**Project Report**

The project is fully working with both implicit and explicit light connections. The final outputted results can be seen in figures 1 through 5.

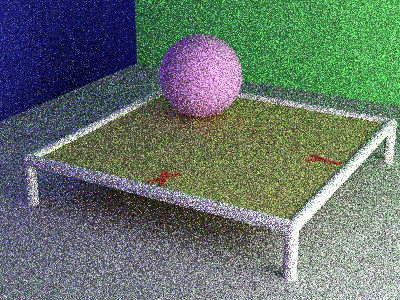
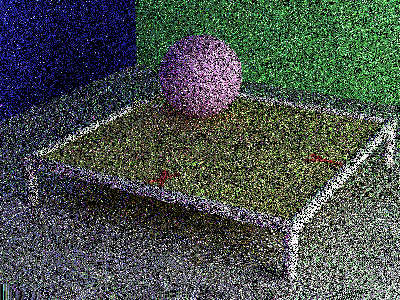


Figure 1: 1 Pass Figure 2: 8 Passes

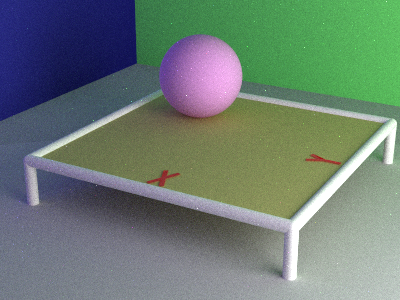
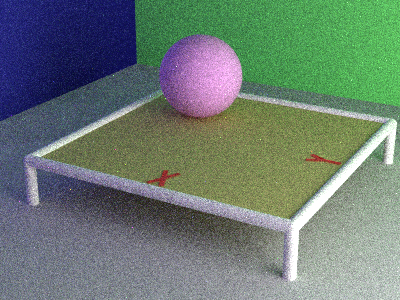


Figure 3: 64 Passes Figure 4: 512 Passes

To confirm that the implicit light connections are working as intended I commented out the explicit connections and outputted the results, which you can see in figures \_\_\_\_\_\_\_\_. I did this after finishing the path tracer to double check that the explicit connections were correct but I also did this when debugging. One of the things that I checked when trying to find the source or bugs was to comment out the explicit connections and see if the bug still existed. Every time I did this the implicit only output looked identical to the explicit and implicit but converging at a slower rate, just as would be expected. This would eliminate the possibility of the bug existing exclusively in the explicit connection part of the code.

Other than testing implicit light connections only I also used a few other methods to find the bugs in my code. The main issue that I ran into when coding was a bug where the image looked as expected except for the whole image’s brightness being too low. I though this might be due to the image viewer I was using however I also noticed that the image was converging too slowly as well so there had to be something wrong with the code. There was also a strange shadow that appeared on one quadrant of the table if the same point function was correctly implemented. The too dark image, with much too slow convergence can be seen in figure \_\_\_\_\_\_\_ and the strange shadow can be seen in figure \_\_\_\_\_\_\_\_\_\_\_\_\_\_. I added debugging output to log the statistics of the program and realized that about 90% of the rays being traced into the scene were returning without finding any color to add to the image. The issue was fixed after normalizing the omega I values, which decreased the miss rate of the rays down to about 25%.

The program still has the functionality of being able to move the camera. This allows for testing cases that cant be seen in the default view. For instance, it allows for making sure that lights in the frame are rendered properly. It also allows for making sure that the back walls and ceiling are rendering correctly. This can be seen in figure \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.