Computer Graphics	Prof. Brian Curless
CSE 591	Spring 2002

Homework #1

Assigned: Wednesday, May 1, 2002
Due: Wednesday, May 15, 2002

Modeler artifact due: Friday, May 3, 2002 (see problem #0)

Directions:

Please provide short written answers to the questions in the space provided. If you require extra space, you may staple additional pages to the back of your assignment. You should work on the assignments ***individually***. Feel free to contact the instructor for clarifications.

0. Modeler artifact

Your modeler artifact is due on Friday, May 3, by midnight. The artifact should consist of the binary, a snapshot of your model in an exciting pose (or an assembly of poses), and a short statement highlighting what, if anything, people should look out for when running your model. If your binary is the same as the one you turned in for project grading, then you don't need to turn it in again. We'll just use the same binary.

1. Short answers

Provide a short answer (typically one or two sentences) to each of the following questions. In each case, you must clearly justify your answer.

a. How are the electrons coming out of the red, green, and blue electron guns of a color monitor different?

b. Can a gradient magnitude image be computed purely as a convolution operation?

c. If you convolve an image with the Laplacian filter, will you typically get an image that looks about the same except a bit sharper?

d. Using BSP trees for hidden surface elimination, does it take extra effort to render a scene after moving the viewpoint?

e. Using BSP trees for hidden surface elimination, does it take extra effort to render a scene after moving an object in the scene?

2. Image processing

a. Describe the effect of each of the following filters. You may ignore effects of performing convolution around the boundary of an image. In addition, indicate which filter will cause the most blurring and which will produce the brightest image. Justify your answers.

0	0	0
0	1	0
0	0	0

0	0	0
1/2	0	1/2
0	0	0

0.11	0.11	0.11
0.11	0.11	0.11
0.11	0.11	0.11

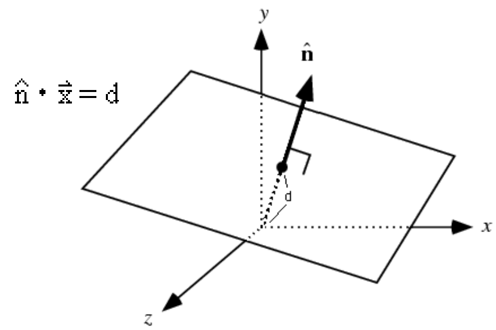
0	0.2	0
0.2	0.4	0.2
0	0.2	0

0	-1	0
0	3	0
0	-1	0

0	0	1/3
0	1/3	0
1/3	0	0

3. Affine transformations

A plane can be described by the following equation and figure:



where $\hat{n}$ is the normal to the plane and d is the closest distance between the origin and the plane (which happens to be along the same direction as the normal).

Consider a plane with a normal vector of (cos θ, sin θ, 0). The equation for this plane can be simplified to:

$$x \cos \theta + y \sin \theta = d$$

Write out the product of 4x4 matrices that would perform a reflection across this plane. You must write out the contents of each matrix. Do not multiply the matrices out. You may write out the final product symbolically, after you have defined which symbol corresponds to which matrix. Show your work.

4. Perspective projections

a. Now recall the transformation matrix for perspective projection:

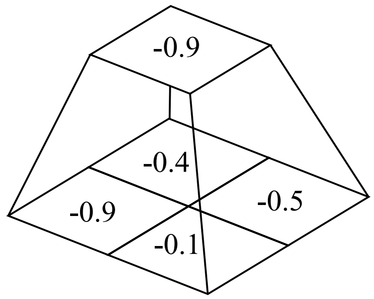
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1/d & 0 \end{bmatrix}$$

If we first apply a uniform scale to all points before performing the projection indicated by this matrix, how will the projected points be different from the case where no scale is applied? Show your reasoning mathematically and explain the result geometrically (in words and/or with a sketch).

b. Using the perspective projection matrix above, we observed in class that any set of parallel lines will converge and intersect at a vanishing point. In fact, some parallel lines actually will ***not*** intersect, i.e., they will remain parallel, even after undergoing this perspective projection. Which lines are these? Geometrically speaking, what do all sets of these lines have in common?

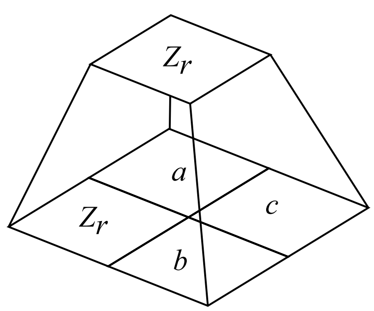
5. Hidden surface elimination

The Z-buffer algorithm can be improved by using an image space "Z-pyramid." The basic idea of the Z-pyramid is to use the original Z-buffer as the finest level in the pyramid, and then combine four Z-values at each level into one Z-value at the next coarser level by choosing the farthest (most negative) Z from the observer. Every entry in the pyramid therefore represents the farthest (most negative) Z for a square area of the Z-buffer. A Z-pyramid for a single 2x2 image is shown below:



a. At the coarsest level of the pyramid there is just a single Z value. What does that Z value represent?

Suppose we wish to test the visibility of a polygon **P**. Let Z_p be the nearest (most positive) Z value of polygon **P**. Let **R** be the smallest region in the Z-pyramid that *completely* covers polygon **P**, and let Z_r be the Z value that is associated with region **R** in the Z-pyramid.



where $a, b, c > Z_r$

5. Hidden surface elimination (cont'd)

b. What can we conclude if $Z_p < Z_r$?

c. What can we conclude if $Z_p > Z_r$?

d. If the visibility test is inconclusive, then the algorithm applies the same test recursively: it goes to the next finer level of the pyramid, where the region **R** is divided into four quadrants, and attempts to prove that polygon **P** is hidden in each of the quadrants **R** of that **P** intersects. Since it is expensive to compute the closest Z value of **P** within each quadrant, the algorithm just uses the same Z_p (the nearest Z of the *entire* polygon) in making the comparison in every quadrant. If at the bottom of the pyramid the test is still inconclusive, the algorithm resorts to ordinary Z-buffered scan conversion to resolve visibility.

Suppose that, instead of using the above algorithm, we decided to go to the expense of computing the closest Z value of **P** within each quadrant, and that the finest level of the pyramid has one pixel per region/quadrant. Would it then be possible to always make a definitive conclusion about the visibility of **P** within each pixel, without resorting to scan conversion? Why or why not?

6. Shading

The Phong shading model can be summarized by the following equation:

$$I_{phong} = k_e + k_a I_a + \sum_i \frac{1}{a + bd_i + cd_i^2} I_{l_i} \left[k_d \left(\mathbf{N} \cdot \mathbf{L}_i \right)_+ + k_s \left(\mathbf{V} \cdot \mathbf{R}_i \right)_+^{n_s} \right]$$

where the summation *i* is taken over all light sources. The variables used in the Phong shading equation are summarized below:

I_{phong} a b c d_i k_e k_a k_d k_s n_s I_a I_{li} $\mathbf{L}_i$ $\mathbf{R}_i$ $\mathbf{N}$ $\mathbf{V}$

You may assume that all light sources are point sources, and the surface is viewed with a pinhole camera model (i.e., perspectively).

a. In general, which of the quantities above are affected if...

- …the viewing direction changes?

- …the position of the i^{th} light changes?

- … the orientation of the surface changes? Assume that the change in orientation is about the point being viewed.

- …the position (but not the orientation) of the point being viewed changes?

b. Given a single light source, describe the relationships between **N**, **L**, and **V** that would result in a point shaded with the Phong model appearing maximally bright.