Craig Harris
harricra@oregonstate.edu
Project 6: Shaders
[Video Link]

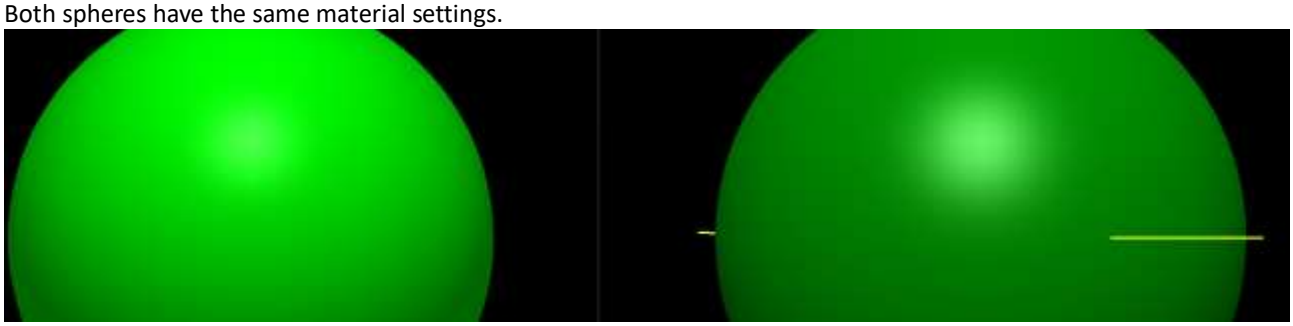
Description:
I created display lists for the OSU Sphere, Torus, and Cone shapes. The shaders are setup in sample.cpp using the glslprogram C++ file provided. Uniform variables for lighting attributes, base color, specular color, shininess, the center and radius of the ellipse (to be superimposed onto the cone/torus/sphere) are setup in sample.cpp. The ellipse attributes are updated in the Display() function while the others are set once in the InitGraphics() function. In the vertex shader, "out" variables are setup to pass the Normal, PointToLight, PointToEye and S and T variable to the fragment shader. The fragment shader, which uses all the uniform variables set up and the variables sent from the vertex shader, first determines whether the current fragment is within the boundary of the ellipse or not. If so, the color is modified. It then handles the per fragment lighting.

The animation of the ellipse's shape can be toggled with 't'.
The animation of the ellipse's position can be toggled with 'k'.
The base shape (cone/torus/sphere) can be scrolled through with 'n'.

Keytime Values:
This is setup to do a sort of squared spiral. It is set up to follow the pairs in the struct, then reverses.

```
ellipseS_Center.Init();
ellipseT_Center.Init();

struct st_pair {
    float s;
    float t;
};
struct st_pair ST_PAIRS[] =
{
    {0.5, 0.5},
    {0.5, 0.5},
    {0.5, 0.5},
    {0.5, 0.5},

    {0.5+.05f, 0.5},
    {0.5     , 0.5+.05f},
    {0.5-.05f, 0.5},
    {0.5     , 0.5-.05f},

    {0.5+.1f, 0.5},
    {0.5    , 0.5+.1f},
    {0.5-.1f, 0.5},
    {0.5    , 0.5-.1f},

    {0.5+.15f, 0.5},
    {0.5     , 0.5+.15f},
    {0.5-.15f, 0.5},
    {0.5     , 0.5-.15f},

    {0.5+.2f, 0.5},
    {0.5    , 0.5+.2f},
    {0.5-.2f, 0.5},
    {0.5    , 0.5-.2f},

    {0.5+.25f, 0.5},
    {0.5     , 0.5+.25f},
    {0.5-.25f, 0.5},
    {0.5     , 0.5-.25f},
};

int num_pairs = sizeof(ST_PAIRS) / sizeof(st_pair);
for (int pair = 0; pair < num_pairs; pair++)
{
    float front_half = (MS_PER_CYCLE / (num_pairs * 2)) * pair;
    float back_half = MS_PER_CYCLE - front_half;
    ellipseS_Center.AddTimeValue(front_half, ST_PAIRS[pair].s);
    ellipseT_Center.AddTimeValue(front_half, ST_PAIRS[pair].t);

    ellipseS_Center.AddTimeValue(back_half, ST_PAIRS[pair].s);
    ellipseT_Center.AddTimeValue(back_half, ST_PAIRS[pair].t);
}
```

What Convinces Me it is Working:
The movement of ellipse's center and radii clearly show what was intended with the keytime and Time variable animations. The color of the base shape and the sphere are correct. The specular lighting effect is not blocky/blotchy as is typical with the non-fragment lighting. You can see in the image on the left, the specular light is a bit more blocky. Both spheres have the same material settings.



Screen Shots: