Examen Final de Curs 2016-17 Q1

Gràfics Nom i Cognoms:

Tots els exercicis tenen el mateix pes.

Exercici 1

Aquí teniu una llista d'etapes/tasques, ordenades per ordre alfabètic. Torna-les a escriure a la dreta, però ordenades segons l'ordre al pipeline gràfic.

- Alpha Blending -

Geometry shader -

Rasterització

- Stencil Test

- Geometry shader
- Rasterització
- Stencil Test
- Alpha Blending

Exercici 2

Aquí teniu una llista d'etapes/tasques, ordenades per ordre alfabètic. Torna-les a escriure a la dreta, però ordenades segons l'ordre habitual al pipeline gràfic.

- Divisió de

perspectiva - Fragment

shader

- Pas a Clip Space

- Rasterització

Pas a clip space

Divisió de

perspectiva

Rasterització

Fragment shader

Exercici 3

El LOD 2 d'una textura té 256 x 128 texels. Quina mida té el LOD 0 d'aquesta mateixa textura?

LOD 0 → 1024 x 512 (LOD 1 → 512 x 256; LOD 2 → 256 x 128)

Exercici 4

En quin espai de coordenades estan x, y en les crides GLSL dFdx, dFdy?

Window space.

Exercicis 5 i 6

Indica quina és la matriu (o **producte de matrius**) que aconsegueix la conversió demanada**, usant la notació següent** (vigileu amb l'ordre en que multipliqueu les matrius):

M = modelMatrix V = viewingMatrix P = projectionMatrix N = normalMatrix	M^{-1} = modelMatrixInverse V^{-1} = viewingMatrixInverse P^{-1} = projectionMatrixInverse I = Identitat

a) Pas d'un vèrtex de object space a model space — I

b) Pas d'un vèrtex de object space a world space — M

c) Pas d'un vèrtex de world space a eye space — V

d) Pas de la normal de object space a eye space — N

e) Pas d'un vèrtex de eye space a clip space — P

f) Pas d'un vèrtex de eye space a world space — V^{-1}

g) Pas d'un vèrtex de clip space a world space — $V^{-1} * P^{-1}$

h) Pas d'un vèrtex de object space a clip space — P * V * M

Exercici 7

Hem produït un fragment amb color RGBA = (1.0, 0.5, 0.0, 0.2) corresponent al pixel (i,j). El color RGBA del pixel (i,j) al buffer de color és (0.3, 0.1, 0.3, 0.7). Indica com hem de configurar la funció de blending si volem que el resultat sigui el color RGB (0.5, 0.2, 0.3). Justifica la resposta.

glBlendFunc(...

(1, 0.5, 0) x sfactor + (0.3, 0.1, 0.3) * dfactor = (0.5, 0.2, 0.3)

- ➔ dfactor = 1 , sfactor = 0.2
- ➔ **glBlendFunc(GL_SRC_ALPHA, GL_ONE)**

Exercici 8

Escriu l'equació general del rendering, amb la formulació vista a classe, indicant clarament el tipus de radiància als diferents termes.

Exercici 9

Indica, en la notació estudiada a classe, L(D|S)*E, quins light paths són suportats per:

(a) Raytracing clàssic

(b) Path tracing

Exercici 10

Volem generar amb RayTracing una imatge 256x256 pixels d'una escena interior tancada. Quants shadow rays caldrà traçar, si tenim dos fonts de llum i els materials són difosos i opacs?

Exercici 11

Completa el següent FS per tal que calculi correctament el terme especular (Phong) de la il·luminació:

```
uniform vec4 matDiffuse, matSpecular, lightDiffuse, lightSpecular;
uniform float matShininess;

vec4 light(vec3 N, vec3 V, vec3 L)
{   vec3 R =
normalize( 2.0*dot(N,L)*N-L );
float NdotL = max( 0.0, dot( N,L ) );
float Idiff = NdotL;
    float Ispec = 0;

        if{(NdotL>0)
              float myDot = .......max( 0.0, dot( R,V ) );


              Ispec = ..............pow( myDot, matShininess );
      }

return
        matDiffuse * lightDiffuse * Idiff
       +  matSpecular * lightSpecular *
}      Ispec;
```

Exercicis 12 i 13

Completa aquest codi, corresponent als dos primers passos de l'algorisme de simulació d'ombres per projecció, amb stencil:

```
// 1. Dibuixa el receptor al color buffer i al stencil buffer
glEnable(GL_STENCIL_TEST);
glStencilFunc(GL_ALWAYS,1,1);
glStencilOp (GL_KEEP,GL_KEEP,GL_REPLACE);
dibuixa(receptor);

// 2.Dibuixa oclusor per netejar l'stencil a les zones a l'ombra
glDisable(GL_DEPTH_TEST);
glColorMask(GL_FALSE,...GL_FALSE);
glStencilFunc(GL_EQUAL, 1, 1);
glStencilOp (GL_KEEP, GL_KEEP, GL_ZERO);
glPushMatrix();
glMultMatrixf(MatriuProjeccio);
dibuixa(oclusor);
glPopMatrix();
```

Exercici 14

Indica com varia l'extensió de la penombra en els següents casos:

(a) La llum es puntual

No hi ha penombra

(b) Apropem oclusor i receptor de l'ombra

Disminueix la penombra.

Exercici 15

glPolygonOffset(factor,units) afegeix un offset a la z en window space d'acord a la següent fórmula:

$$Dz * factor + r * units$$

Com es calcula Dz i què representa?

$Dz = max(\partial z/\partial x, \partial z/\partial y)$

Representa quan tangencial és la primitiva a la direcció de visió.

Exercici 16

Tenim una aplicació que no suporta MipMapping, però volem simular el resultat de GL_LINEAR_MIPMAP_LINEAR en GLSL. Completa aquest codi, on lambda és un float amb el nivell de LOD (no necessàriament enter) més adient pel fragment:

```
vec4 sampleTexture(sampler2D sampler, vec2 texCoord, float lambda)
{
        vec4 color0 = textureLod(sampler, texCoord, floor(lambda));
        vec4 color1 = textureLod(sampler, texCoord, floor(lambda)+1);
        return mix(color0, color1, fract(lambda));
}
```

Podeu assumir que textureLod(P,sampler,lod) fa un accés bilineal a textura al punt P usant el nivell especificat a lod.

Exercici 17

Què fa aquesta matriu?

$$\begin{bmatrix} 1-2a^2 & -2ba & -2ca & -2da \\ -2ba & 1-2b^2 & -2cb & -2db \\ -2ca & -2cb & 1-2c^2 & -2dc \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a) Projecció respecte una font posicional situada al punt (a,b,c)
(b) Projecció respecte una font direccional situada al punt homogeni (a,b,c,d)
(c) Reflexió respecte un pla (a,b,c,d)
(d) Projecció ortogonal sobre el pla (a,b,c,d)

(c); n'hi ha prou amb provar amb el pla (0,1,0,0)

Exercici 18

Indica en quin interval (el més ajustat possible) poden estar les coordenades Z dels punts interiors a la piràmide de visió d'una càmera perspectiva, segons l'espai considerat. Pots fer servir ±∞ si s'escau.

- Object space: (-∞,+∞)
  Eye space: (-∞,0)
  NDC: [-1,1]
  Window space: [0,1]

Per alinear (establir la correspondència espacial) els models de volum obtinguts amb les diferents modalitats de captació d'imatge mèdica (MRI, CT...)