CPSC 314 **Final Exam** **December 5, 2008**

Name: ____________________ **Student ID:** ____________________

You have 2.5 hours to complete the exam. Write your answers in the spaces provided.

No aids (books, notes, calculators, mobile phones, robots, PDAs, music players, death rays, other electronic devices, etc.) are permitted.

1) Write down a parametric (explicit) description of the line (in 3D) that passes through two distinct points $\vec{x}_0 = (x_0, y_0, z_0)$ and $\vec{x}_1 = (x_1, y_1, z_1)$.

$\vec{x}(s) = \vec{x}_0 + s(\vec{x}_1 - \vec{x}_0)$

2) Why did we use ≤ 0 and ≥ 0 instead of < 0 and > 0 in the tests for triangle rasterization?

This biases the test to include points exactly on the edges of triangles, avoiding **cracks**.

3) Demonstrate that the sum of the edge functions we used for rasterization is constant (independent of the point being tested).

The brute-force way is to write out the edge functions in terms of coordinates, add them up, and show all the terms with the point's coordinates cancel out.

The cute way is to remember the edge function is twice the signed area of a triangle with that edge and the point. Adding them up geometrically gives twice the signed area of the original triangle, independent of the point.

4) Write down the formula for Gouraud interpolation of colour in a triangle, and the formula for Phong normal interpolation, using barycentric coordinates

Gouraud: $\vec{C} = \alpha \vec{C}_0 + \beta \vec{C}_1 + \gamma \vec{C}_2$

Phong normal:

$$\hat{n} = \frac{\alpha \hat{n}_0 + \beta \hat{n}_1 + \gamma \hat{n}_2}{\|\alpha \hat{n}_0 + \beta \hat{n}_1 + \gamma \hat{n}_2\|}$$

5) Explain how matrix multiplication is used to express translation of 3D points.

Add a fourth coordinate of 1, and use a 4×4 matrix as shown for translating by (a, b, c):

$$\begin{pmatrix} x+a \\ y+b \\ z+c \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

6) How is a given 4×4 transformation matrix M used to transform a ray in 3D?

Say the ray has origin (x, y, z) and direction (d, e, f). The transformed ray has origin

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = M \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

and direction

$$\begin{pmatrix} d' \\ e' \\ f' \\ 0 \end{pmatrix} = M \begin{pmatrix} d \\ e \\ f \\ 0 \end{pmatrix}$$

Name: ______________________ **Student ID:** ______________________

7) Of the two perspective projection matrices A and B below, why is B more useful?

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \qquad B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

Multiplication by B perserves depth information, whereas in A the third coordinate always ends up as 1 after homogenization.

8) Why is "instancing" a useful feature of hierarchical modeling?

You can store a single pristine read-only copy of geometry, and render multiple copies of it in different locations and orientations.

9) How do you clip a line segment (like the edge of a triangle) against the near clipping plane in camera space?

Say the segment has endpoints $\vec{x}_0$ and $\vec{x}_1$. If $z_0 \geq -n$ and $z_1 \geq -n$ the edge is completely clipped away; if $z_0 < -n$ and $z_1 < -n$, it is not clipped at all. Otherwise, a point at fraction θ along the line has coordinates

$$\vec{x}_0 + \theta(\vec{x}_1 - \vec{x}_0)$$

Solve for θ where z matches the near clipping plane $-n$:

$$\begin{aligned} z_0 + \theta(z_1 - z_0) &= -n \\ \theta &= \frac{-n - z_0}{z_1 - z_0} \end{aligned}$$

This gives a new endpoint $\vec{x_0} + \theta(\vec{x}_1 - \vec{x}_0)$ to go with the existing endpoint with $z < -n$.

10) Describe a scene where ray-casting would be much more efficient than Z-buffer hidden surface removal, and explain why.

If the depth complexity is high, ray-casting with a good acceleration structure can avoid doing many ray-triangle tests, whereas Z-buffer still has to rasterize every triangle to find the frontmost.

11) What is the camera-space ray corresponding to pixel (i, j) in an $M \times N$ image for an **orthographic** projection with near clipping plane at $z = -n$ and the usual l, r, b, t parameters to describe the left, right, bottom and top camera space x and y coordinates on the near clipping plane.

same as midterm 2

The origin is

$$\left(\frac{i + 0.5}{M}(r - l) + l, \quad \frac{j + 0.5}{N}(t - b) + b, \quad -n \right)$$

and the direction is $(0, 0, -1)$.

12) How do you intersect a ray with a sphere?

same as midterm 2

13) Write down pseudo-code for intersecting a ray with a large collection of objects with a BVH; you don't need to write out the details of intersecting with an individual bounding box.

Call this on the root node of the BVH:

Name: ______________________ **Student ID:** ______________________

```
check(node, ray):
if the ray intersects the node's bounding volume:
  if there is geometry stored in this node:
    check ray against every piece of geometry, returning true if intersects.
  if this node has children:
    check ray against every child, returning true if any of them intersect.
  return false
else
  return false
```

14) Upon which of the following does Lambertian shading depend: the direction to the light $\vec{l}$, the reflection direction $\vec{r}$, the normal $\hat{n}$, the viewing direction $\vec{d}$?

Just $\vec{l}, \hat{n}$.

15) What is ambient light supposed to model?

All the indirect illumination: light bouncing off another surface (or volume) before hitting the surface being rendered.

16) How do raytracers handle shadows?

When gathering light, secondary shadow rays are shot towards light sources to determine visibility: if there is something in the way, that light is not included, producing a shadow.

17) Give both an advantage and a problem with Gouraud shading applied to a model with the glossy Phong material model.

Gouraud is fast, since the formula is only evaluated at the vertices.

However, if a bright highlight occurs between mesh vertices, it won't show up in the rendering, which will look bad.

18) Give a formula for the surface normal of the ellipsoid described implicitly by $3x^2+y^2+5z^2-10=0$.

$$\hat{n}(x,y,z) = \frac{(6x, 2y, 10z)}{\|(6x, 2y, 10z)\|}$$

19) Give both an advantage and a disadvantage to using a procedural texture instead of a voxel array for 3D textures.

Advantage: it may need a lot less storage

Disadvantage: it could be a lot slower to evaluate the procedural texture than simply looking up values in a 3D array.

20) How does OpenGL deal with the magnification and minification problems?

Magnification: bilinear interpolation

Minification: mipmaps

21) What's the simplest way to mitigate quantization artifacts?

Add a small random noise to the value before quantization (i.e. dithering).

22) Write down how to test if a sphere in 3D overlaps an axis-aligned bounding box.

not covered this year

Name: ______________________ **Student ID:** ______________________

23) How might you implement displacement mapping in a ray-tracer?

not covered this year

24) List three different colour spaces.

not covered this year

25) What are metamers, why do they exist, and why are they crucial for computer displays?

Metamers are light signals with different spectra which appear the same to the human eye. The human eye only had three types of cone cells to detect colour, so many different spectra elicit the same response.

Because we are fooled by metamers, a computer display can just shine red, green and blue lights to give the appearance of almost any colour we can see (even if it produces a very different spectrum).

26) Why can't you mix red, green and blue light to produce every colour possible?

Because negative light doesn't exist: in some cases, the RGB values necessary to produce a particular light sensation will drop below zero and thus cannot be physically realized.