

Test 2
CS 5610/6610
Advanced Computer Graphics
Fall 2010

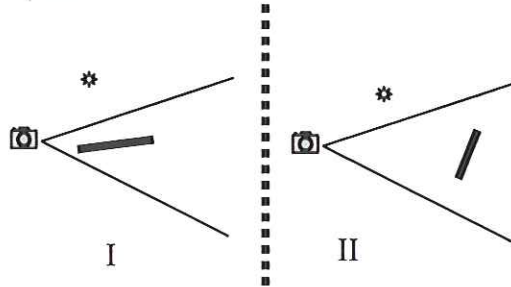
Name: Key

Student ID: _____

Rules:

1. Open book and no notes
2. No calculators, computers, or phones
3. CS 5610 students, answer any 4 questions (no extra credit)
4. CS 6610 students, answer all 6 questions

1. [20 pts] Which leads to more aliasing errors with shadow maps and why?
 (I) a large polygon fully in the field of view from the camera and the light whose normal is nearly orthogonal to the camera or (II) a large polygon fully in the field of view from the camera and light whose normal forms a similar angle to both the light and the camera.

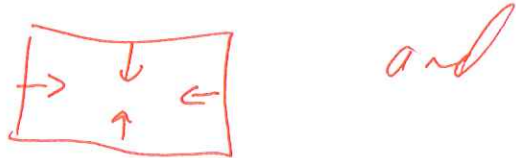


largest aliasing
 Polygon will have z-quantization effects which will lead to errors when fragment is transformed back to the light

- 5 not correct reason
- 10 correct reason but wrong case
- 15 wrong answer

2. [20 pts] You are given a room with 4 walls which are to bump-mapped with a brick normal-map. Describe the shader code needed (Pseudo code with comments is fine) to bump map each of the walls.

Assume walls all have normals pointing in



are aligned with the $+x$ $-x$ $+y$ $-y$ axes

vertex shader

$o_{position} = \text{mul}(\text{modelViewMatrix}, \text{position})$
 $o_{texture\ coord} = \text{texture\ coord}$

~~normal = normal~~

if $\text{normal}.x > 0$ // point in $+z$
 $\text{normal} = \text{normal}.zyx$

else if $\text{normal}.x < 0$ // ~~point~~ point in $+z$
 $\text{normal} = -(\text{normal}.zyx)$

else if $\text{normal}.y > 0$ // point in $+z$
 $\text{normal} = \text{normal}.xzy$

else if $\text{normal}.y < 0$ // point in $+z$
 $\text{normal} = -(\text{normal}.xzy)$

$\text{light\ direction} = \text{lightPos} - \text{position}$

frag Shader

$\text{light} = \text{normalize}(\text{lightDir})$

// texture lookup for normalmap

$\text{normalTex} = \text{tex2D}(\text{NormalMap}, \text{textureCoords}.xyz)$

$\text{normal} = \text{ScaleBias}(\text{normalTex})$

~~$\text{color} = \text{dot}(\text{normal}, \text{light})$~~

-5 no texture lookup

-2 everything right but no mention of swizzle

-10 didn't explain the tangent space or provide pseudocode

-5 good GLSL code but not explanation of tangent space

3. [20 pts] You are given a room with 4 walls, a ceiling and a floor. There is a rectangular area light source on one of the walls.

(5 pts) a. What is the minimum number of polygons needed to form the enclosure for a radiosity solution? You can assume polygons with holes are valid.

4 walls
 1 ceiling
 1 floor
 1 light

7

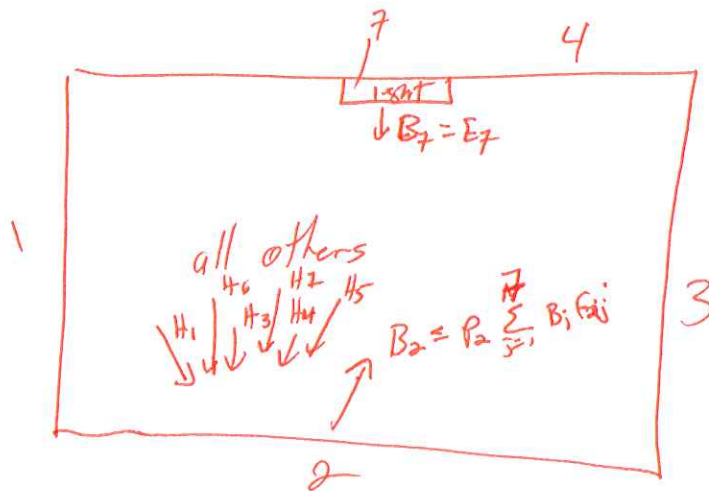
(5 pts) b. Would this produce a decent picture? Why or why not, and if not, how would you fix it?

no, need to sub-divide (refine) the mesh

(10 pts) c. Draw the scene showing the walls (labeled) in the enclosure (looking down from the ceiling; you do not need to draw the ceiling and floor, 2D is OK). Use labels/arrows to describe the radiosity equation for the light and the wall with the window, include any information you can about what would be the values of the terms in the diagram. State any assumptions you are making.

