Practice Midterm
CMSC 427
Distributed Thursday, March 10, 2005
Midterm: Thursday, March 17, 2005

General Guidelines: The midterm will focus on topics that have been discussed in class. I will not ask questions about material from the book that has not also been discussed in class or in a problem set. However, there will be no questions about OpenGL or programming problems.

The midterm will be closed book, with no notes or calculators. This means that problems should not require the use of a calculator; if you feel you need one when you are doing a problem this probably means there is an easier way to do the problem.

Topics and things to know about them:

1. Geometry (needed for problems on the next two topics)
    a. How to take an inner product, and what this does.
    b. How to take a cross product, and what this does.
    c. What a unit vector is, and how to find the magnitude of a vector.
    d. How to write the equations for a 3D line, or for a plane in 3D.
2. Transformations
    a. How to write rotation about the *x*, *y*, or *z* axis as a matrix.
    b. How to write translation as a matrix.
    c. How to combine these. For example, rotate about a point not at the origin.
    d. How to rotate or translate a line.
    e. How to relate the rows of a rotation matrix to the axes of a new coordinate system
        This includes writing the coordinates of points relative to a new coordinate system.
3. Projection
    a. Perspective projection
        i. How to find the image coordinates of a 3D point
        ii. Some consequences of that, such as properties of vanishing points, knowing when a line in 3D does not project to a line in the image, …..
    b. Orthographic projection
        i. How to find the image coordinates of a 3D point.
        ii. Consequences of orthographic projection, such as that it preserves parallelism and ratios of areas.
4. Discretization
    a. How to represent a line with pixels.
        i. Naïve algorithm
        ii. DDA algorithm
    b. How to fill a polygon.
        i. How to tell if a point is inside a polygon.
            1. Implicit function test.

        2. Crossing number test.
    ii. Flood fill algorithm.
    iii. Scan line algorithm
5. Sampling and Aliasing
    a. Creating an image by sampling
    b. Aliasing – what it means, what causes it.
        i. Spatial
        ii. Temporal
    c. Anti-aliasing
        i. Smoothing
        ii. Supersampling and smoothing
6. Texture
    a. Perlin noise algorithm
    b. A basic idea of how Efros and Leung's texture synthesis algorithm works
7. Color
    a. Superposition
    b. Color Spaces – CIE, RGB, CMYK, HSV.

Sample Problems (Note, some of these problems may be a bit more involved than ones I'd ask on a time-limited exam).

1. Create a matrix that rotates points 90 degrees about the point (1,1).
2. What is the distance from (3,2) to (7,5) in the direction (1,2)?
3. Provide any two rows of a 4x4 matrix that will transform 3D points as they would appear in the coordinate system of a viewer centered at (3,2,0) facing in the direction (1,2,3). The new *z* coordinate should describe the distance from the viewer to a point, in the direction that they are viewing it.
4. Suppose you have a camera with a focal point at (0,0,0) and an image plane of z=1. Give an example of a right triangle in 3D that will also appear in the image as a right triangle, assuming perspective projection. Do the same thing for orthographic projection.
5. Prove that parallel lines in the world do not always appear as parallel lines with perspective projection.
6. Explain how you would adapt DDA to discretely represent the boundary of a circle. What pixels would be filled in to represent a circle centered at (20,20) with a radius of 5?
7. Suppose you have an image with vertical, black and white stripes that are five pixels wide, and you want to shrink this image to half its current size in each dimension. Is there a problem with aliasing? If not, why not? If yes, how would you prevent this by smoothing using averaging? How much would you need to smooth? There may not be a simple, exact answer to this question, so just think about it and describe the consequences of different possible choices.
8. Suppose we apply the Efros and Leung texture synthesis algorithm to an image consisting of vertical black and white stripes. What do you think we might get? How might this depend on the parameters that we use?

9. If a pixel has RGB values of (.7, .3, .2), how would you describe its HSV values? If a pixel has a hue of green, a saturation of .5, and a value of .4, describe it with RGB values.
10. Suppose you had a monitor that emitted light that was either cyan, magenta or yellow. How could you use this to create white light?