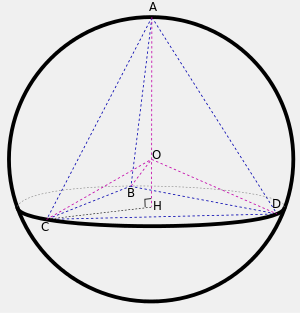
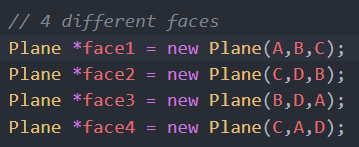
**COSC363 Assignment 2 Report**

**Build Command:** **Linux** = cd Linux && make, **Windows** = cd Windows && make

**Features:**

1. **Tetrahedron**

* The Tetrahedron is drawn as shown

Figure 1- https://en.wikipedia.org/wiki/Tetrahedron

* Constructed 4 faces using the plane class.

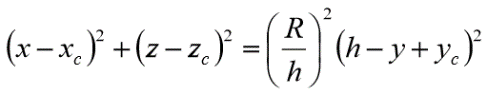
**Ray equation​**

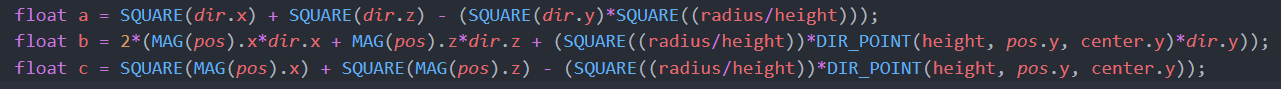
* Sub Ray equation(s) in to intersection equation to find roots.
* Related to specific object: d = direction, o = origin, c = centre, t = distance from ray’s origin to point of ray

1. **Cone**

* Cone class was created in *Cone.h* with a *intersect* and a *normal* method.
* Normal Method:

Normalize’s the given point and returns a unit normal vector

* Intersect Method:

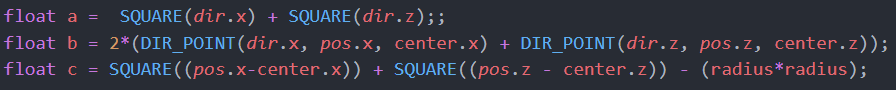
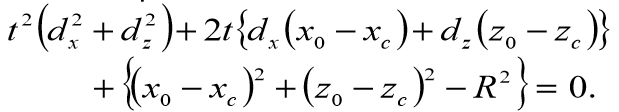
Contains the intersection equation of the cone object and finds the roots below

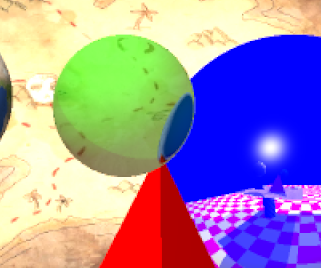
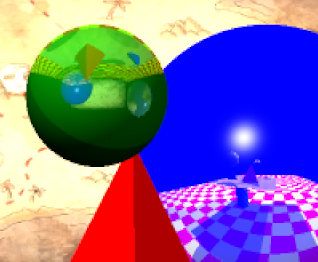
1. **Cylinder**

* Cylinder class was created in *Cylinder.h* with a *intersect* and a *normal* method.
* Normal Method:

Normalize’s the given point and returns a unit normal vector

* Intersect Method:

Contains the intersection equation of the cone object and finds the roots below

****

1. **Refraction and Transparency**

* Algorithm:

ray is traced twice, need 2 normals and refractive rays

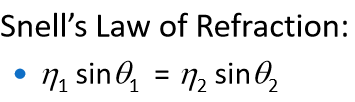
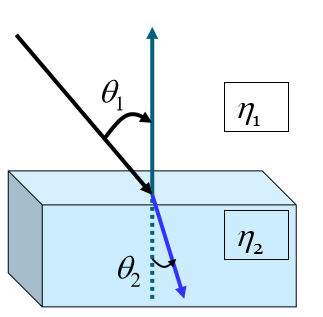
* This is described using Snell’s Law

