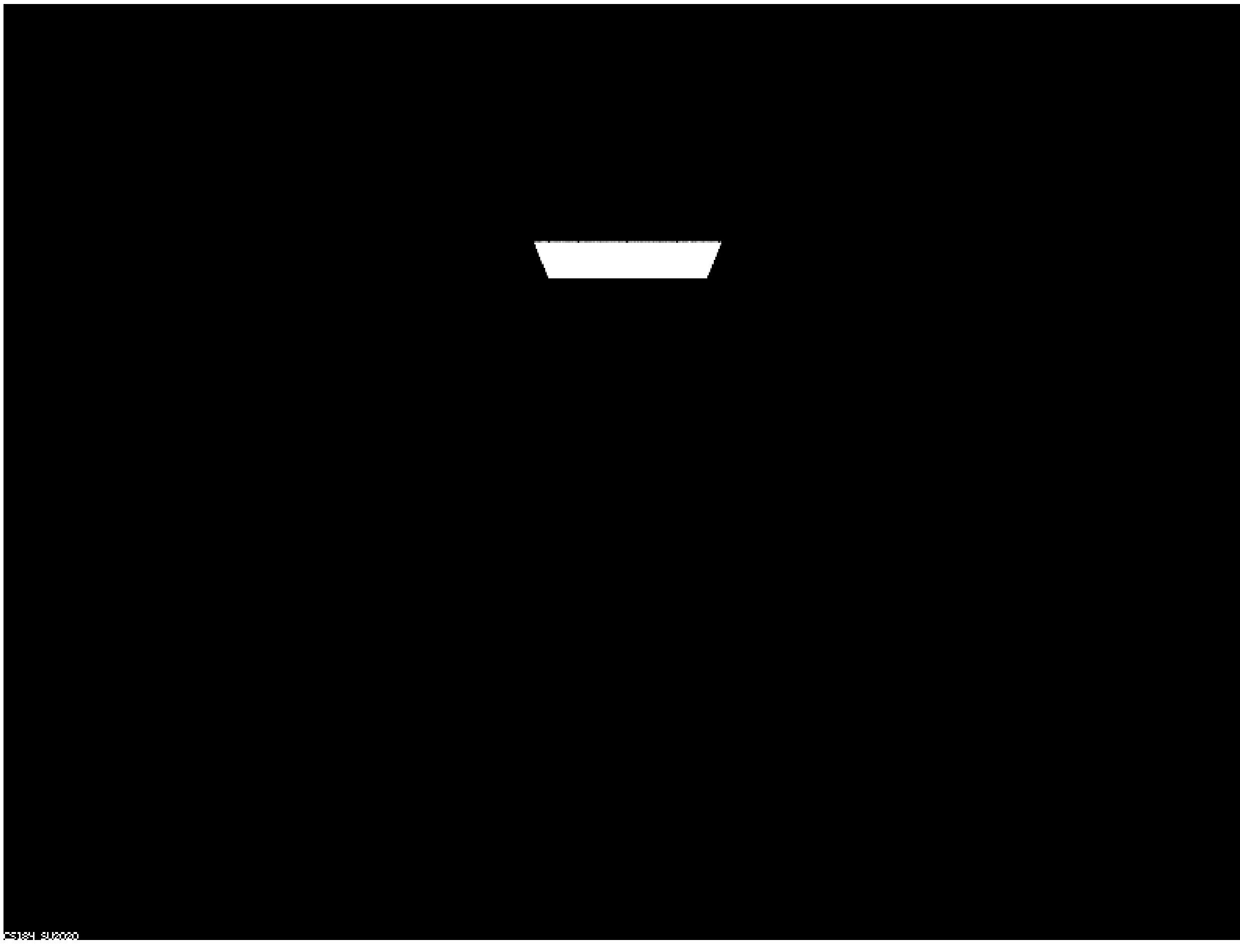
Overview

- Since there were only two parts of this project, the coding portion didn't take as long as project 3-1. We had to go through the reflection and refraction for the mirror and glass spheres. Then we had to simulate a thin lens in order to change the lens radius and focal distance. The main issue we had was trying to run the pathtracer program. This was our main roadblock because we were unable to see our images and thus couldn't test if our code was correct. We had M1/M2 Macs which were unable to test locally. We tried to ssh, but realized that we needed to be on campus using eduroam. Since we live off campus, we needed to wait until the next day to travel to campus to test our code and render our images.

