Computer Graphics Assignment 2:
Obj Viewer

2015005078 정진교

1. How to run my program

   I used version 3.7.2 of Python, version 1.16.2 of numpy, version 3.1.0 of OpenGL and version 1.7.1 of glfw.

   You can just run using this command – "python3 2015005078-class2.py"

2. Which requirements I implemented

   I implemented requirements that I manipulate the camera with mouse movement, load an obj file and render it and lighting.

   First, I used the camera function of ClassAssignment1 so I will skip the explanation. I also drew the reference grid plane.

   Second, I implemented opening an obj file by dragging and dropping it into my obj viewer window. I opened and read an obj file using glfwSetDropCallback. I declared the list in a function so only one obj file is executed at a time. I opened the file and read the lines one by one, finding the v, vn and f tokens and parsing the arguments. The v token's arguments are stored in vparr and the vn token's arguments are stored in vnarr. After incrementing the count according to the number of arguments in the f token, the values stored in vparr and vnarr for three arguments are stored in varr to create a triangular mesh. If there are more than three, the first argument is used in duplicate. Then my program prints File name, Total number of faces, Number of faces with 3 vertices, Number of faces with 4 vertices and Number of faces with more than 4 vertices and render it using varr. If the Z key is pressed, the wireframe / solid mode is executed alternately.

   Third, I made two light sources.

3. Lighting configuration

   My program has two light sources. One is in (4., 5., 6.) and the other is in (-3., 4., -5.). Both are point lights.

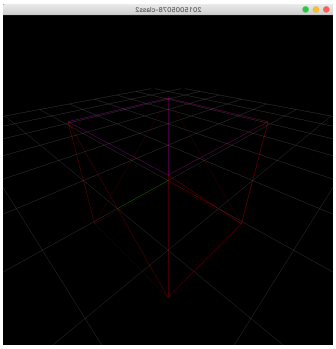
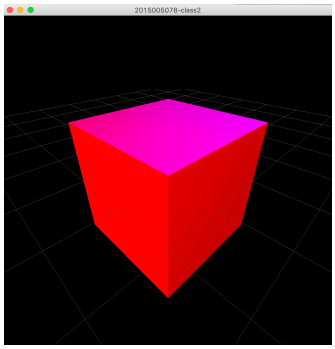
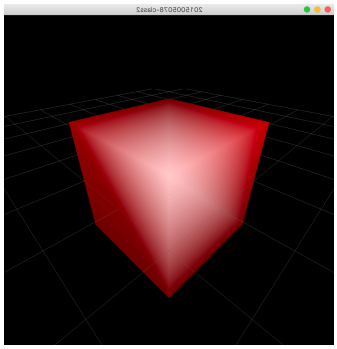
4. Extra credits

   I used glDrawArrays() to render a triangle mesh in shading using normal data in obj file mode. I used glDrawElements() in forced smooth shading mode.

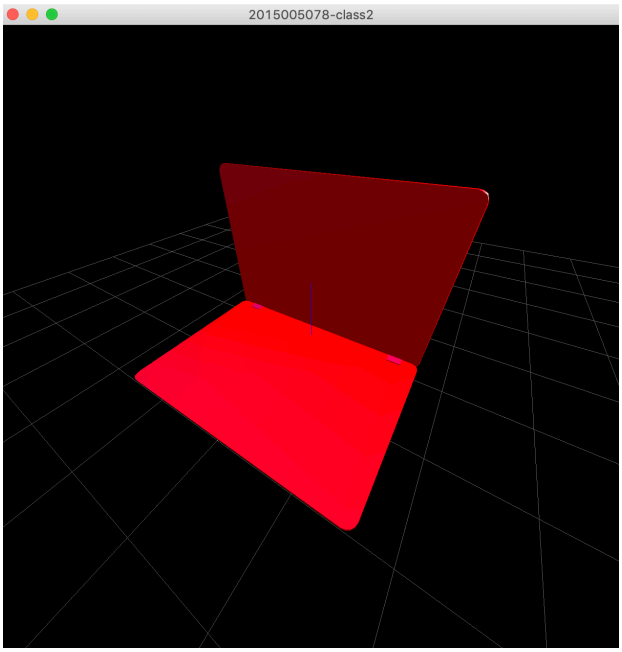
   After incrementing the count according to the number of arguments in the f token, three arguments are used to calculate the face's normal vector and are stored in iarr. If there are more than three, the first argument is used in duplicate. The face's normal vector is added to the smooth list at each vertex index. After reading all the file, the values in the smooth list are averaged per vertex and stored in sarr. If the S key is pressed, [shading using normal data in obj file]/[forced smooth shading] mode is executed alternately.

5. A few screenshot images of your program

```
File name : /Users/jinkyo/정진교/3-1/컴퓨터그래픽스/실습/ClassAssignment2-obj/cube-tri
.obj
Total number of faces : 12
Number of faces with 3 vertices : 12
Number of faces with 4 vertices : 0
Number of faces with more than 4 vertices : 0
```

```
File name : /Users/jinkyo/정진교/3-1/컴퓨터그래픽스/실습/ClassAssignment2-obj/Lowpoly_
Notebook_2.obj
Total number of faces : 161
Number of faces with 3 vertices : 57
Number of faces with 4 vertices : 92
Number of faces with more than 4 vertices : 12
```




```
File name : /Users/jinkyo/정진교/3-1/컴퓨터그래픽스/실습/ClassAssignment2-obj/Intergal
actic_Spaceship-(Wavefront).obj
Total number of faces : 55120
Number of faces with 3 vertices : 55120
Number of faces with 4 vertices : 0
Number of faces with more than 4 vertices : 0
```