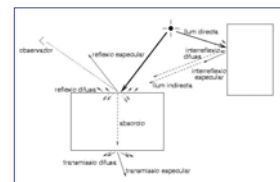


Classe 7: contingut

- Realisme: Il·luminació (2)
 - **Breu recordatori de models empírics**
 - Il·luminació en OpenGL 3.3 (1)
 - Càlcul de color en vèrtexs
 - Shading de polígons
 - Suavitzat d'arestes
 - Il·luminació en OpenGL 3.3 (2)
 - Càlcul de color en fragments

Models d'il·luminació (recordatori)

- Els models d'il·luminació simulen les lleis físiques que determinen el color d'un punt. El càlcul exacte és computacionalment inviable.
- Classificació dels models d'il·luminació:
 - Models Locals o empírics
 - Models Globals: traçat de raig, radiositat



- Focus puntuals
- Objectes opacs
- No ombres
- No interreflexions de llum entre objectes
- No miralls



- Focus puntuals
- Ombres
- Reflexió especular de llum rebuda d'altres objectes
- Miralls



- Ombres i penombres
- Reflexió difusa de llum rebuda d'altres objectes
- No Miralls

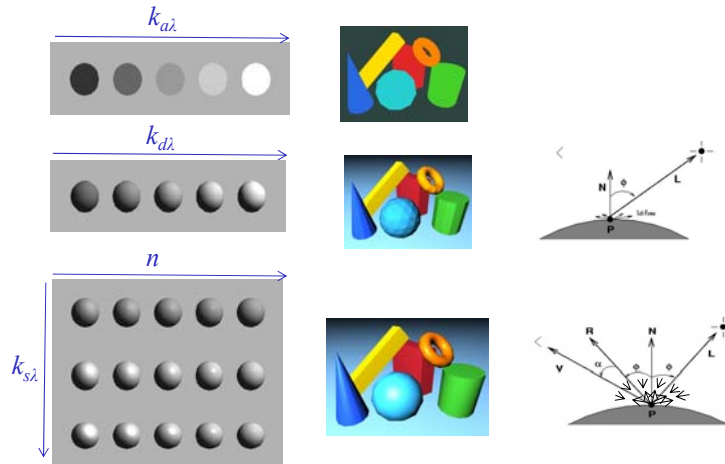
Resum models empírics

$$I_{\lambda}(P) = I_{a\lambda}k_{a\lambda} + \sum_i (I_{f_i\lambda} k_{d\lambda} \cos(\Phi_i)) + \sum_i (I_{f_i\lambda} k_{s\lambda} \cos^n(\alpha_i))$$