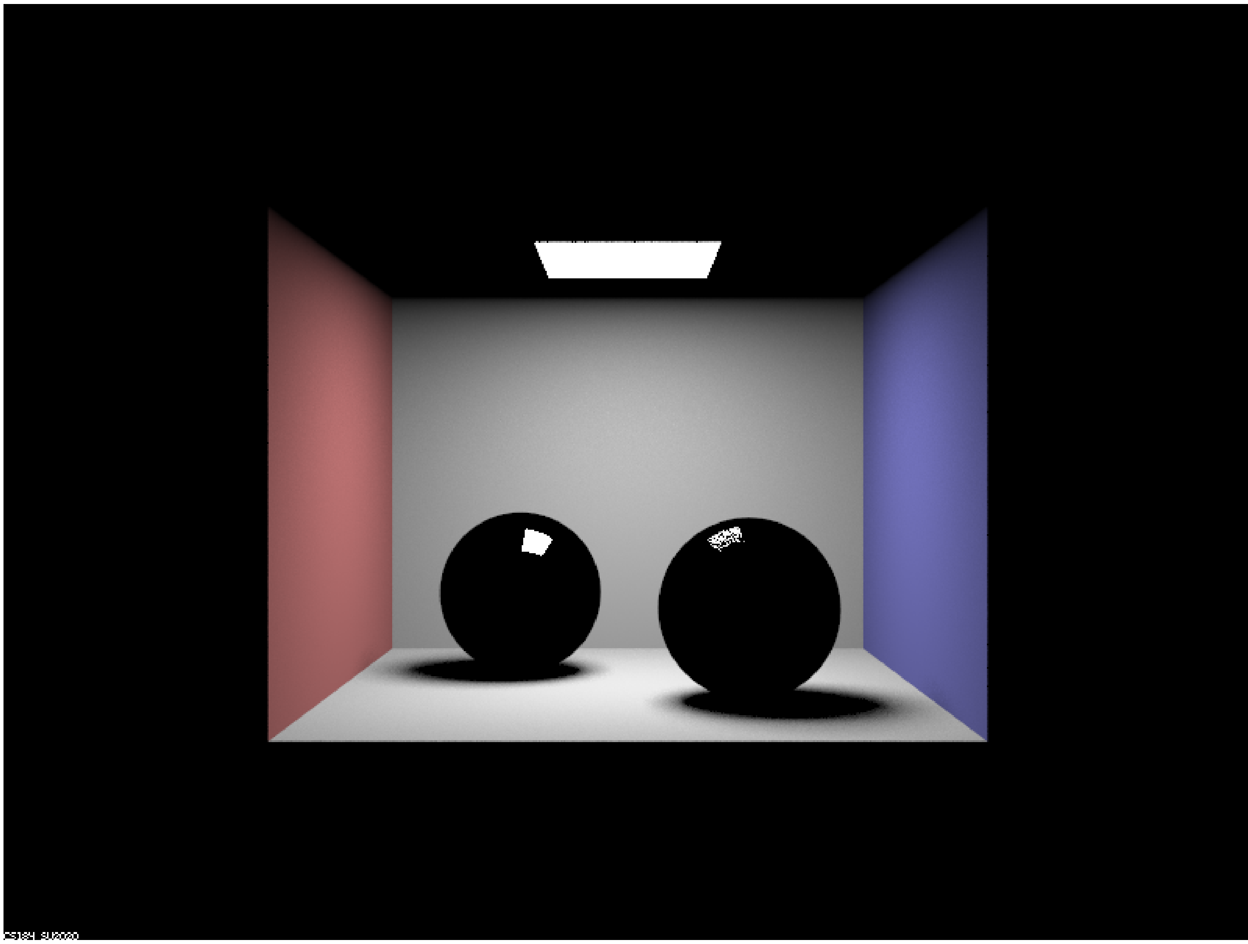
Part 1

- Implementation
    - For reflection, we reflected wo about normal vector (0, 0, 1) and stored the result in wi. For mirror sample_f, we reflected, set the pdf to 1, and returned the reflectance over the absolute value of cosine of wi to cancel out the at least one bounce radiance function. For refraction, we first set the correct eta value depending on wo.z, then we checked if we have total internal reflection. Lastly, we set the correct wi and normalized it based on our calculated eta. For task 4, we implemented glass. Like before, we calculated the eta. If there is total internal reflection, we set the pdf, wo, and then return the absolute value of cosine of wi just like the mirror. Otherwise, we compute Schlick's internal reflection and use it to determine reflection or refraction.
- Show a sequence of six images of scene CBspheres.dae rendered with max_ray_depth set to 0, 1, 2, 3, 4, 5, and 100. The other settings should be at least 64 samples per pixel and 4 samples per light. Make sure to include all screenshots.

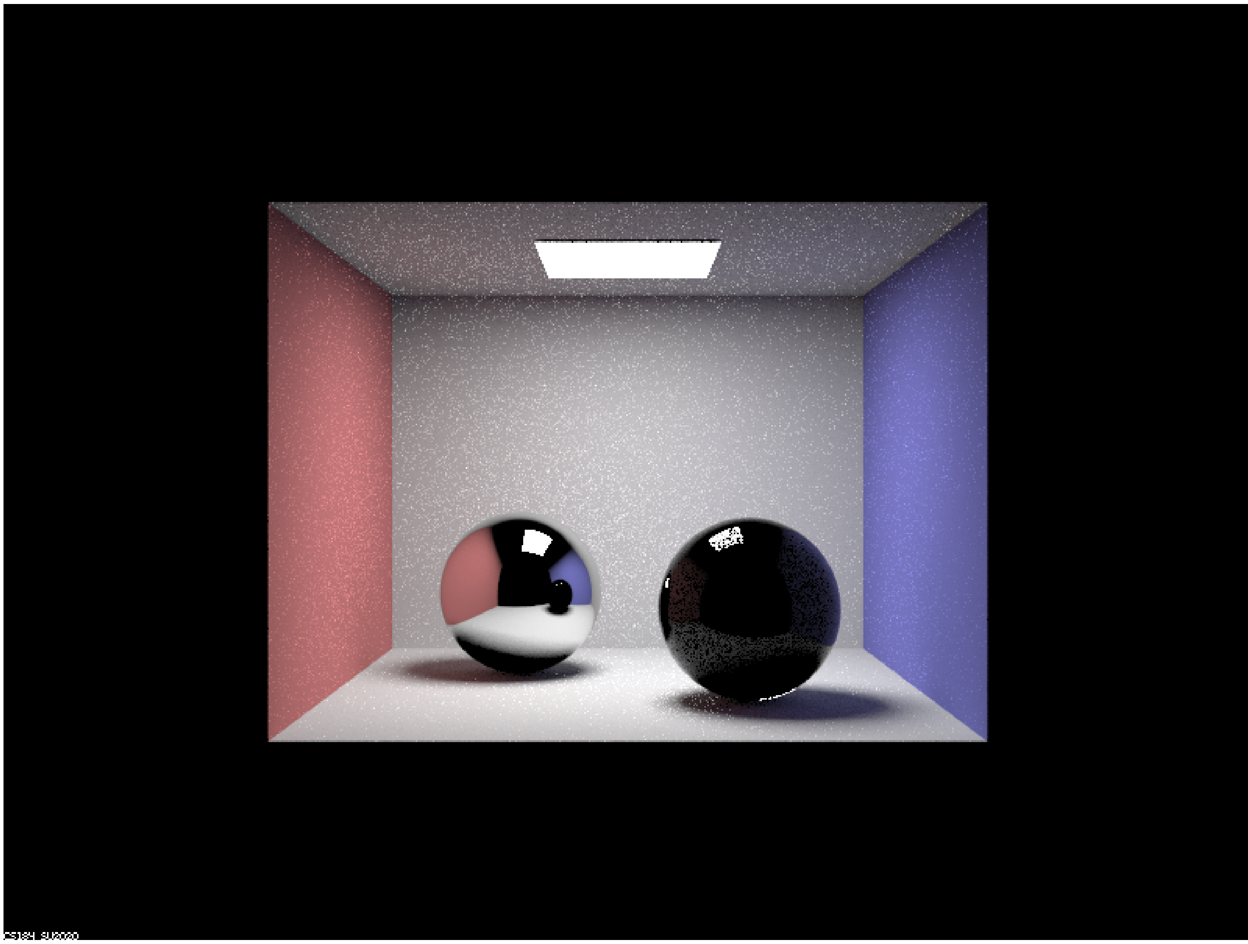
- 0:

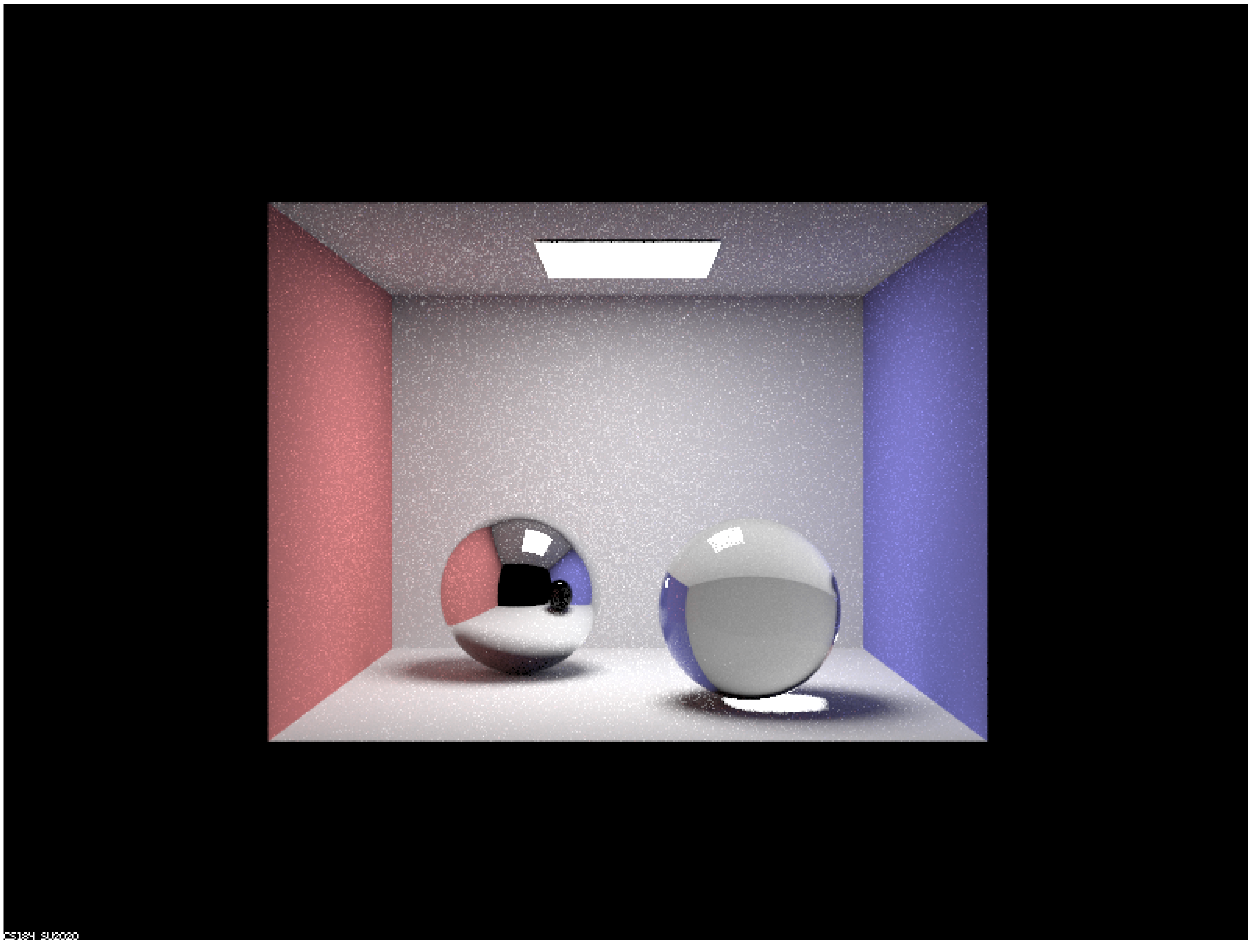
- 1:



