6.837 Introduction to Computer Graphics
Final exam
Monday, December 13, 2010 1:35-4:30pm
Two sheets of notes (4 pages) allowed

Name:

1	/ 29
2	/ 25
3	/ 13
4	/ 13
Total	/ 80

1   2D Ray Tracing [/29]

Ray tracing is usually used to compute 2D images of a 3D scene. We seek to adapt ray tracing to two-dimensional flatland, and compute 1D images of a 2D scene. Points now only have two coordinates x and y.

1.1   Ray equation

Given an origin (x_O, y_O) and a direction (x_D, y_D), what is the parametric equation for the x and y coordinates of a ray? [**/2**]

1.2   Circle equation

The equivalent of a sphere in 2D is a circle. What is the implicit equation for a 2D circle of radius r centered at the origin? [**/2**]

1.3 Ray-circle intersection

Write the equation for the intersection of the above ray and circle, and provide the solution(s). Comment on the number of intersections and the test to distinguish the various cases. [/9]

1.4 Normal

What is the 2D unit normal to the circle at a point (x, y)? [/2]

1.5 Triangles becomes segments

The 2D equivalent of a 3D triangle is a segment defined by two vertices. We seek to use barycentric coordinates α and β to perform ray-segment intersection.

Write a point P on the line (ab) as a barycentric combination of the two vertices a and b. [/**2**]

What is the conditiont on α and β so that they represent barycentric coordinates?[/**2**]

What extra condition(s) is satisfied when P is inside the segment? [/**2**]

Reduce the above equation to write P as a function of a single barycentric coordinate. [/**2**]

Given 1D texture coordinates u_a and u_b at the two vertices and the barycentric coordinates α and β for P, what is the texture coordinate at P? [/**2**]

1.6 Ray-segment intersection

Write the system of two equations that define the intersection between the segment $[ab]$ and a ray defined by an origin (x_O, y_O) and a direction (x_D, y_D). You do not need to solve the system, but write it in a matrix form. [/**4**]

2 Kd-tree acceleration [/25]

We are now back to 3D ray casting and seek to accelerate it. We consider a generalization of kd-trees where the splitting plane is arbitrary and given by a general plane equation $P.N = d$

```
KdTreeNode:
  KdTreeNode* backNode, * frontNode //children
  Vector3D N
  float d //the splitting plane is defined by N.P=d
  boolean isLeaf
  ListOfTriangles listTri //only for leaves
```

We seek to write the traversal routine `float KdTreeNode::intersect(orig, dir, tstart, tend)` for a ray `(orig, dir)` where `tstart` and `tend` have already been computed and represent the first and last intersection of the ray with the node. It returns the distance to the closest hit. Assume you are given a routine
`float ListOfTriangles::intersect(orig, dir, tstart, tend)`
that returns the distance to the closest hit for a list of triangles.

We assume that the back and front nodes are ordered correctly with respect to the ray direction and we ignore the case where the ray direction is parallel to the splitting plane.

Write the base case (both the test and the action). [/3]

What is t at the intersection between the ray and the splitting plane? You can use mathematical notations. [/3]

Make a drawing of the three possible cases for the configuration of the ray, the node, and the splitting plane. [/4]

Write the full `intersect` method. [/**15**]

```
float KdTreeNode::intersect(Vector 3D orig, Vector3D dir, float tstart, float tend)
```

3   Rasterization [/13]

3.1   Difference with Ray Casting

If the pseudocode for ray casting is

```
for every pixel (ray)
   for every triangle
      does ray hit triangle?
      keep closest hit?
```

What is the pseudocode for rasterization?[/3]

3.2   Visibility

Ray casting maintains the t value for the closest intersection. What is the corresponding mechanism in rasterization? [/2]

3.3   Alias-free shadow maps

How do alias-free shadow maps differ from regular shadow maps? [/4]

3.4 Interpolation

Can we interpolate texture coordinates linearly in screen space? Draw a figure to explain.[/4]
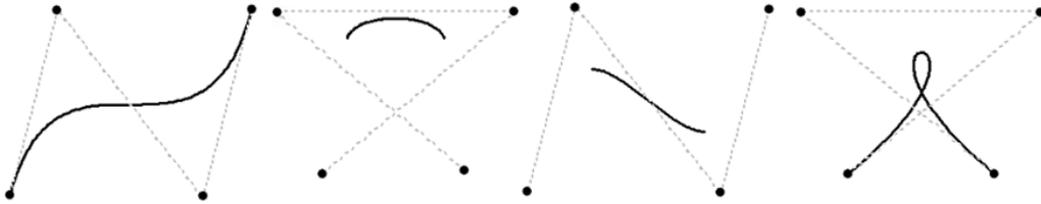
4 Misc [/13]

What does it mean for a function f to be linear? [/2]

Is SSD (skinning) usually implemented in vertex or pixel shaders? [/2]

How many texture pixels must be read in order to perform trilinear lookups? [/2]

What is the difference between lerp and slerp? (no formulae, just behavior). [/2]

Label the curves below as Bézier or B-spline. [/2]

What is the 4x4 matrix for a rotation by angle θ around the y axis?[/3]

6.837 Computer Graphics
Œæ¦Á01G