The radiosity equations:

$$B_i = E_i + \rho_i H_i, \quad H_i = \sum_{j=1}^N B_j \frac{A_j F_{ji}}{A_i}, \quad B_i = E_i + \rho_i \sum_{j=1}^N B_j F_{ij} \quad 1 \leq i \leq N$$



where each H_i is the B_i for that wall

4. [20 pts] Explosion maps:

(5 pts) a. What do the C_{ijk} regions represent in the Explosion Map?

They represent distinct reflected regions relative to the tessellation of the reflector

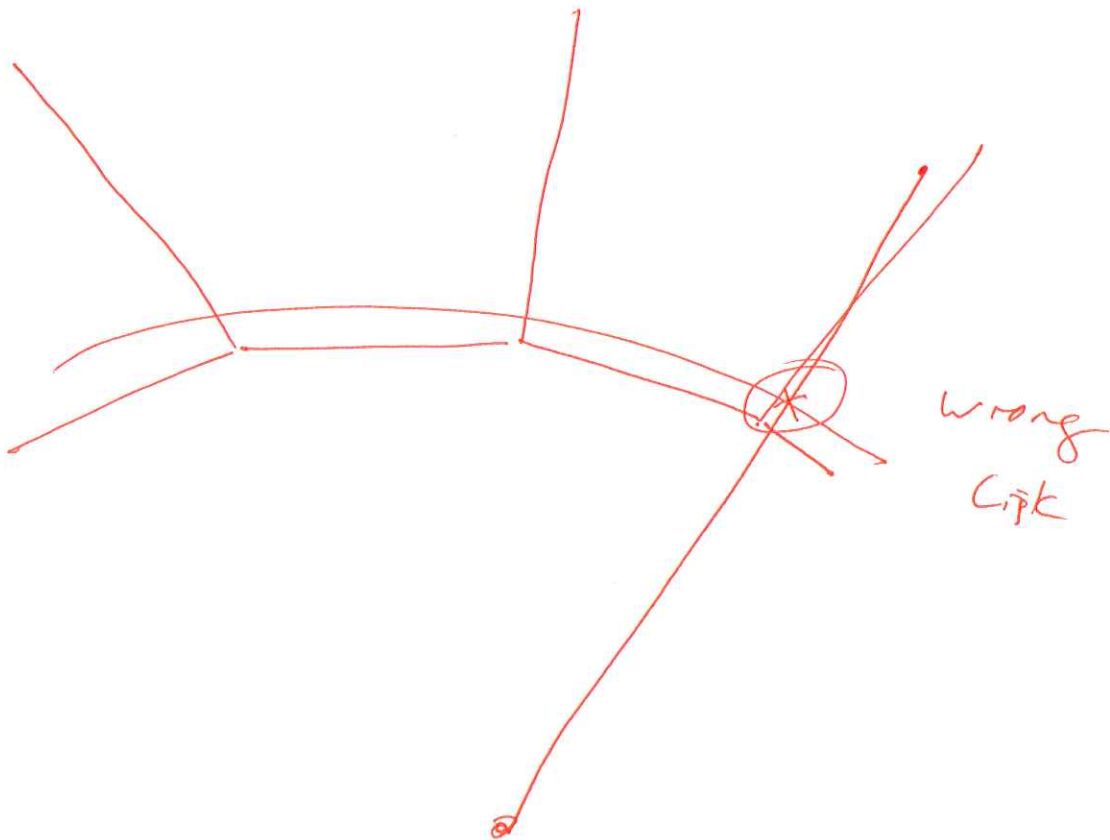
(5 pts) b. How are the C_{ijk} regions represented in the Explosion Map (how are they formed and stored)?

no
no
formed
formed: each reflection Vector is formed for each vertex in the reflector
stored: These are intersected with a sphere and the image (sphere-map) of these is stored in a texture

(5 pts) c. Are these guaranteed to be correct when indexing a reflected geometry vertex?

no

(10 pts) d. Draw a diagram to support your answer (with a short description).



5. [20 pts] How is Toon Shading computed in each of the shaders: vertex and fragment?
Provide pseudo code (with comments) for doing toon shading.

Fragment shader does a
table lookup based on N.L
alternatively, if then else can be used

vertex: pass N ; color

fragment: index = N.L
color = IDtex(index)

6. [20pts] What is the shader code difference between Gouraud shading and Phong shading (think vertex/fragment)? You do not need to provide an equation, just explain the shader differences.

Gouraud: per vertex shading
shaded color is computed per vertex
and interpolated to fragment

Phong: per fragment shading
Normal is passed per vertex
per color is computed in frag prog
using same equation as Gouraud
difference is in specular highlights,
diffuse is the same.