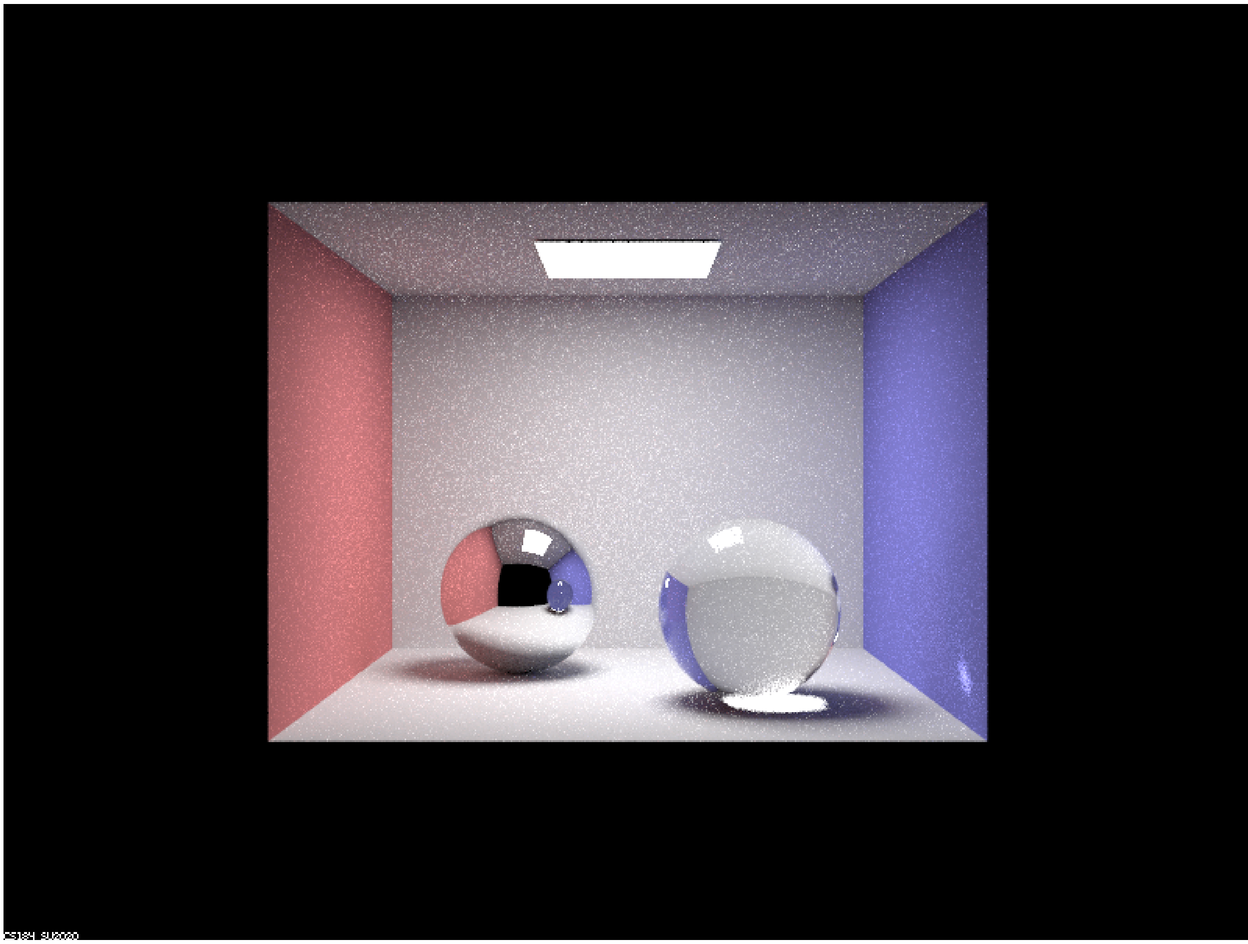
- 2:



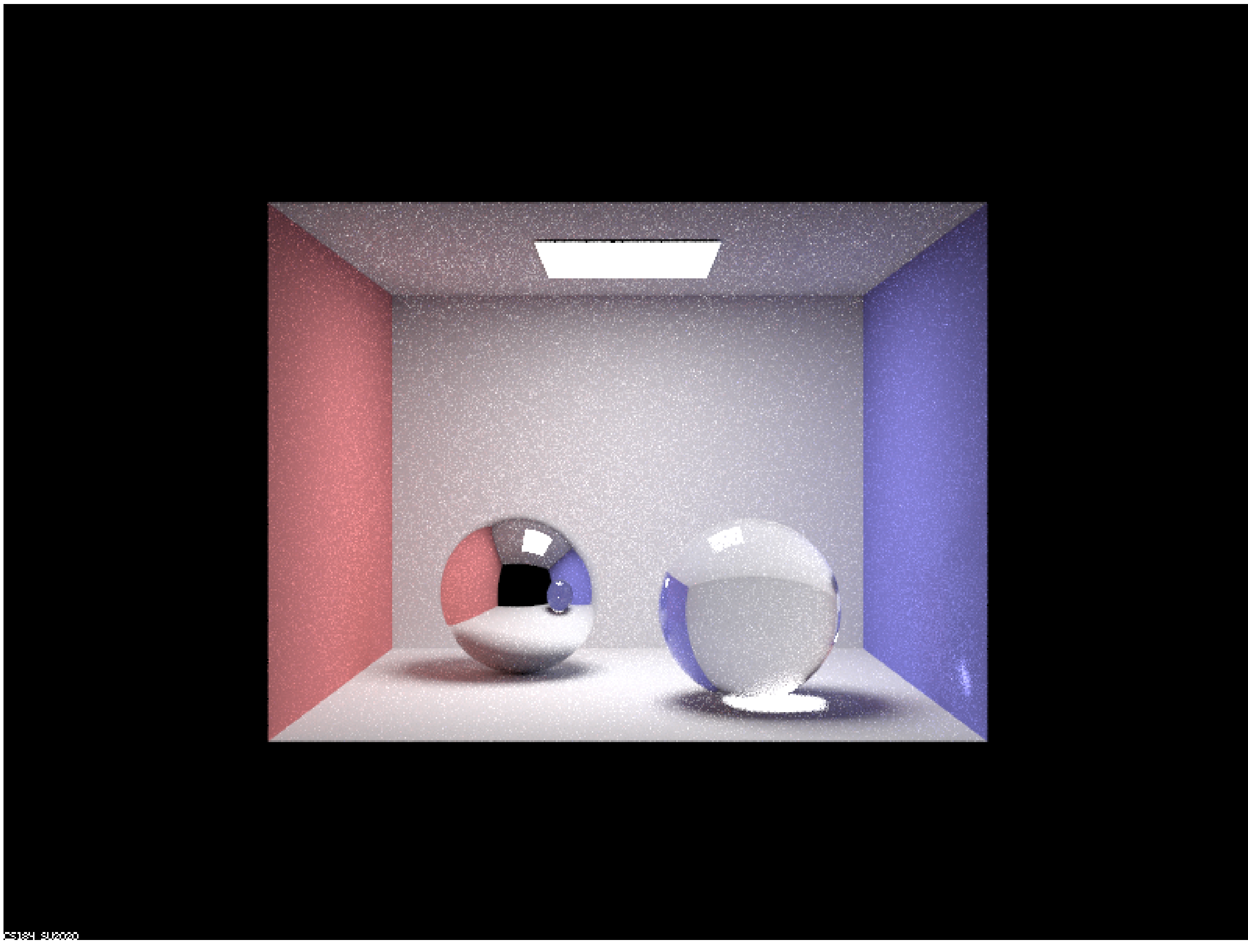
- 3:



