CSCI 480: Computer Graphics

Spring 2009 Midterm Exam

3/9/09, 6:30-8:00 PM

Question	Your score	Max score
1		6
2		8
3		5
4		6
5		4
6		6
7		4
8		8
9		8
10		8
11		10
12		8
13		10
14		5
15		4
Bonus		5
Total		105

HAVE FUN!!

Question 1: 6 (2+2+1+1) points

What contributions to CG are you aware of, made by Phong, Sutherland, Jim Blinn and Ed Catmull?

Question 2: 8 points

How does the depth-map-based soft-shadows technique work? In what sense is it "a cheat on top of a cheat"?

Question 3: 5 points

In classic raytracing, rays are sent out from each rendered pixel into the scene, which is backwards compared to real-life where rays emanate from light sources and reach the eye/imaging element. Why does the raytracing algorithm do it backwards?

Question 4: 6 points

In HW2 you are asked to draw the wireframe rendering of a simple scene specification (with multiple meshes). In essence, what are the three nested loops that are required to do this?

Question 5: 4 points

What is the single biggest reason for using matrices in CG (to do rotation, scaling, translation

and other similar transformations)?

Question 6: 6 points

Omitting all the detail, what is the idea behind the Liang-Barsky line clipping algorithm? Explain using an illustration.

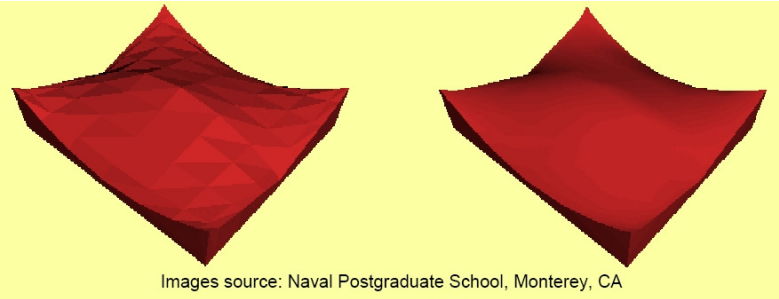
Question 7: 4 points

The following is the classic "lighting equation", showing the ambient, diffuse and specular terms. Why is the summation (across all lights) limited just to the last two terms?

$$I = k_a\ I_a + \sum_{l=lights} \left(k_d\,(N \cdot L) + k_s\,(N \cdot H)^n \right) I_l$$

Question 8: 8 points

The following figure shows two renders of the same polymesh. What are two ways to obtain the smoothness shown on the right? Explain.



Images source: Naval Postgraduate School, Monterey, CA

Question 9: 8 points

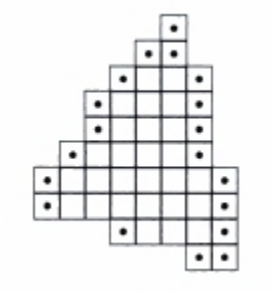
As you (should) know, the following matrix is used for perspective projection. It is used to transform a point [x y z w] in camera space, into NDC (2D) space. What would be the result of replacing 'z' by 'log(z)' before projecting? You can assume that z>1.

$$P = \begin{bmatrix} \frac{2near}{right-left} & 0 & \frac{right+left}{right-left} & 0 \\ 0 & \frac{2near}{top-bottom} & \frac{top+bottom}{top-bottom} & 0 \\ 0 & 0 & -\frac{far+near}{far-near} & -\frac{2far \cdot near}{far-near} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

Question 10: 8 points

This question has to do with the Manhattan Metric, aka "taxicab geometry". Given integer coordinates (x_1, y_1) and (x_2, y_2) in a standard Cartesian grid, what is the distance between the two points? We're not talking about the Euclidian ("as the crow flies") distance, which would be $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. Rather, one point can be reached from the other only by traveling along X or Y (as in city blocks).

