

Assignment 2. Lighting and Shading

Total of Points of the Assignment: 20

Your implementation of assignment 1 can transform and project triangular models. For the rendering, however, it still contains no lighting and shading. The goal of this 2nd assignment is to extend your previous program to lighting and shading implementation.

You are required to write your program using OpenGL shading language (GLSL) to shade a triangular model. Your program should provide at least the following features:

- a) Implement Phong Illumination Model in GLSL. Please ignore the distance attenuation factor, i.e., by setting the whole attenuation factor to be 1. For computing the specular component of a light source, please use the halfway vector instead of the reflection vector. **(8 points)**
- b) Support for at least two light sources: one being static and one being dynamic. The static one is a directional light with direction (0, -1, -1) in camera coordinate system. The dynamic light is a point source rotating around the center of the triangular model. **(3 points)**
- c) For the dynamic light, it should be initially placed at the location of the camera, i.e., (0, 0, 0) in camera coordinate system. The user can choose to rotate it around X, Y, or Z-axis of the world coordinate system, but keeping the rotation center fixed. Be able to turn the rotation of light source on and off. When rotation is turned on, the light source will automatically rotate around the model. When rotation is turned off, the light source will stop rotating and be placed wherever it is. Have a “reset” option to reset the dynamic light to its initial location. **(3 points)**
- d) Turn light(s) on and off. **(1 point)**
- e) Support both flat and smooth shading models. **(1 point)**
- f) Support for interactive changes of the (RGBA) values associated with the global ambient light. **(1 point)**
- g) Support for interactive changes of the (RGBA) values associated with the ambient, diffuse and specular component of the light sources. **(2 points)**
- h) Handling of visibility z-buffering. You need to load an .obj file named “two_cubes.obj” to show z-buffering effect **(1 point)**

(Hint for feature (e): After OpenGL 4.0, flat shading is facilitated by the interpolation qualifiers available for shader input/output variables. `glShaderMode(GL_FLAT)` is no more available. Try “flat/smooth interpolation qualifiers” instead).