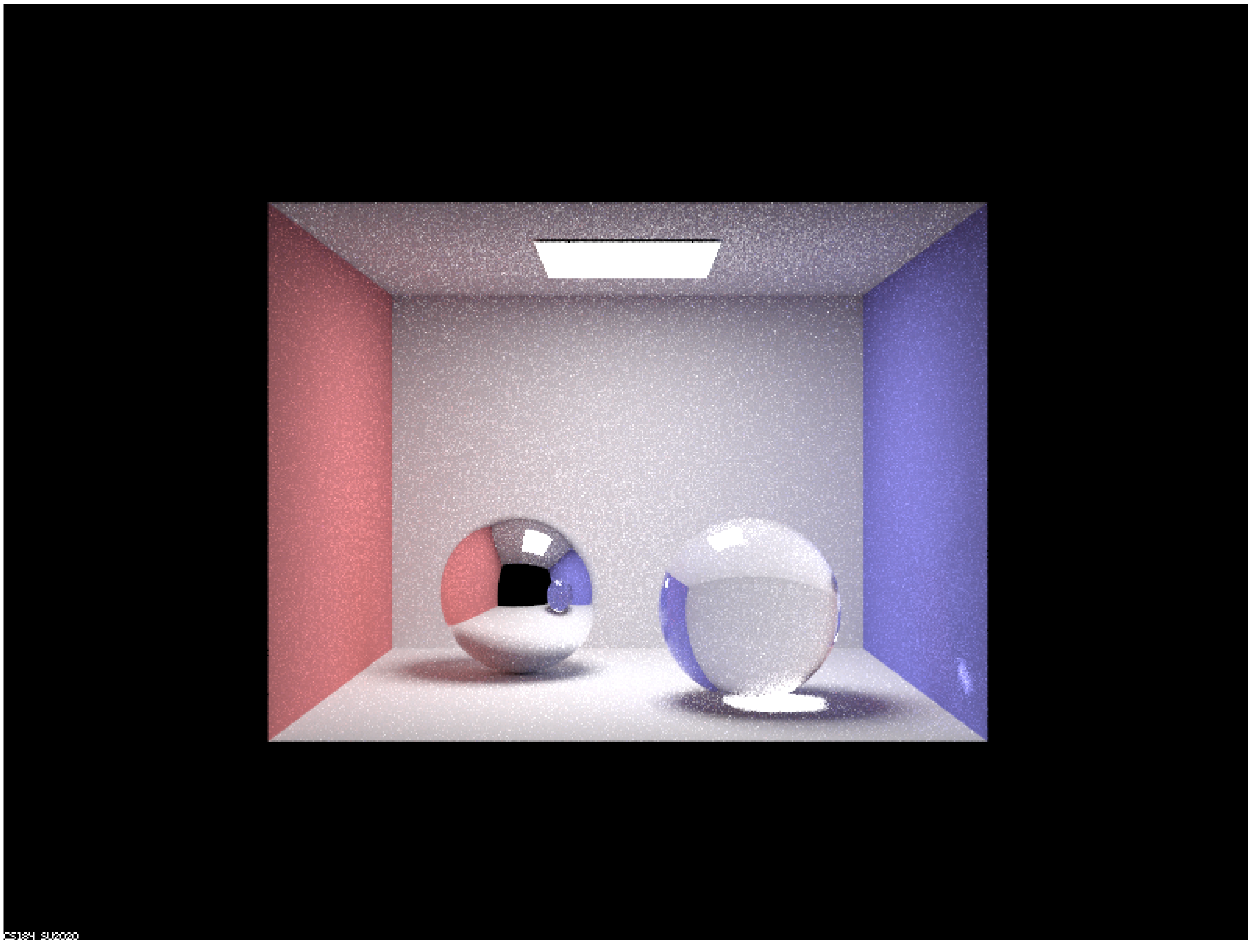
- 4:



- 5:



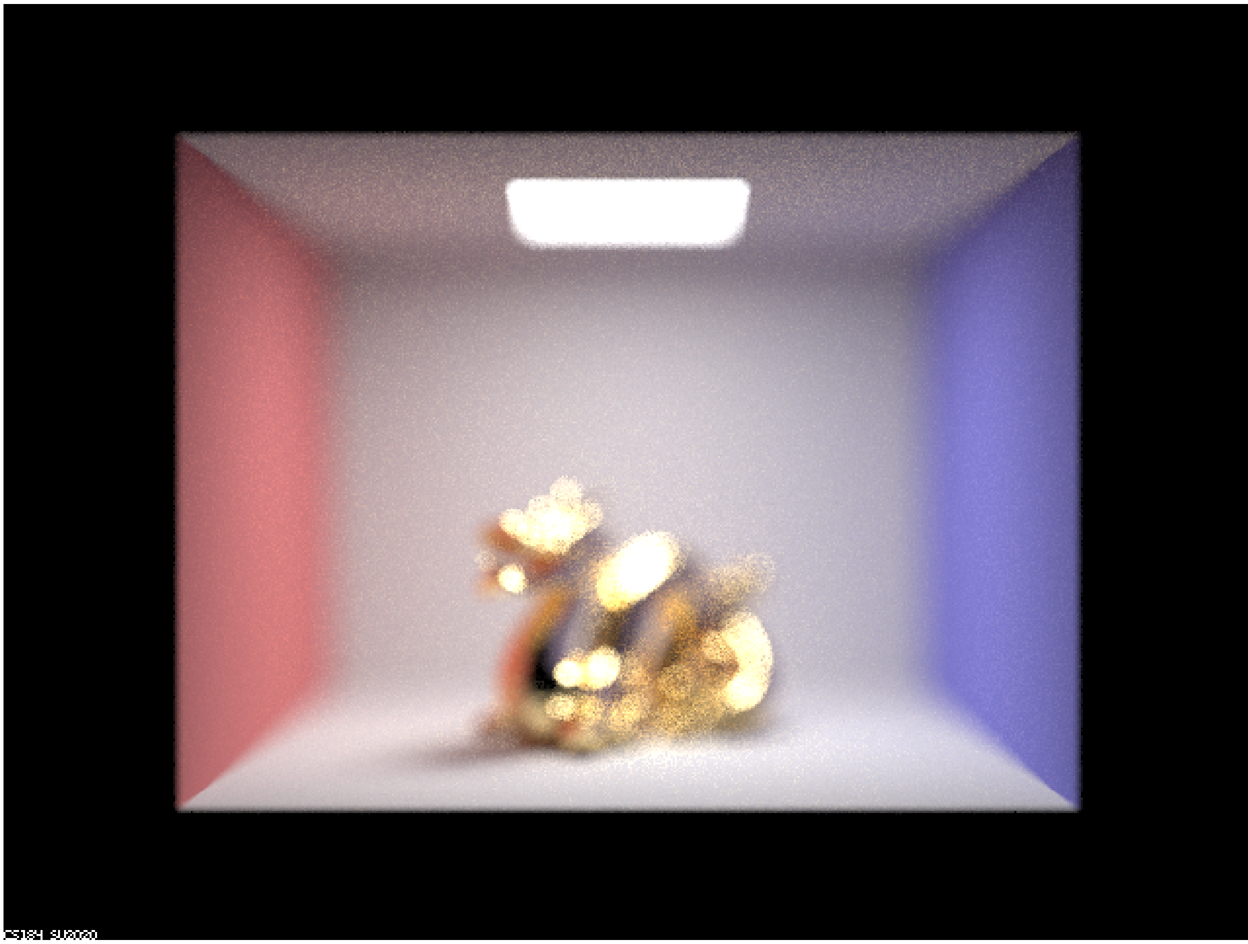
- 100:



