Assignment 4 for csci580

*** Adding Shading to the Gz rendering library ***

You will add a flexible shader to your Gz rendering library. The renderer should continue to support flat-shading (like in HW2/HW3). In addition, you should now support Phong shading of vertices with linear (screen space) color interpolation over the triangle. AND, you should support full Phong shading at each pixel with bilinear (screen space) normal interpolation over the triangle.

The shading mode is selected by choosing an interpolation mode.
The GZ_INTERPOLATE flag selects between flat shading, vertex lighting with color interpolation, and normal interpolation with each pixel getting a separate lighting computation.

You must support two light types GZ_AMBIENT_LIGHT and a directional light specified by GZ_DIRECTIONAL_LIGHT. Both are of type GzLight (in Gz.h). The direction vector is only valid for directional lights and it is the vector from the scene to the light in image-space. Direction is constant since we're assuming the light is very far away. Ambient light direction has no meaning and is not used.

The API specifies that only one ambient light is supported, but multiple directional lights are legal (up to 10). Lights are specified like most everything else, through a call to GzPutAttribute(). (see below)

The hw4 directory has the Gz.h file with comments for the items you need defined for the API.

Here is the list of items....

GzPutAttribute must accept the following tokens and values:

token	value type	
GZ_RGB_COLOR	GzColor	set default flat-shade color
GZ_INTERPOLATE	int	shader interpolation mode
GZ_DIRECTIONAL_LIGHT	GzLight	add a directional light
GZ_AMBIENT_LIGHT	GzLight	set ambient light color
GZ_AMBIENT_COEFFICIENT	GzColor	Ka ambient reflectance coef's
GZ_DIFFUSE_COEFFICIENT	GzColor	Kd diffuse reflectance coef's
GZ_SPECULAR_COEFFICIENT	GzColor	Ks specular reflectance coef's
GZ_DISTRIBUTION_COEFFICIENT	float	specular power

Below are flag values used for GZ_INTERPOLATE values

Set GZ_INTERPOLATE as the shader interpolation mode to use
GZ_FLAT 0 /* do flat shading with GZ_RBG_COLOR */
GZ_COLOR 1 /* interpolate vertex color */
GZ_NORMALS 2 /* interpolate normals */

Gz.h also specifies a GzLight type which is needed since the application has to set up lights.

P.S. If you have compile error related to GzLight, please delete dummy definition in your “rend.h” file. It’s defined in the new gz.h.