CS 465 Prelim 1

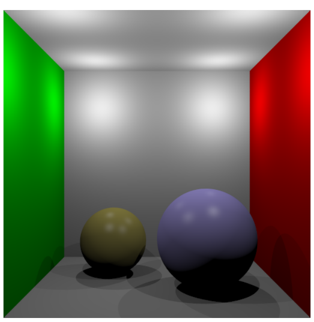
Tuesday 4 October 2005—1.5 hours

Problem 1: Image formats (18 pts)

1. Give a common pixel data format that uses up the following numbers of bits per pixel: 8, 16, 32, 36. For instance, given 24 bits per pixel you might answer "8 bits for each of red, green, and blue."

2. Explain an advantage of a display system that uses nonlinear intensity quantization (that is, one that requires gamma correction) over a display that uses linear quantization, given that the purpose of the display is to show images to a human viewer.

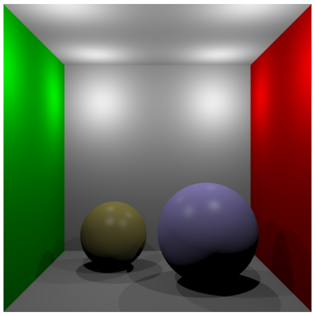
Problem 2: Ray tracing bugs (36 pts)

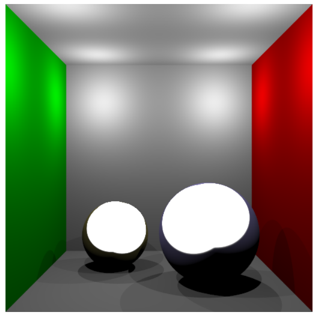


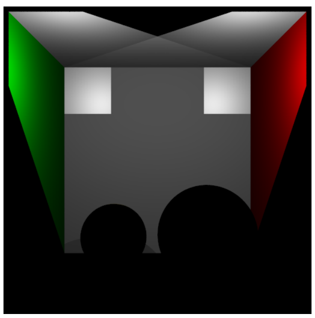
Here is a correct ray-traced rendering of a scene consisting of a five-sided cube containing two spheres lit by four lights near the top corners of the box. The size of the cube is two units on a side.

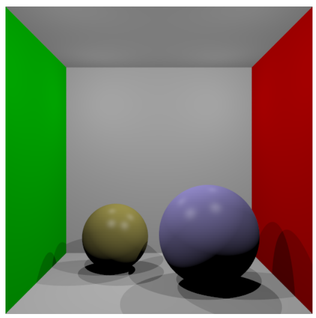
We modified the ray tracer to introduce the bugs described on the next page. Match the bugs with the corresponding incorrect outputs. For each case, give a brief explanation of how the bug leads to the visible changes in the image.

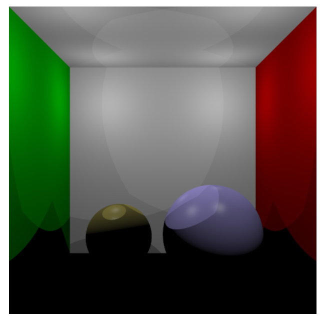
For example, the image labeled "Example" matches the bug "Failing to normalize the half vector in Blinn-Phong shading." A suitable explanation is "If the half vector is not normalized its length will sometimes be greater than 1. This can result in the dot product with the normal being greater than 1, and when this number is taken to a large power it becomes very large, leading to the very bright areas on the spheres."

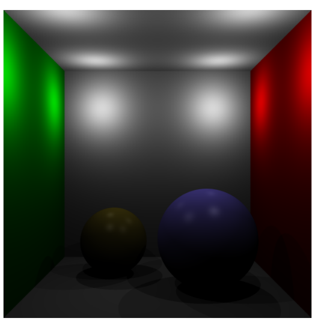

Correct

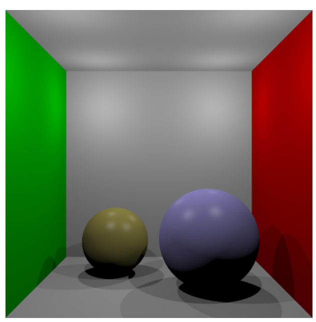

Example

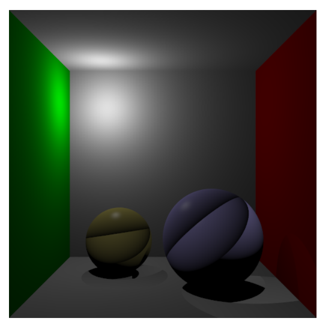

A


B


C


D


E


F

0. *Example*: Failing to normalize the half vector in Blinn-Phong shading
1. Omitting gamma correction
2. Overwriting the shading result for each light source, rather than adding
3. Omitting the $1/r^2$ falloff for light sources
4. Assuming shadow rays' directions are always normalized when in fact they are not
5. Forgetting to normalize the light vector in shading computations
6. Failing to end shadow rays at the light source

Don't forget the explanations!

Problem 3: Sampling and reconstruction (22 pts)

1. Suppose you are using a reconstruction filter for translating images around in the plane without changing their size, and it performs well for that application. For each of the following other applications, will it be necessary to make the filter larger, make it smaller, or keep it the same size relative to the input pixel grid?

   (a) Reducing the image (downsampling)

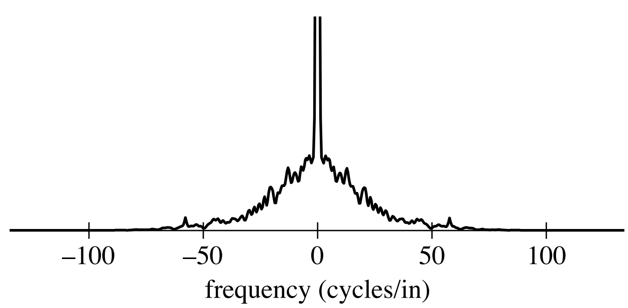
   (b) Enlarging the image (upsampling)

2. Suppose I am antialiasing an image in a ray tracer by supersampling it using a particular filter. Despite my using that filter, the image still has stair-steps on the edges of objects.

   (a) In what way do you need to change the filter to reduce this problem?

   (b) In what way will the image be degraded if you go too far?

3. Suppose you are scanning a photographic print. The intensity of the photograph along any horizontal or vertical line has a Fourier transform that looks about like this:



   Your scanner can operate at various resolutions, but it always point samples the image. Should you use 50, 100, or 200 samples per inch to achieve the most compact result without introducing aliasing artifacts? How can you tell?

Problem 4: Compositing (24 pts)

Suppose we have an image of a gray elephant, with an alpha matte to delineate foreground from background. The image is stored with pre-multiplied alpha.

1. If we accidentally use the image in a program that expects non-premultiplied alpha, will the edges come out too dark, about right, or too light if the background is:

   (a) solid black

   (b) solid white

   (c) about the same color as the elephant

   Explain why by writing down the equations for the correct and incorrect results.

2. If we do the compositing correctly but overlay the image twice in the same place, will the edges come out too dark, about right, or too light if the background is:

   (a) solid black

   (b) solid white

   (c) about the same color as the elephant

   Explain why informally. There is no need to write down the equations for this part (we did not find them all that helpful).

Assume in all cases we are using the **over** operation.