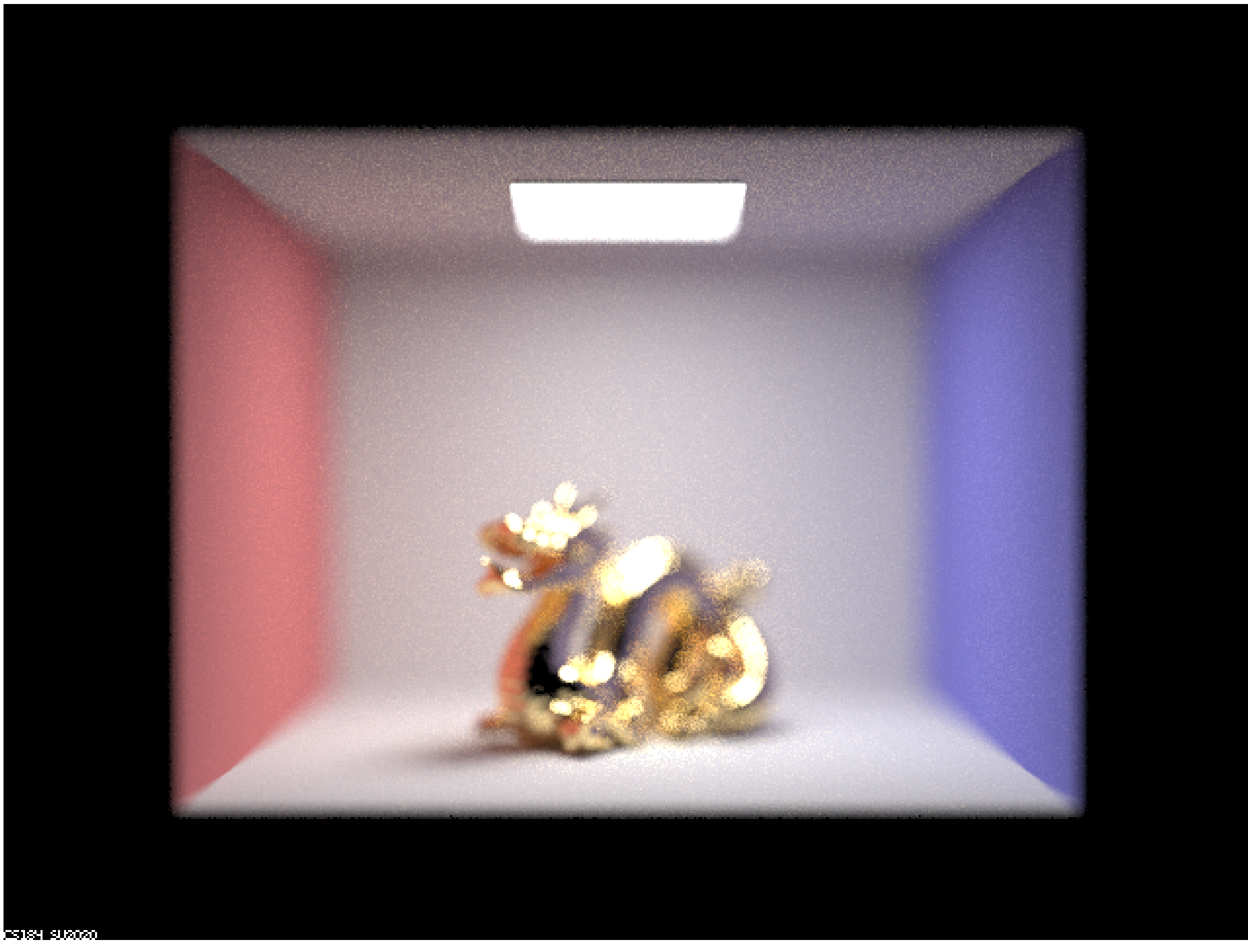
- Point out the new multibounce effects that appear in each image. Explain how these bounce numbers relate to the particular effects that appear. Make sure to include all screenshots.
    - At m = 0, we only see the light. At m = 1, we start to see objects directly illuminated by the light. At m = 2, we can see the mirror effect on the left ball. At m = 3, we can see that the right ball is lit up because the light now went through the ball. At m =4, we see the light go through the ball the hit the ground to make the glassy effect. At m = 5, we see that there is now light on the right wall due to refraction of the right glass ball. At m = 100, the image is brighter because of the subsequent bounces, but there are no big changes.

