**Cosc342 Assignment 2a – Ray Tracer Report**

**Testing Code**

Testing my code largely consisted of the same procedure so I will talk about it first. I usually set up a scene with a plane as the floor and a point light set so any object added centrally should cast a shadow on the floor plane. I would usually add an object centrally and rotate it in all directions to make sure the image looked realistic regardless of orientation when testing lighting and objects.

**Diffuse Shading**

Diffuse shading was easy as it was part the labs. To prevent negative lighting, I added an if statement to check that the dot product of the unit light direction and unit normal from hit point were positive before calculating the colour.

**Specular Shading**

Specular shading was similar programmatically to diffuse shading with a different formula and new and different parameters. I gathered the unit reflection about normal and unit view direction and check that the dot product of these was above zero before calculating the colour.

**Shadows**

To implement shadows I cast a secondary ray from the hit point to the direction of the light. I added an if statement that checks if the distance of the first intersection of the shadow ray is greater than the distance to the light source. I moved my light calculations in the body of this if statements so they are only calculated if not in shadow. I made slight error here as the logical operator should have been ‘>=’ rather than ‘>’ as this did not work with directional lighting as both values are often infinity.

**Mirror reflections**

For mirror reflections I cast a mirror ray from the hit point in a direction determined by view direction and the normal from hit point. This was executed recursively with a call to compute colour, making sure to reduce rayDepth by 1 each time. Mirror reflections were a special case for testing my code because the recursion needed to be tested. I set up a narrow hallway of mirrors and placed object centrally to make sure reflections stopped at the given depth.

**Rendering of planes**

I reused the code from the lectures to render planes. I tested planes by rendering them in different positions and rotations with different lighting conditions.

**Rendering of cubes**

Rendering cubes was part of the labs so they were pretty straightforward. The biggest challenge was getting the normals right. This was just a matter of visualising what direction the planes faced on the unit cube.

**Rendering of cylinders**

Rendering cylinders was challenging because it programmatically required mixing aspects of the cube with the sphere. This was easy to image but harder to work out mathematically and implement. The solution was to use the quadratic formula with the equation for a circle to find the intersection points. I then added an if statement to restrict the unit cylinder to ±1 on the Z-axis to make it a cylinder with finite length. I then used planes as the end caps restricted them a circular shape using the equation for a circle and an if statement.

When testing my cylinder, it looked good at first but as soon as I rotated it under any light source it would look strange on the curved surface near end caps. This was because it had the normals of a sphere so I needed to remove the Z component in my normal calculations to change them to cylinder normals.

**Spotlights**

It took me a while to get my head around spotlights. After being reminded that I could use dot products to effectively calculate the angles between two unit vectors I was able to work out how to restrict the light to a cone. To test spotlights I use different coloured spotlights in an overlapping pattern like a Venn diagram to check that the lights worked both independently and collectively.

**Direction lights**

Directional lights were very simple to implement because all the attributes are set by the scene file or are constant so no calculations are required. I had trouble testing directional lights because I made an error in my light calculation code as mentioned in the shadows section. After this was fixed they worked as expected.

**Sample scene**

With my sample scene I decided to show off the directional lighting and shadows by making a sequence of pillars with my cylinder objects to project the light through. I restricted reflections to the spheres so the scene is not completely washed out and I liked the way the pillars reflected on them. I also used a point light to add some more shadows to the scene in different directions. Spotlights were used in a Venn diagram like configuration to show how they interact and also see some more shadow effects.