

Menglong shi

Description of the picture below

- P1. The scene without anti-aliasing and fog.
- P2. The scene with anti-aliasing and without fog
- P3. The scene with anti-aliasing and fog.
- P4. The scene with anti-aliasing, and without fog, cap of the cylinder

First Part (basic ray tracer marks:10)

- 1. A good and meaningful spatial arrangement.
- 2. One transparent sphere in front of the cylinder
- 3. All objects have shadows.
- 4. The box around the scene is made up of planes.
- 5. The floor and table have chequered patterns.

The second part (extension marks:7)

- 1. Objects have a sphere, plane, cylinder, or cone. (2 more obj)
- 2. Refraction by the middle of the sphere with eta 1.5
- 3. Two light sources, which cause different shadows and 2 specular highlights
- 4. Adaptive anti-aliasing (show below the First one without it, the second one with it) , compare with the 10% of the mean, and has largest difference with mean.
- 5. Non-planar obj texture, the sphere on the cone.
- 6. Fog showed in the third picture.
- 7. The cylinder with a cap. Without cap picture 4.

Technique used:

- 1. For a cylinder, the intersection equation is used, and the cap of it which determined by if one of the intersection points high value is greater than $y_c + h$ and another is below that point then the intersection should be set at the top cap. Well, I didn't implement the bottom cap as it is hard to see, the return intersection to y_c and the condition is if the intersection point is less than y_c .

2. Same for cone, but work the equation by hand. And also note 7.
3. Refraction follow by lection note
4. Two lights use note 7, but I add the coefficient to both diffuse and specular terms to press the brightness. And shadows condition also learn from here.
5. Adaptive Anti-aliasing is makes it more smooth as it takes average color values of one pixel which has 4 ray, if one of it has significant difference it will generate 4 ray again for that grid.
6. Sphere texture, learn from web,
$$u = 0.5 + \arctan2(dz, dx) / (2 * \pi)$$
$$v = 0.5 - \arcsin(dy) / \pi$$
7. Fog, learn from the note, use linearly blend, if the object in the range of t_1 and t_2 , then its color will be $\text{color} = (1 - t) \text{color} + t \text{white}$. Range - 80, -200 for z_1 and z_2 , which are behind most the obj.

Time take:

With anti-aliasing, it takes around 9s, without it 4s.

Run:

`cmake CmakeLists.txt`

`Make`

`./RayTracer.out`

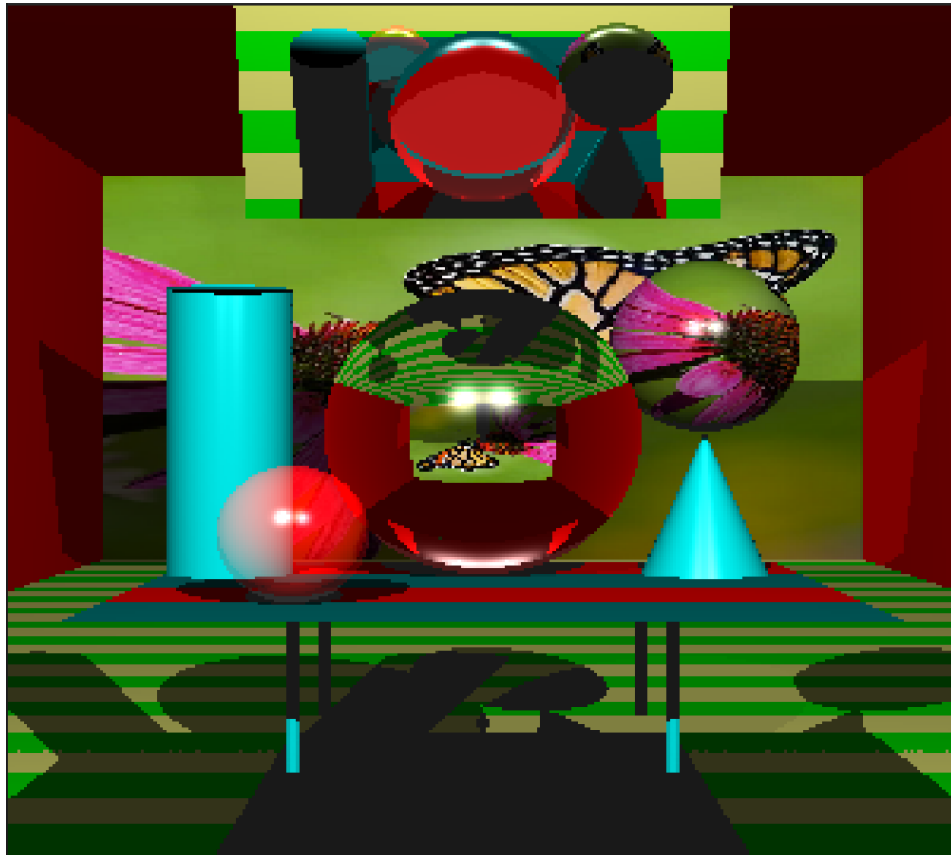
Reference:

<https://stackoverflow.com/questions/22420778/texture-mapping-in-a-ray-tracing-for-sphere-in-c>

https://learn.canterbury.ac.nz/pluginfile.php/4143108/mod_resource/content/29/Lectures/Lec08_RayTracing.pdf

https://learn.canterbury.ac.nz/pluginfile.php/4143109/mod_resource/content/22/Lectures/Note07_RayTracing.pdf

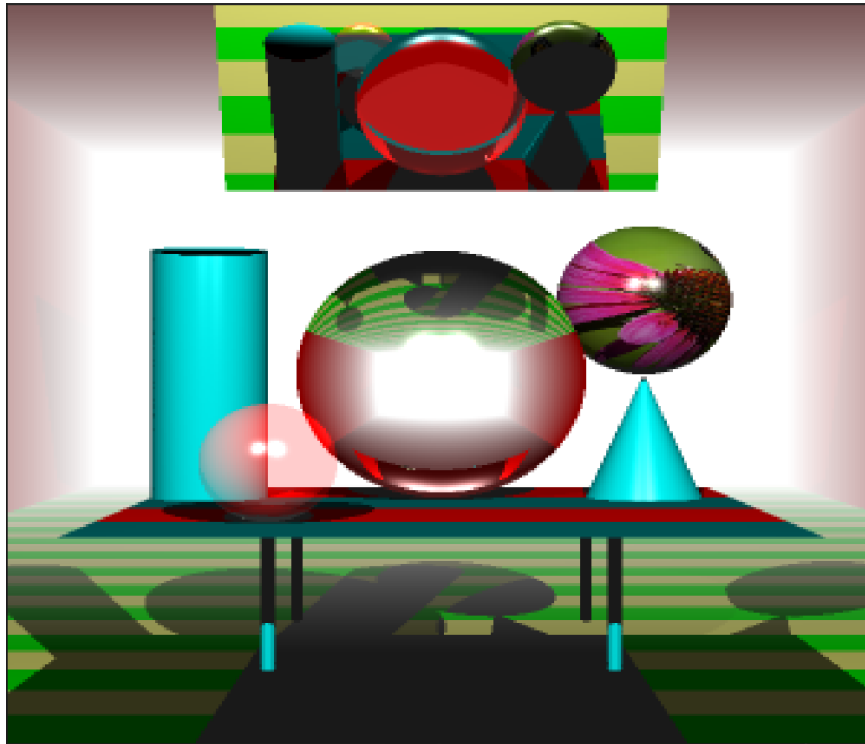
P1



P2



p3



p4