Part 4

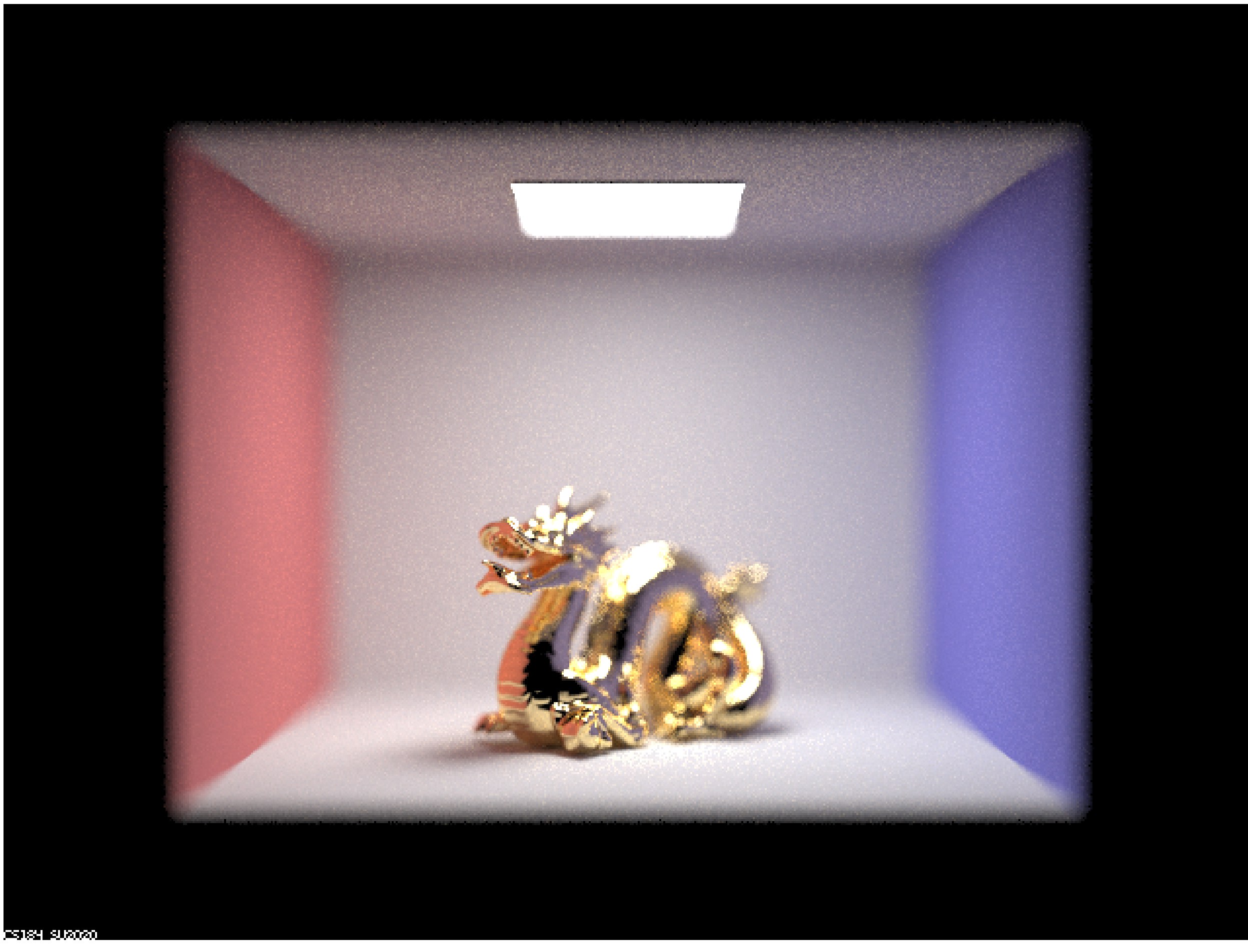
- Implementation:
  - I used project part 3-1's code to find out the direction of the generated ray and sampled the thin lens using rndR and rndTheta. After calculating pLens and pFocus using the samples and focal distance, I used the camera-to-world conversion and normalized the direction vector. Lastly, I created the ray and set the min and max of the ray to its respective nClip and fClip.
- In a few sentences, explain the differences between a pinhole camera model and a thin-lens camera model.
  - There are more rays of light coming from the same point through the lens than the pinhole camera. With a pinhole camera, you cannot selectively focus on one part of the image. With a thin-lens camera, you can change what you focus on depending on the lens radius and focal distance.
- Show a "focus stack" where you focus at 4 visibly different depths through a scene. Make sure to include all screenshots.
  - 4:



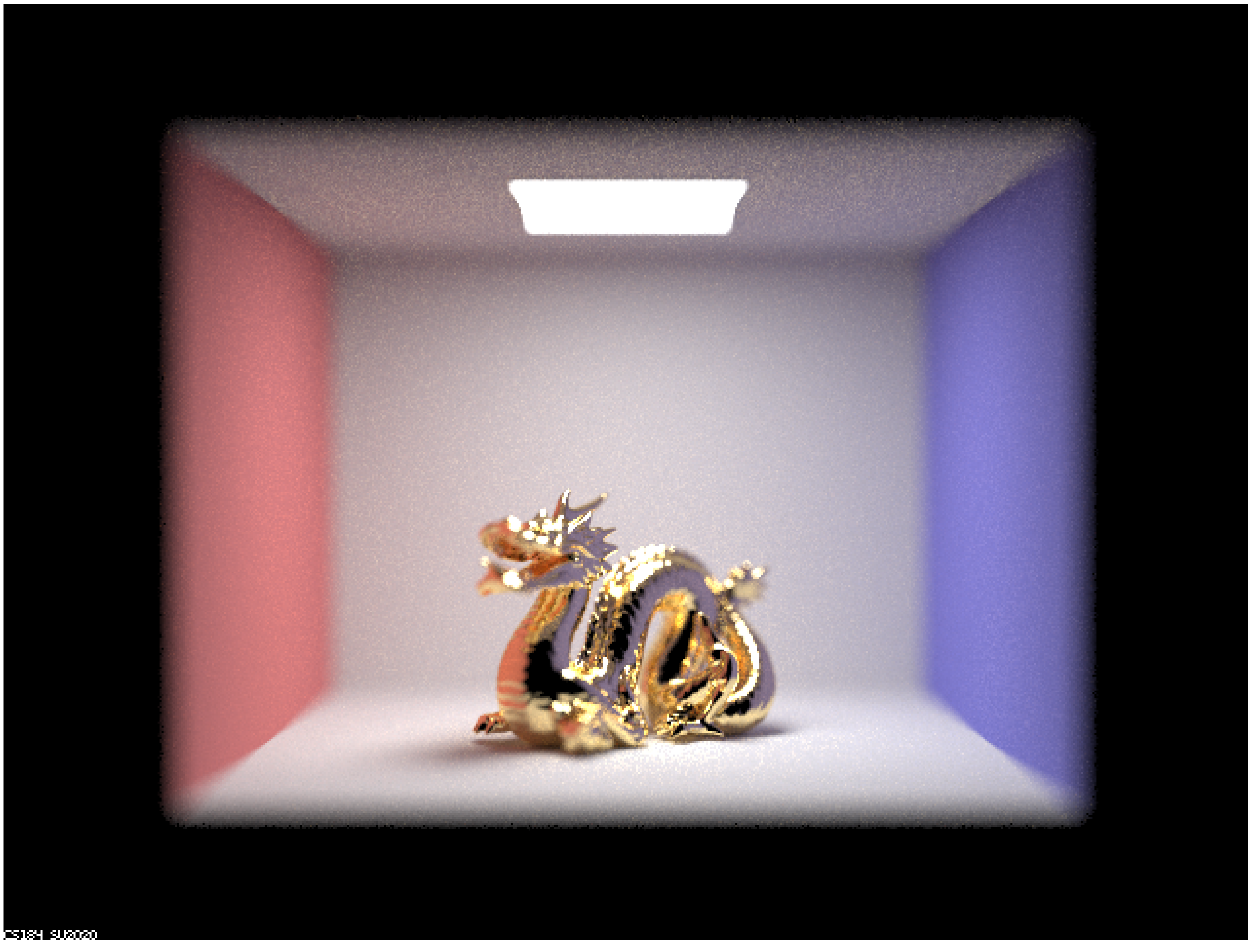
- 4.25:



- 4.5:

