Li Liu

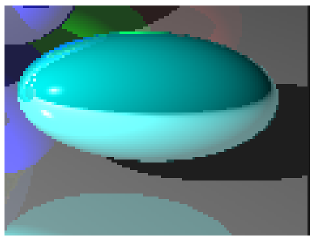
Professor Eric

CSE287 Section A

October 15, 2019

Report for Project 01

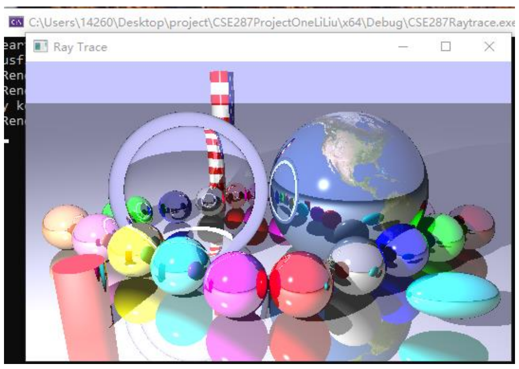
Quadric Surfaces --- Working

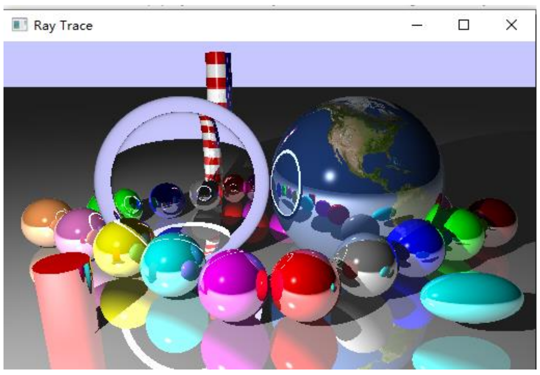


Shadows --- working

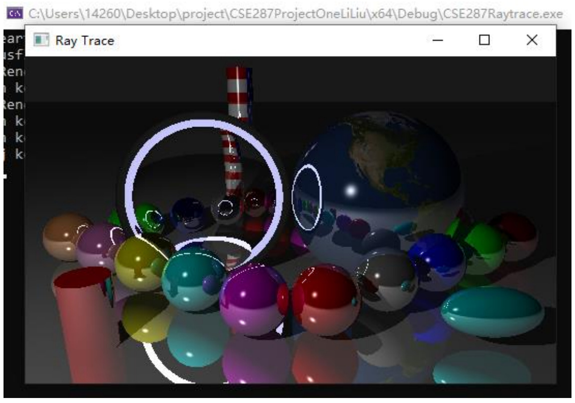
Inter-Object Reflections --- working

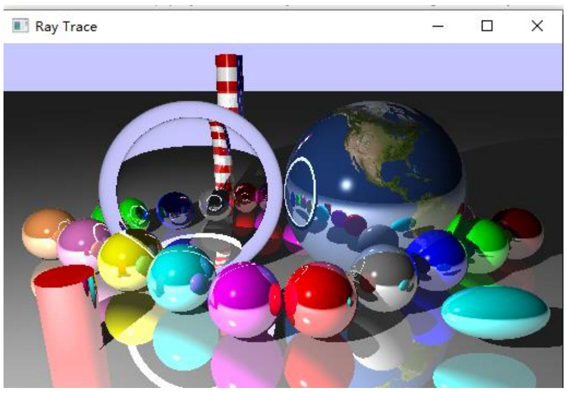
Reflect the “Sky” --- working


C:\Users\14260\Desktop\project\CSE287ProjectOneLiLiu\x64\Debug\CSE287Raytrace.exe
Ray Trace

