

Aaron Templeton

U0734119

CS 4600 – HW5

Raytracing

### Part 1 – Ray Casting

I followed the homework description for this part and checked all the spheres for intersection. If the `doesIntersect` was true, color the pixel red. Also change the original color to be white, then we get the correct background color if no intersection occurs. Then, instead of setting the spheres colors to red, I just changed the code to use the sphere color that it should.

### Part 2 – Shadow Casting

In this part, I added two loops to check the light positions and the intersection of the spheres again. If there are no intersections in this loop, then you get the new color by adding the closest spheres color to create the shadow.

### Part 3 – Illumination Model

Added the new way to calculate the color by using the Diffuse function. The Diffuse function needs the surface normal. In the diffuse function, check if the dot product of the direction vector and the normal is equal to zero. If it is don't change rescolor, otherwise, set the resColor to  $.33 * kd * L \cdot N$ .

The last part involved the phong function. The phong function needs the ray normal, for this just use the `rayDirection` vector. In the homework description it says to use the diffuse function add the value to the product `newColor * ks * f;`