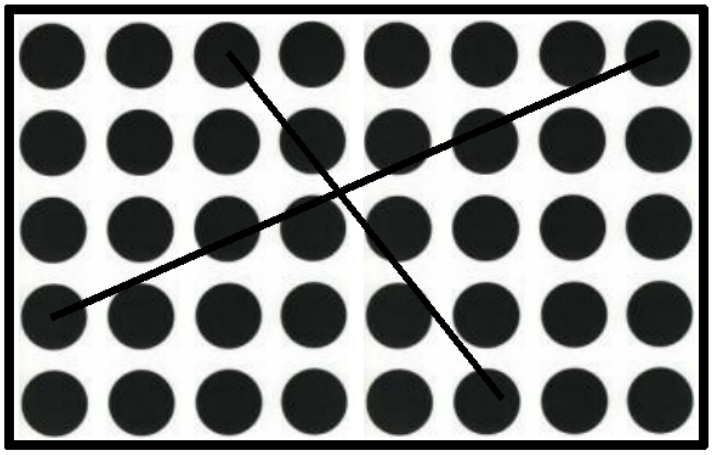
Question 11: 10 (5+5) points



The figure shown above illustrates "fence shading" of a polygon (where normals are interpolated along edge pixels and are used to calculate illumination - resulting illumination is interpolated at the interior). How would you compare this against two better-known shading alternatives, when it comes to running time and image quality?

Question 12: 8 (4+4) points

Below are shown two continuous (non-digital) lines drawn over a grid of pixels (black circles). Place small check marks next to the pixels that should make up each line's digital version. Also indicate appropriate pixel midpoints (which you'd be using to choose pixels for the digital lines, if you were to use the Midpoint Algorithm) for both lines.



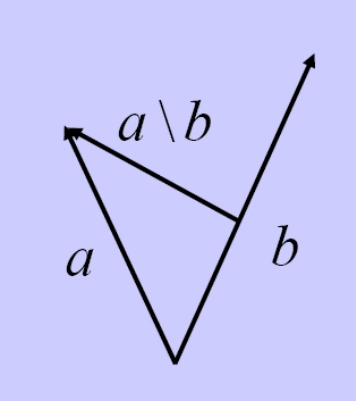
Question 13: 10 points

Below is shown the top part of a .sff mesh description (for a mesh in object space), and a fully rendered character on the right. What are the steps in between? In other words, how does the mesh description turn into recognizable form? You need to be specific but don't need to list equations/algorithms.

```
3263
 4
   2.229345 -0.992723 -0.862826
   2.292449 -0.871852 -0.882400
   2.410367 -0.777999 -0.841105
   2.407309 -0.974980 -0.805091
 4
   2.407309 -0.974980 -0.805091
   2.410367 -0.777999 -0.841105
   2.539200 -0.727778 -0.750475
   2.520417 -0.954785 -0.739445
 3
   2.520417 -0.954785 -0.739445
   2.539200 -0.727778 -0.750475
   2.637655 -0.768176 -0.637039
 4
   2.637655 -0.768176 -0.637039
   2.669281 -0.930664 -0.557166
   2.514167 -1.077721 -0.706614
   2.520417 -0.954785 -0.739445
 4
   2.520417 -0.954785 -0.739445
   2.514167 -1.077721 -0.706614
   2.386465 -1.116066 -0.761367
   2.407309 -0.974980 -0.805091
 4
   2.407309 -0.974980 -0.805091
   2.386465 -1.116066 -0.761367
   2.180012 -1.130557 -0.821812
   2.229345 -0.992723 -0.862826
```

Question 14: 5 points

What is a\b (the 'projection operator') in the figure below?



A. This is in the notes..

Question 15: 4 points

Why do we use 4x4 matrices for rotation, scale and translation in 3D CG (and not 3x3)?

A. Because of translation - it requires a 4x4, 'forcing' rotation and scale to be 4x4 as well (so that they can be inter-multiplied).

Bonus question: 5 points

What makes the following numbers unique?
123, 210, 456, 234, 543, 654, 321, 345, 432

A. These are the only 3-digit numbers that you'd see on a clock :)