Figure 4.1 – Taken from lecture 8, slide 19

Figure 3: ETA=1.6

Refraction

Figure 2: ETA=1.003

Transparent

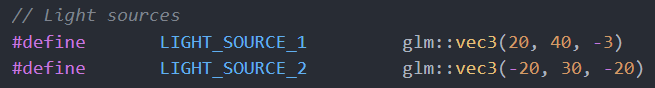
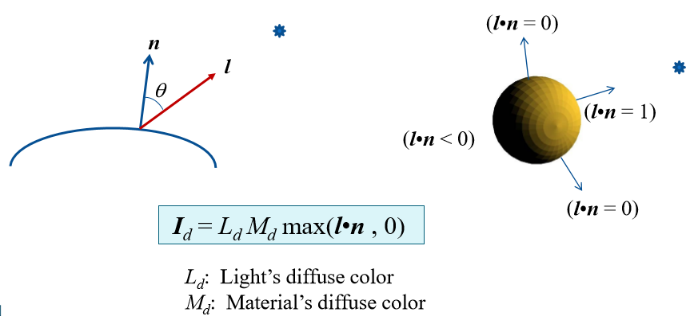
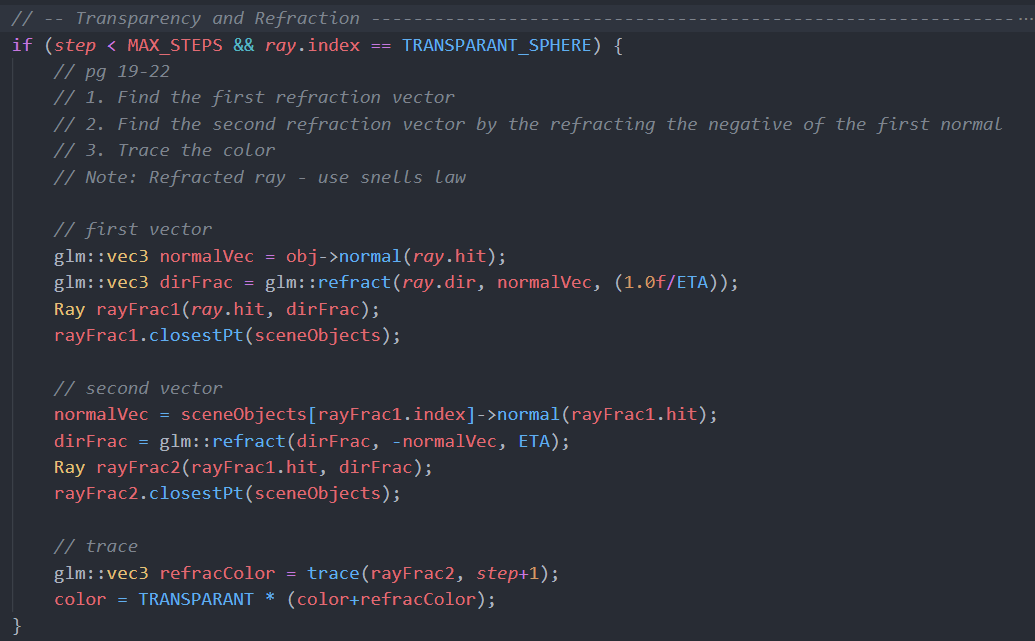
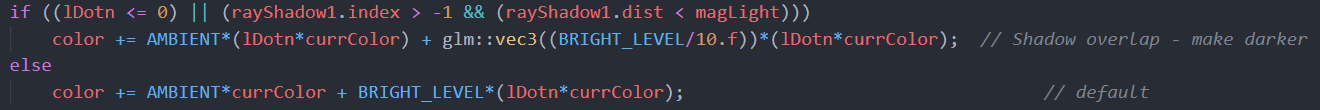
**

Figure 5 – Taken from lecture 7, slide 5

1. **Light Sources – shadows**

Figure 4.2 – Code for part 4

* There are two light sources in the scene.
* Light source 1 reflections are made up of *ambient + diffuse* (shown in fig 5 and 6) and a brightness of 30 %
* Light source 2 is made up of *ambient + diffuse + specular* but reflections are just *ambient.* At 100 %