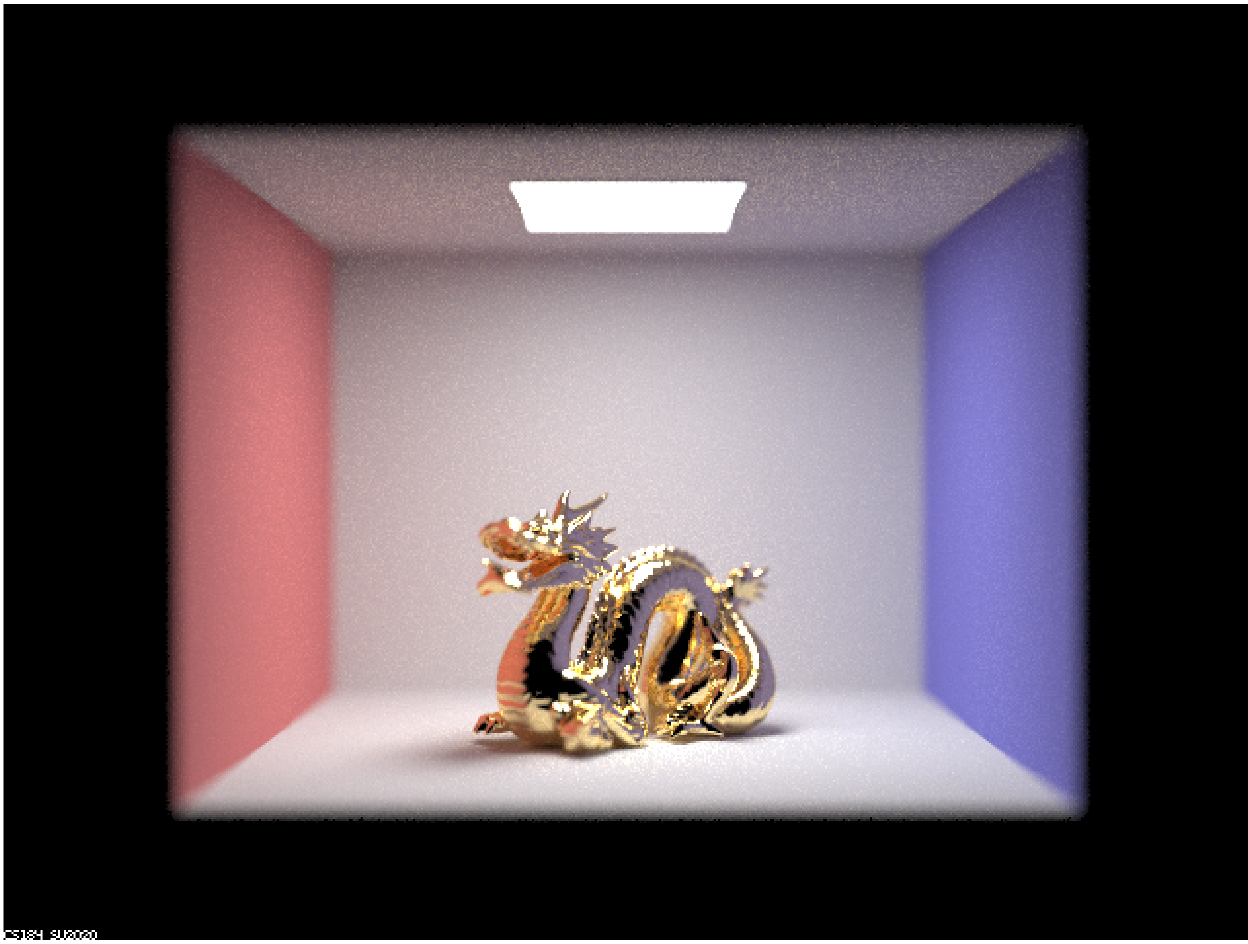


- 4.75:

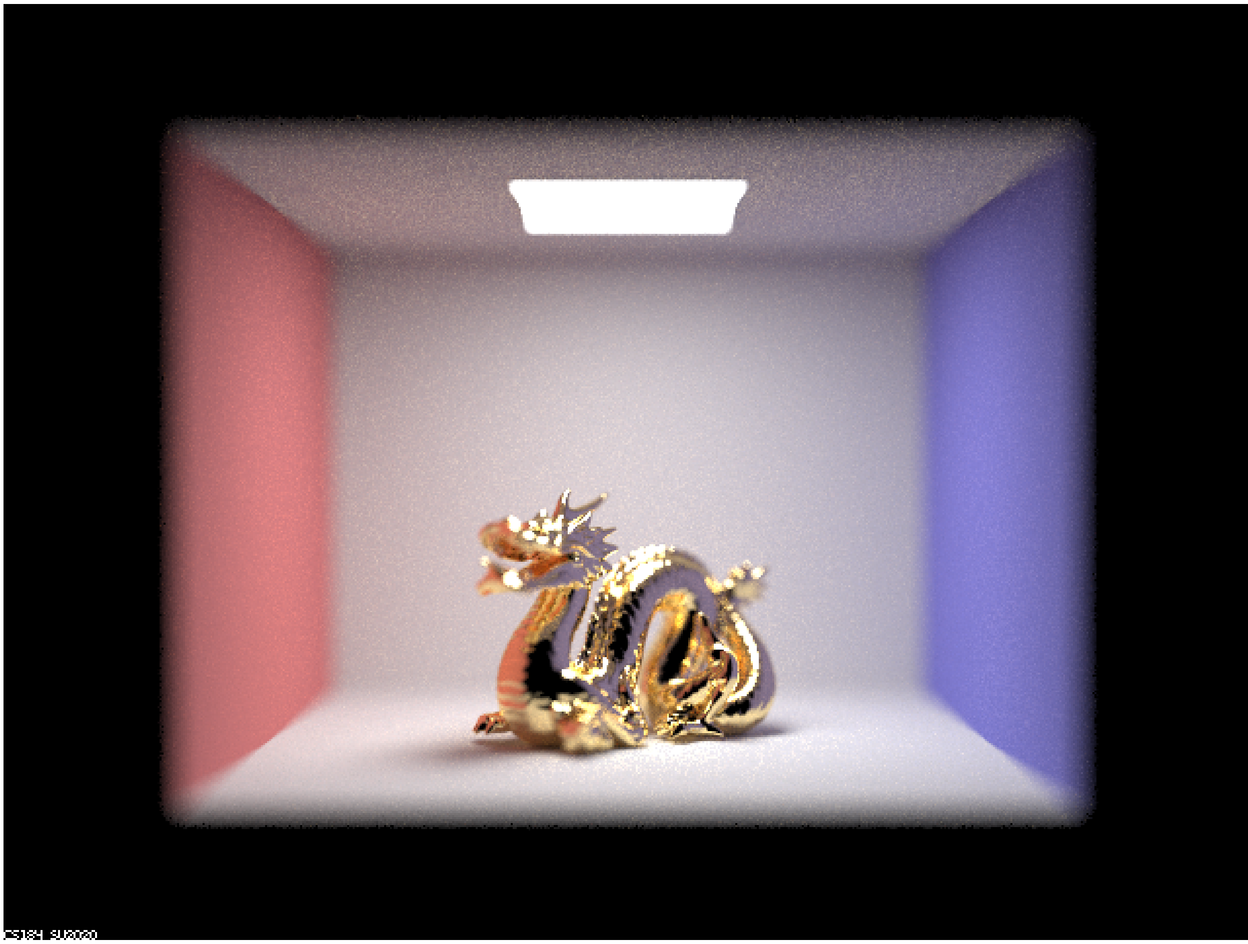


  - As we increased the focal distance, the dragon became more and clear as it focused on the dragon more. At 4.5 we were only able to focus on the bottom of the dragon.
- Show a sequence of 4 pictures with visibly different aperture sizes, all focused at the same point in a scene. Make sure to include all screenshots.

