02561 COMPUTER GRAPHICS IMM. DTU

Exercise 4: Local illumination / Phong lighting and Gouraud / Phong shading

	Angel: chapter 5.1 – 5-10
Purpose	The purpose of this exercise is to get acquainted with lighting and shading in OpenGL and GLSL. You will calculate the shading of an object based on the ambient, diffuse and specular properties of the material and the light source.
Part 1 Gouraud shading	Modify the vertex shader implement Phong ligthing using Gouraud. Use the book as inspiration. Some hints: The lightPosition in the shader is in eye-coordinates. All uniforms and vertex attributes are already setup – you only need to change the vertex shader. The suggested order: • Set the color to the ambient contribution • Compute the diffuse contribution • Add the diffuse contribution to the color • Compute the specular contribution • Add the specular contribution to color
Part 2 Phong shading	• Implement Phong shading using the vertex and fragment shader. ○ The normal, eye space position and the light position should all be transferred from vertex shader to fragment shader using varying variables (in/out variables). ○ Compute the ambient, diffuse and specular contribution in the fragment shader. • Extend the program to toggle between point light and directional light. (Hint: Use the `w` component of the LightPosition to store the light type, where 0 means directional and 1 means point light). ○ Modify shader to use support directional light ○ Modify the `keyboard` function in main-04-02.cpp ○ Depending on how your directional light is implemented, you may need to change the position (x, y, z) of the light position.

Exercise 4: Local illumination / Phong lighting and Gouraud / Phong shading

Part 3 Shading and light questions	Answer the following questions: a) What is the difference between Phong shading and Phong lighting b) What is the difference between Gouraud shading and Phong shading. What is the pros and cons of each? c) What is the difference between point light and directional light? d) Has the eye position any influence on the shading of the object? e) What is the effect of setting the specular term equal to (0,0,0)? f) What is the effect of increasing shininess exponent? g) Compare your implementation of in Part 1+2 with the Phong model (Angel chapter 5.3). Did you make any simplifications of the model? If so explain the simplifications you did and the impact of these simplifications. h) Explain the importance of the normal matrix. What is the purpose of the normal matrix and how is it computed. i) In what coordinate space are you computing the light? (Model-space, world-space, eye-space, clip-space)?
Part 4 Multiple light sources	We will here look at how to support multiple light-sources affecting an object. • Copy the phong shader from part 2 as a starting point. • Extend the C++ and glsl to use three light positions instead of one. • In the fragment shader iterate over the light positions and sum up the result. Warning: ambient should only be added once. • Set the three light positions to three different positions in the scene, so distinct highlights can be seen. You may need to adjust the light intensity for all 3 light sources to be seen.
Part 5 Optional Multiple light sources cont.	Extend part 4 to use different light colors. • Material color (ambient, diffuse and specular) needs to be changed into a fragment shader uniform. • Light color (ambient, diffuse and specular) needs to be changed into a fragment shader uniform. There need to be one diffuse and specular color for each light position. • Ambient-product, diffuse-product and specular-product needs to be computed in the fragment shader and removed as a shader-uniform. • Set the light diffuse color to be red, green and blue.