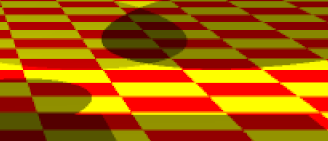


Figure 6 – First two lines controls shaded area – shown below

1. **Non-Planar object textured**

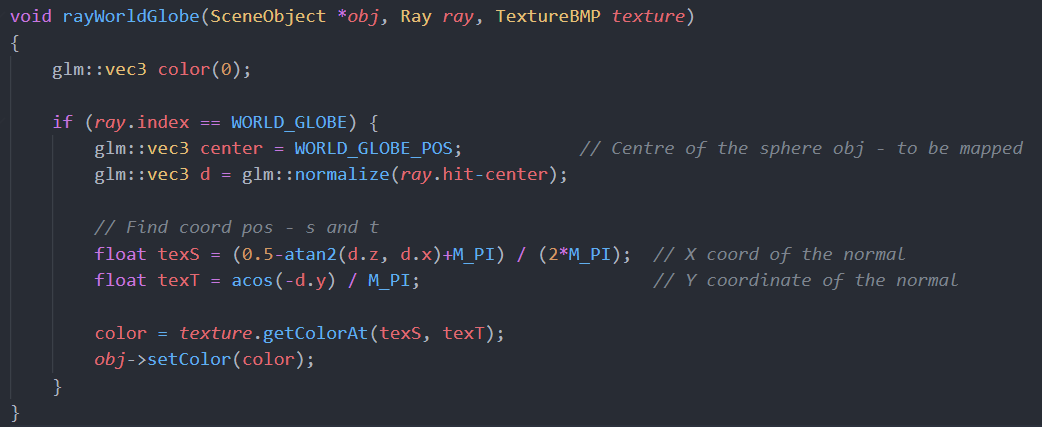
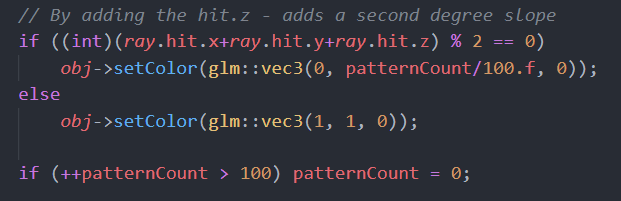
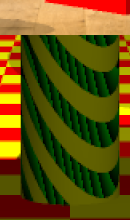
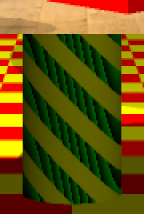
Earth.bmp is mapped to the sphere – algorithm in function – *texS* and *texT* lines

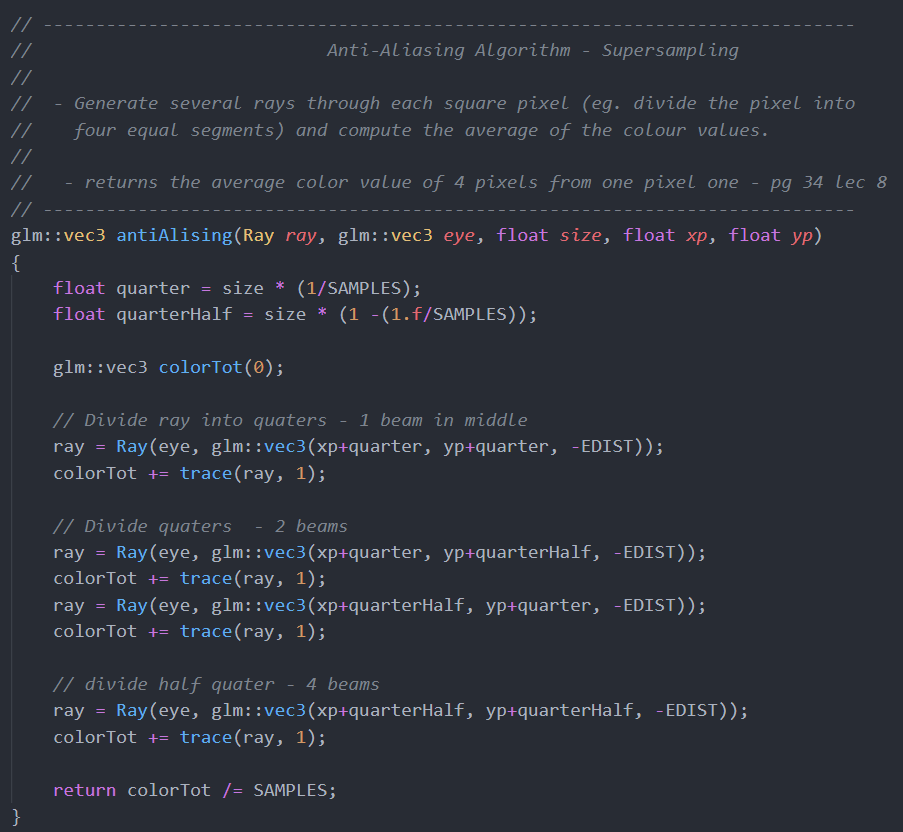
Figure 7 – Shadows overlapping



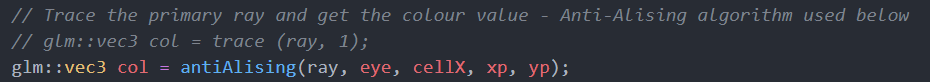
1. **Non-Planar object textured pattern**

The Cylinder is textured by the code above

2nd degree slope- quadratic

****

1. **Anti-Aliasing – Super Sampling**

Algorithm:

Code and Usability

1. Square Pixel is divided in to 4 sub pixels – 1 beam.
2. Two beams are generated by dividing again.
3. 4 beams generated by dividing by a quarter and a half i.e 0.75

Figure 8 shows Aliasing enabled and 9 is with no aliasing