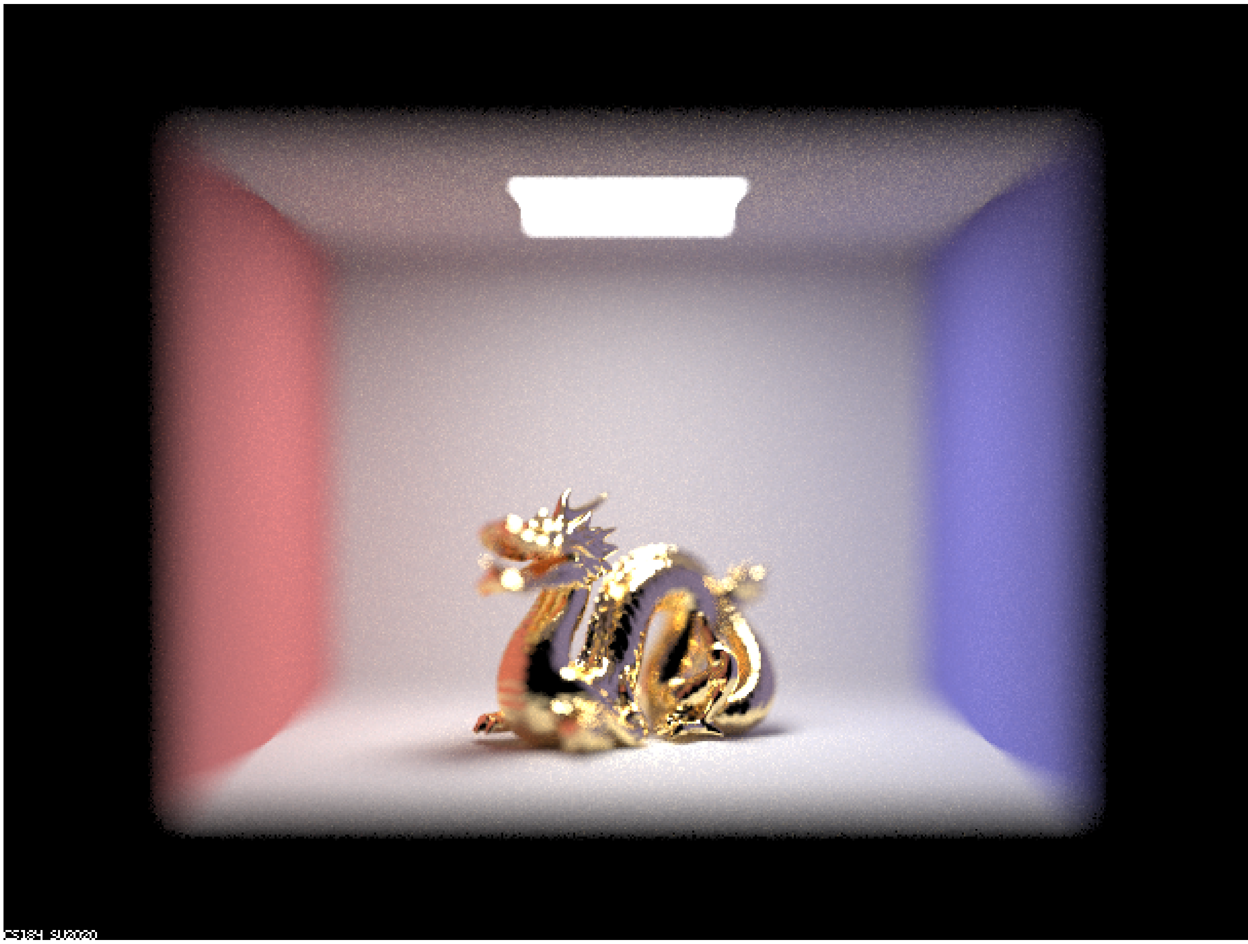
- 0.15:



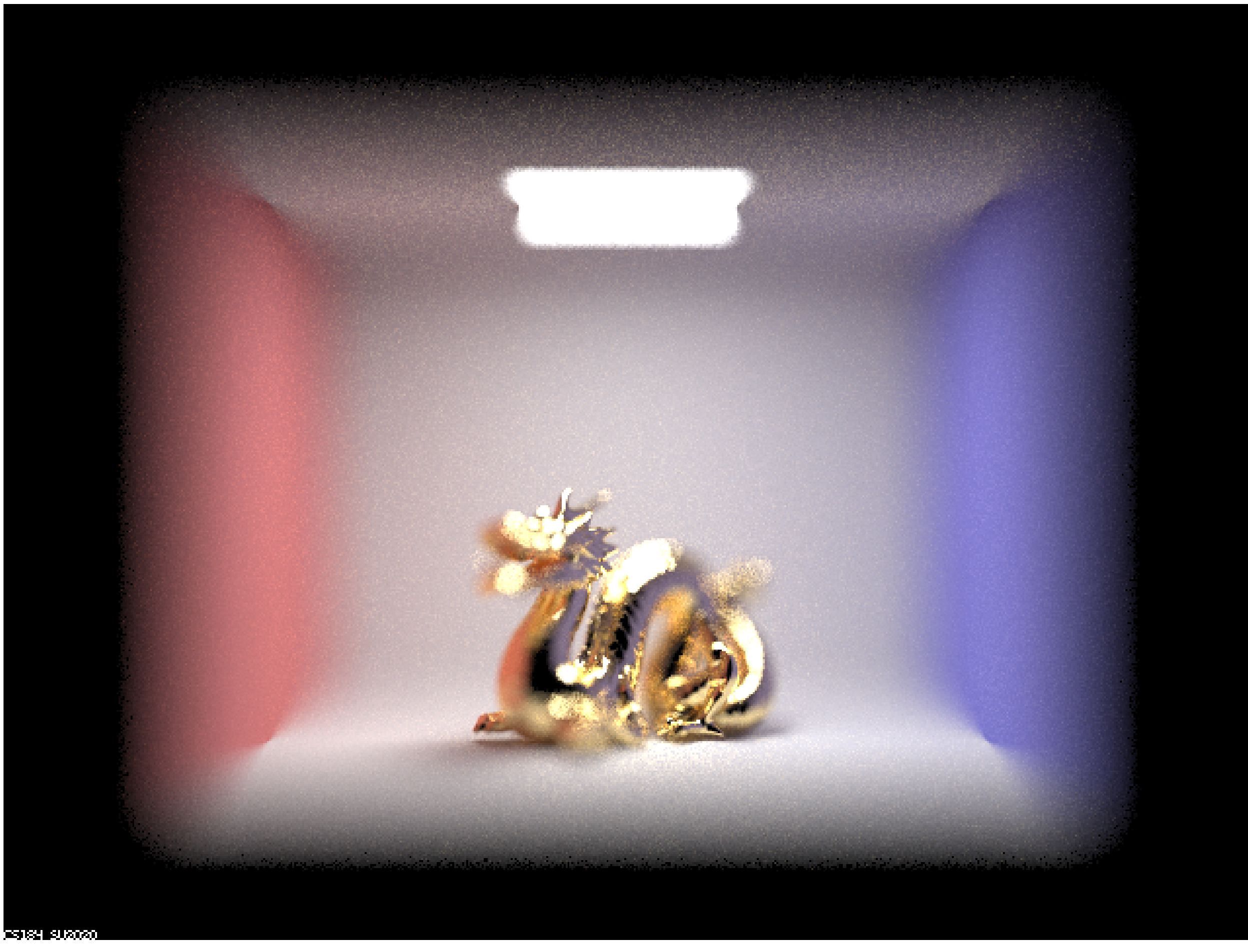
- 0.23:



- 0.35:



- 0.7:

  - As we increased the aperture/lens radius it became less and less focused. As we have a larger lens, the blurry area increases and the focused area decreases.

- At the end, if you worked with a partner, please write a short paragraph together for your final report that describes how you collaborated, how it went, and what you learned.
  - We did the coding part of this project together pretty quickly and we had to go together to campus in order to test our code together. We learned that we should not rely on local testing and that testing the projects may change for the worse because we had M1/M2 Mac computers. We also realized that rendering/testing on the hive takes much longer since we had to transfer between local and hive each time we made a change or wanted to look at the images.