as simple as possible and all ideas came from lecture example.

The Problems that I faced

I believe that my program can be improved a lot.For example, in flythrough mode, the way that I wanted to implement is that flythrough mode will be keep going as long as the user presses the key. But, now , even though it has flythrough mode, the user needs to press key again and again to move. Second problem that i faced is that the 0bject became upside down when user presses the left button of the mouse for the first time. There might be some minor errors that can be fixed easily for that. I just needed more time to fix it.

The extra features

As extra features, i could do more than 1 extra feature.

1. Background options : The background will be changed to other colors if the user presses “B” button. But, I couldn’t find a way to load “image” to the background.

2. Flythrough camera : is implemented . The user needs to press arrow keys to test “flythrough mode”. The only bug that hasn’t fixed is that the user needs to press the arrow key button again and again to move around .

3. Auto-generated normal : Normal will be calculated when vertex normals are not specified as part of the model as shown in lecture.