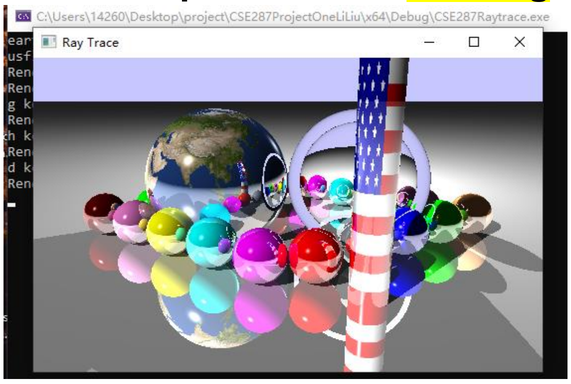


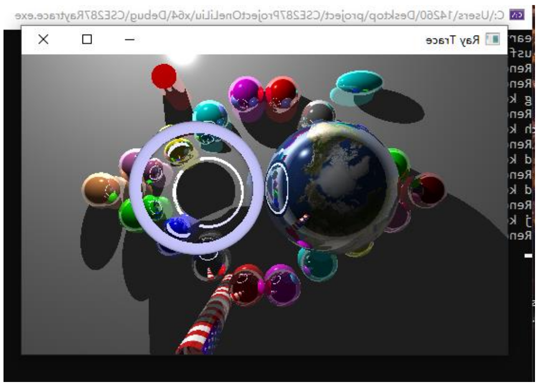
Day and Night --- working

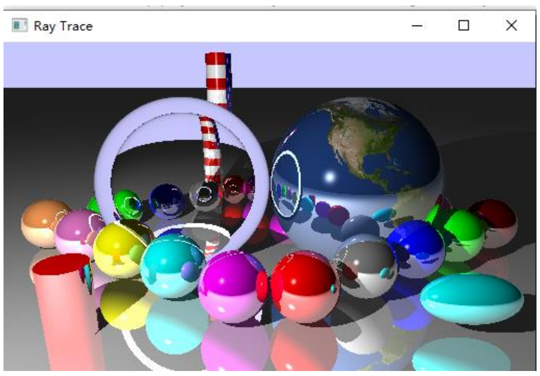

C:\Users\14260\Desktop\project\CSE287ProjectOneLiLiu\x64\Debug\CSE287Raytrace.exe
Ray Trace

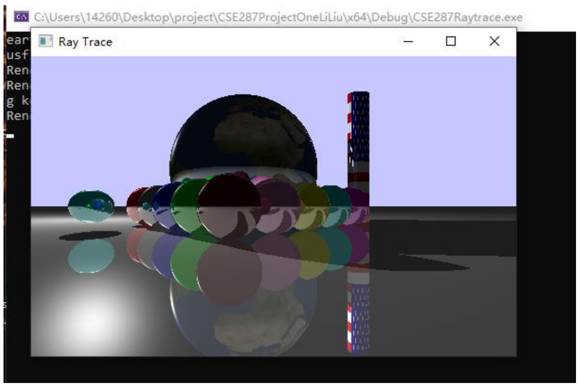


Multiple Views --- working


C:\Users\14260\Desktop\project\CSE287ProjectOneLiLiu\x64\Debug\CSE287Raytrace.exe
Ray Trace



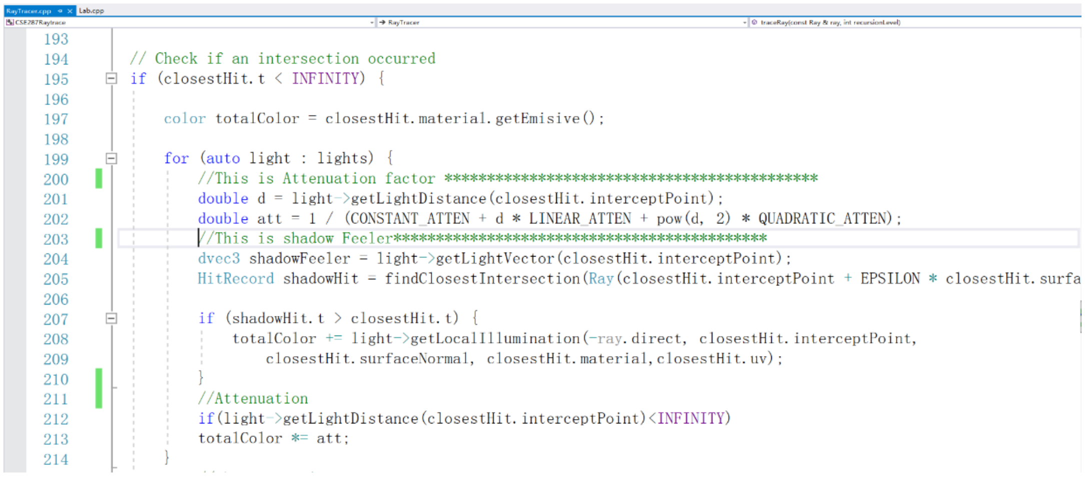



C:\Users\14260\Desktop\project\CSE287ProjectOneLiLiu\x64\Debug\CSE287Raytrace.exe
Ray Trace

