CS 4620 Final Exam

Tuesday 16 December 2008—2.5 hours

Problem 1: Ray Tracing

The following pseudocode intersects a ray with some geometric shape.

```
function surfaceIntersect(Ray r)
   Point3 p = r.origin;
   Vector3 d = r.direction;
   p.x = 2 * p.x;
   d.x = 2 * d.x;
   [r1, r2] = quadraticRoots(d.x*d.x + d.y*d.y,
                             2*(p.x*d.x + p.y*d.y),
                             p.x*p.x + p.y*p.y - 1);
   t1 = -p.z/d.z;
   t2 = (1 - p.z)/d.z;
   tmin = max(min(r1, r2), min(t1, t2));
   tmax = min(max(r1, r2), max(t1, t2));
   if (tmin < tmax) {
      if (tmin > 0) return tmin;
      if (tmax > 0) return tmax;
      }
   return INFINITY;
```

The function `quadraticRoots` returns the two roots of a quadratic with the given coefficients, if there are two roots.

1. What is the shape? Give a detailed and precise definition, including all dimensions.
2. When there are no roots, what values could `quadraticRoots` return for `r1` and `r2` that will make this code work correctly without changes?
3. Give an example of a hit with `d.z == 0` and explain what values the variables in this function take on and how the return value arises.

Problem 2: Transformation matrices

Classify each of the following 2D homogeneous matrices as follows: (a) rotation, (b) mirror reflection, (c) uniform scale, (d) nonuniform scale, (e) translation, (f) shear, or (g) combination.

Earlier categories take precedence over later categories if more than one applies. There is not a one-to-one mapping between the matrices and the categories.

1. $\begin{bmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

2. $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

3. $\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

4. $\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

5. $\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

6. $\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

7. $\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Problem 3: Image resampling

Suppose we are upsampling a grayscale image whose contents look like this:

$\vdots$

	0	1	0	1	0	1	
	1	0	1	0	1	0	
$\cdots$	0	1	0	1	0	1	$\cdots$
	1	0	1	0	1	0	
	0	1	0	1	0	1	
	1	0	1	0	1	0	

$\vdots$

and one of the output samples lands (1) exactly on a zero input pixel, or (2) exactly halfway between two rows and two columns of pixels (that is, equidistant from the four nearest input pixels). What are the values of that pixel if we are using (a) bilinear interpolation or (b) the B-spline filter?

Note: The B-spline reconstruction filter is the same piecewise cubic function as the spline basis function of the same name, so you can evaluate it using its spline matrix:

$$M_{\text{bs}} = \frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}$$

(This problem has four parts: 1a, 1b, 2a, and 2b.)

Problem 4: Pipeline

Consider an equilateral triangle in the x-z plane, with one vertex at $(1, 0, 0)$ and the other two also located one unit from the origin. It is viewed by a camera looking at the origin from the viewpoint $(0, 2, 0)$, and illuminated by a point light located at $(x, 2, 0)$. It is shaded with the Blinn-Phong illumination model, with $k_d = 0$, $k_s = 1$, and $n = 25$. Shading is being computed with local light and viewer, using either per-vertex or per-fragment shading. The normals at all three vertices point in the y direction.

1. Find a value of x for which fragment shading obviously computes a brighter value at the center of the image than vertex shading, and estimate the ratio by which it's higher.

2. Find a value of x for which vertex shading obviously computes a brighter value at the center of the image than fragment shading, and estimate the ratio by which it's higher.

Your estimates only need to be accurate to within 10%. To aid you in back-of-the-envelope calculations, the following are accurate within 5%: $(\cos 22.5°)^{30} \approx 1/11$, $(2/\sqrt{5})^{30} \approx 1/28$, and $(\cos 30°)^{30} \approx 1/75$. Also, $(\cos 45°)^{30} = 2^{-15}$*Hint:* Computing the exact ratio for part 2 would require more work than necessary.

Problem 5: Curves (15 points)

I want to approximate a unit circle using four Bézier spline segments. Suppose I decide to do this by making the spline tangent to the circle at the points $(\cos \frac{n\pi}{4}, \sin \frac{n\pi}{4})$ for $n = 0, \ldots, 7$.

1. Give a set of control point positions that will achieve this. You can give the points for just one of the four segments; presumably the others will be obvious by symmetry.
2. Is your spline always outside the circle, always inside the circle, or does it cross the circle?

The matrix for the Bézier spline is:

$$M_{\text{bez}} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Problem 6: Meshes (15 points)

The two triangle strips $[1, 3, 0, 2, 4, 6, 5, 7]$ and $[4, 0, 5, 1, 7, 3, 6, 2]$ are supposed to represent a cube, but there is a problem.

1. What is the problem?
2. Fix the problem by replacing the second triangle strip with a different triangle strip.
3. Give an indexed mesh representation for the resulting surface, including vertex positions with $x, y, z \in \{0, 1\}$ that make the surface correctly oriented.