Q1)

How do you compile and run your code? Specify which OS you have tested on.

Answer: I compile and run my code through ubuntu (I have a Windows laptop/os and I use ubuntu/WSL.) To compile and run my code, I go to the build folder in assignment 4 and run `make` (if the build is empty, I first run `cmake ..` within the ./assignment4/build area). In order to view my demo, you can run the code given at the bottom of the page.

- You can also toggle `-jitter` to allow for 20 super samples from a uniform distribution within each pixel. For example,

  ```
  ./assignment4 -input scene01_plane.txt -output 01_antialias.png -size 800 800 -jitter
  ```

- You can also toggle `-filter` to weight each sample by its gaussian and take a weighted average.
- Also there are 3 other assets that include fog. These are:
  - `./assignment4 -input scene08_bunny_fog.txt -output 06_fog.png -size 800 800 -bounces 4` which is the scene 6 bunny but with pink fog
  - `./assignment4 -input scene09_arch_fog.txt -output 07_fog.png -size 800 800 -bounces 4 -shadows` which is the scene 7 arch but with orange fog.
  - `./assignment4 -input scene10_arch_foreshortened.txt -output 06_fog2.png -size 800 800 -bounces 4 -shadows` which is the scene 7 arch but I made the fog around 10 times longer and adjusted the camera angle so the arch goes from foreground to background in order to better visualize the fog.

Within the fog scene files, the fog is depicted with a `Fog` keyword and three struct elements:

- `color`: the color of the fog
- `density`: a float representing the attenuation of the fog. The fog is calculated with exponential attenuation, so the amount of original color is $e^{-\texttt{density}\cdot\texttt{dist\_to\_camera}}$
- `opacity`: the maximum opacity of the fog color (so `opacity` = 0.25 means 25% fog color). I basically cap the attenuation at $1 - \texttt{opacity}$.

Q2)

Did you collaborate with anyone in the class? If so, let us know who you talked to and what sort of help you gave or received.

Answer: Not as much. I mainly coded in the same location as Alicia Lin, we didn't really talk much though.

Q3)

Were there any references (books, papers, websites, etc.) that you found particularly helpful for completing your assignment? Please provide a list. In particular, mention if you borrowed the model(s) used as your artifact from somewhere.

Answer: I mainly used stack overflow and the GLM documentation. I also used ChatGPT to find approaches to debugging.

Q4)

Are there any known problems with your code? If so, please provide a list and, if possible, describe what you think the cause is and how you might fix them if you had more time or motivation. This is very important, as we're much more likely to assign partial credit if you help us understand what's going on.

Answer: I believe I have satisfied all the requirements. One thing is that when I rendered scene 3 multiple times, occasionally the entire background is black (but the object is fine), but it's not an easily or deterministically replicable bug, so I'm not super sure if it's a problem with my code or something else.

Q5)

Did you do any of the extra credit? If so, let us know how to use the additional features. If there was a substantial amount of work involved, describe how you did it.

Answer: Yes. I added a simple fog obeying Beer-Lambert's Law (light through a medium such as fog has exponential attenuation). I also added antialiasing through supersampling toggleable through `-jitter` and a gaussian filter toggleable through `-filter`. I include images of the examples including and without the bonus features in the pngs in the root folder.

Q6)

Do you have any comments about this assignment that you'd like to share? We know this was a tough one, but did you learn a lot from it? Or was it overwhelming.

Answer: This was fun! One thing was in step 2, it said "You must implement `TraceRay()`" which made me think I had to fully implement TraceRay with recursion and shadows and BRDF at once with no guideance.... Took an embarrassing amount of time to notice. Also, I confirmed with classmates, but I wasn't super sure that the material's diffuse/specular/ambient colors were the k_a, k_d, k_s in the equation. I had the misconception that they were global constants instead of parameters of the material, and I think it'd be helpful if the lecture mentioned this.