Simple Polygon Surfaces --- not working

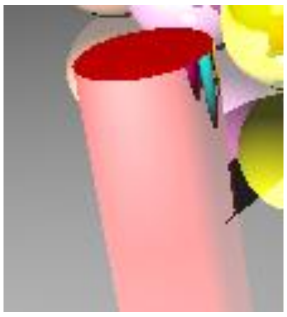
Attenuation --- working

Here is my code for attenuation:

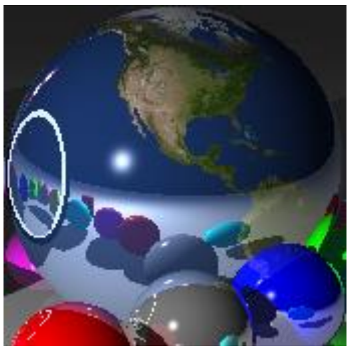


```

// Check if an intersection occurred
if (closestHit.t < INFINITY) {

    color totalColor = closestHit.material.getEmisive();

    for (auto light : lights) {
        //This is Attenuation factor ********************************************
        double d = light->getLightDistance(closestHit.interceptPoint);
        double att = 1 / (CONSTANT_ATTEN + d * LINEAR_ATTEN + pow(d, 2) * QUADRATIC_ATTEN);
        //This is shadow Feeler********************************************
        dvec3 shadowFeeler = light->getLightVector(closestHit.interceptPoint);
        HitRecord shadowHit = findClosestIntersection(Ray(closestHit.interceptPoint + EPSILON * closestHit.surfa

        if (shadowHit.t > closestHit.t) {
            totalColor += light->getLocalIllumination(-ray.direct, closestHit.interceptPoint,
                closestHit.surfaceNormal, closestHit.material,closestHit.uv);
        }
        //Attenuation
        if(light->getLightDistance(closestHit.interceptPoint)<INFINITY)
        totalColor *= att;
    }
```

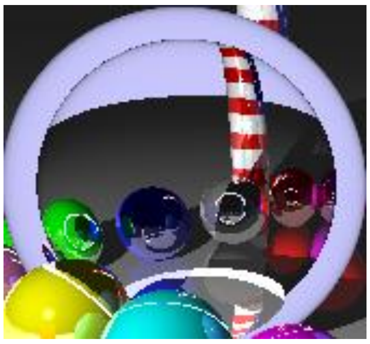
“Capped” Objects ---working



Texture Mapping --- working

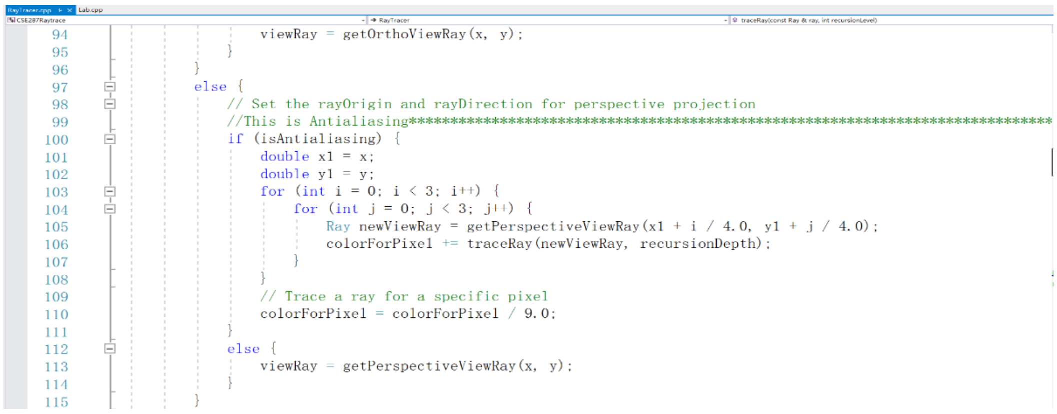


Transparency and Refraction --- working but not perfect



Antialiasing --- working

Here is my code for Antialiasing:



```
        viewRay = getOrthoViewRay(x, y);
    }
}
else {
    // Set the rayOrigin and rayDirection for perspective projection
    //This is Antialiasing***************************************************************************
    if (isAntialiasing) {
        double x1 = x;
        double y1 = y;
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                Ray newViewRay = getPerspectiveViewRay(x1 + i / 4.0, y1 + j / 4.0);
                colorForPixel += traceRay(newViewRay, recursionDepth);
            }
        }
        // Trace a ray for a specific pixel
        colorForPixel = colorForPixel / 9.0;
    }
    else {
        viewRay = getPerspectiveViewRay(x, y);
    }
}
```

OFF:

ON: