Assignment 3: Shiny Gloomy Bunny

In this assignment, you will simulate a single spot light around the model of a bunny. As the light position will be constantly updated, sometimes the bunny will be lit, sometimes it will look dark as shown in the figure. You'll be able to switch between Phong shading and Gouraud shading.

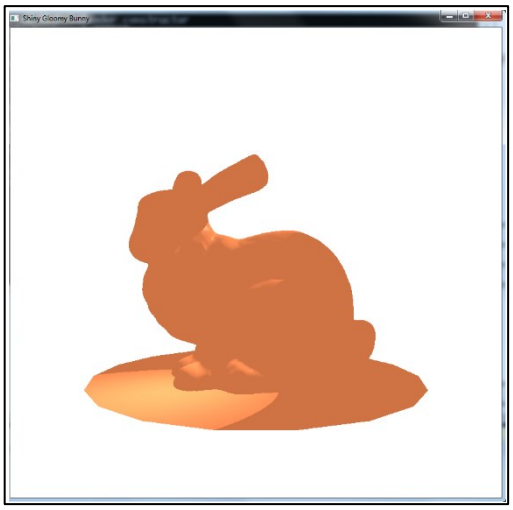

Figure 1: Shiny Gloomy Bunny

Implementation:

You can use the files for Assignment 2. Here are some steps that you need to implement for this assignment:

- Download the files for Assignment 2;
- Shade the bunny as you did for Assignment 2;
- Take the camera position to (0.0f, 0.0f, 30.5f);
- Unitize the model by enclosing in a box of dimension 2X2X2 centered at (0.0, 0.0, 0.0);
- Uniformly scale the model by 7.0f;
- Draw a disc of radius '10' at the bottom of the bunny
- Use the following position of the light as mentioned below. As shown below, position of the spot light will be constantly updated as the variable 'angle' varies. Use 'glutTimerFunc' to update the angle automatically at a certain time interval;

```
vec4 lightPos = vec4(10.0f*cos(angle), 10.0f, 10.0f*sin(angle), 1.0f);
```

- In your application, you should be able to switch between Phong shading and Gouraud shading;

Spot light implementation:

Spot light is considered as a point light that emits light within a restricted cone as shown in Figure 2. Dot product of the spot light's focus direction and the light direction (negated) is computed (shown as angle 'β' in figure) and compared to a precomputed cosine cutoff value (15 degree in our case) to determine whether the position of the surface is inside or outside the spotlight. If 'β' is less than the cutoff value, that part of the surface is within the cone, otherwise surface will be lit only using the ambient component of the light.

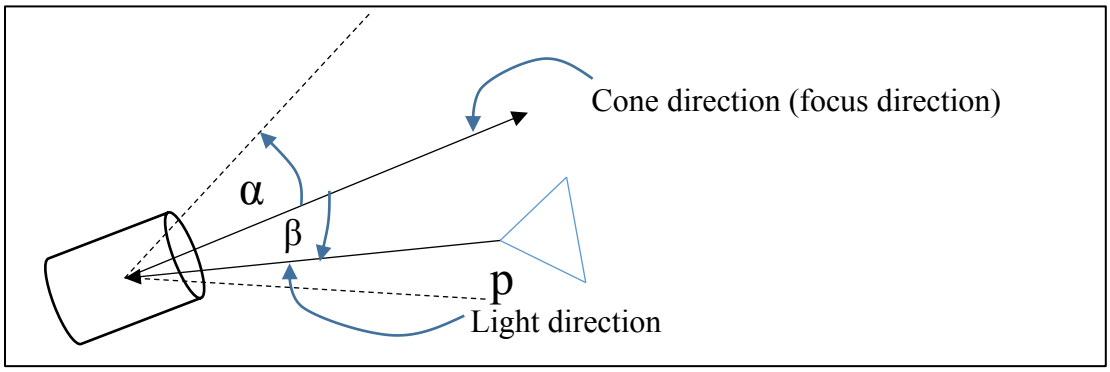

Figure 2: Spot light computation.

In **main.cpp**, you need to add information about the following uniform variables. You also need to add these uniform variables in your shader files.

```
glUniform1f(glGetUniformLocation(currentProgram, "Spot_exponent"), 30.0f);
glUniform1f(glGetUniformLocation(currentProgram, "Spot_cutoff"), 15.0f);
glUniform3fv(glGetUniformLocation(currentProgram, "Spot_direction"), 1,
(GLfloat*)&spot_direction);
```

You need to compute the direction of the spot light and relate it to the uniform variable "Spot_direction" defined in the shader.

You need to calculate the angle 'β' and compare it with the given cutoff value to compute the light intensity within the cone of the spot light.

If 'β' is less than the cutoff value, the attenuation factor is computed as $\cos^{Spot.expo}(\beta)$. This attenuation factor is multiplied by the intensity of the spot light. Finally, along with the ambient component, diffuse and specular terms are added to calculate the final light intensity of the surface inside the cone.

Submission:

Submit the assignment in a zipped file via canvas. Name the file as **Firstname_Lastname_3_CSCD470.zip**. Deadline is Tuesday, May 8, 11:59 pm.

This assignment carries a weightage of 15% of this course.