Aportació ambient

Aportació difusa

Aportació especular



IDI 2018-2019 2Q

3

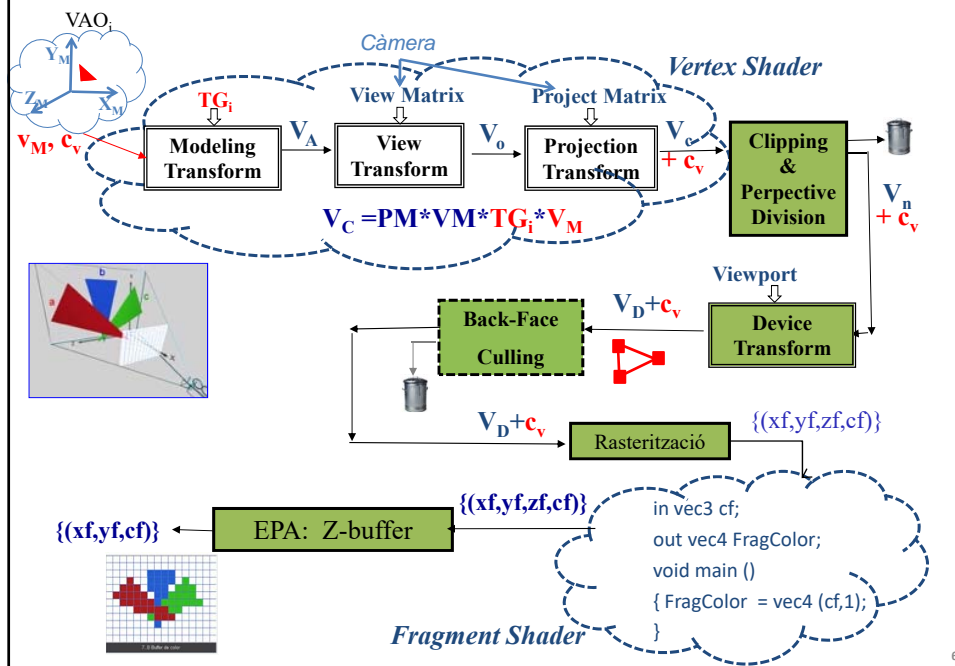
Classe 7: contingut

- Realisme: Il·luminació (2)
 - Breu recordatori de models empírics
 - **Il·luminació en OpenGL 3.3 (1)**
 - Càlcul de color en vèrtexs
 - Shading de polígons
 - Suavitzat d'arestes
 - Il·luminació en OpenGL 3.3 (2)
 - Càlcul de color en fragments

IDI 2018-2019 2Q

5

Procés de visualització: recordatori



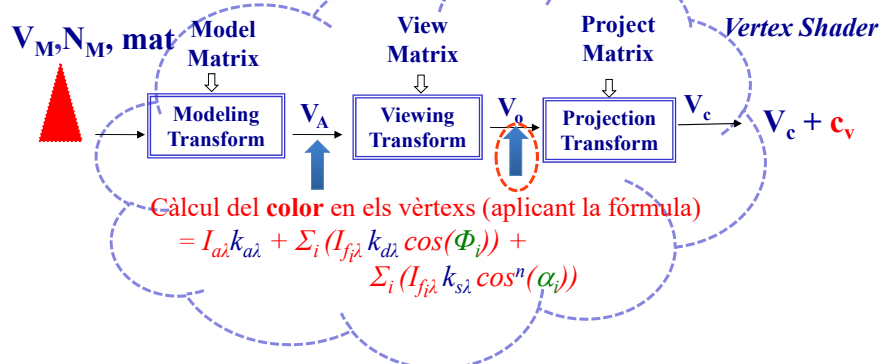
Procés de visualització: Càlcul color en Vèrtex

- Atributs:

Coordenades vèrtex, normal i constants de material per vèrtex en VBOs dels VAO

- Uniforms:

- Fonts de llum actives => color, posició
- Llum ambient



Càlcul color en un punt: models empírics

El càlcul el farem per cada vèrtex (al Vertex Shader)

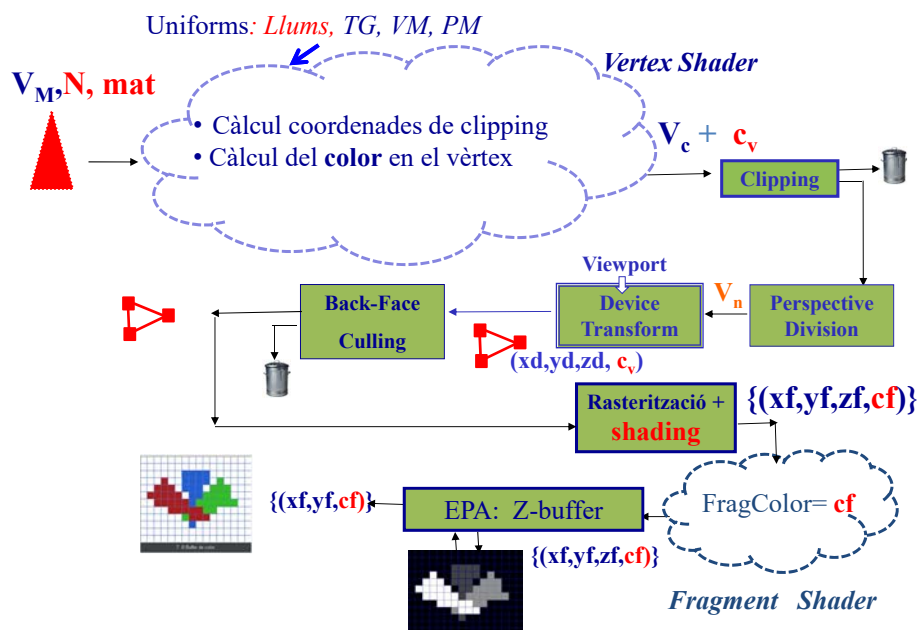
I el farem en SCO, per tant:

- Cal passar la posició del vèrtex a SCO
 - multiplicant per (**view * TG**)
- Cal passar el vector normal a SCO
 - multiplicant per la matriu **inversa** de la **transposada de (view * TG)**, -li direm **NormalMatrix-**
 $\text{mat3 NormalMatrix} = \text{inverse}(\text{transpose}(\text{mat3}(\text{view} * \text{TG})))$
- La posició del focus de llum també ha d'estar en SCO
 - Multiplicat per **view** (si no la tenim directament en SCO)

IDI 2018-2019 2Q

8

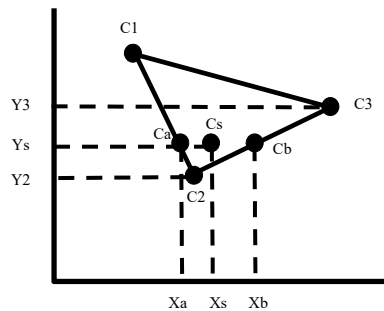
Procés de visualització: Shading (colorat) de polígons



9

Shading (colorat) de polígons

- Colorat Constant \equiv **Flat shading** $\rightarrow C_f = C_l$
color uniforme per tot el polígon (funció del color calculat en un vèrtex); cada cara pot tenir diferent color.
- Colorat de Gouraud \equiv **Gouraud shading** \equiv **Smooth shading**



$$Ca = \frac{1}{Y1 - Y2} (C1(Ys - Y2) + C2(Y1 - Ys))$$

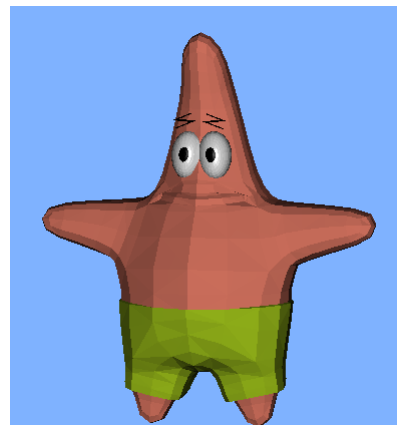
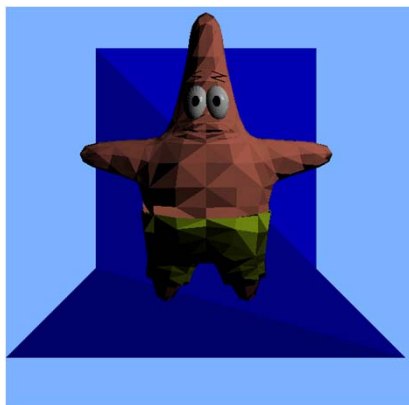
$$Cb = \frac{1}{Y3 - Y2} (C2(Y3 - Ys) + C3(Ys - Y2))$$

$$Cs = \frac{1}{Xb - Xa} (Ca(Xb - Xs) + Cb(Xs - Xa))$$

IDI 2018-2019 2Q

10

Flat versus Gouraud Shading

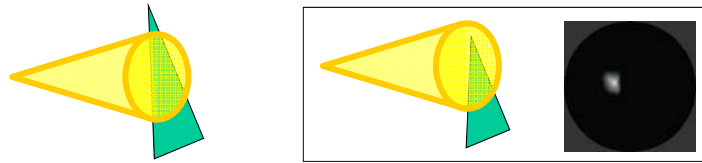


IDI 2018-2019 2Q

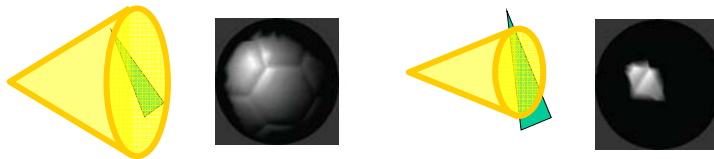
11

Limitacions del colorat de polígons:

- Taca especular en mig d'una cara → desapareix → discretitzant millor
- Taca en un vèrtex



- Il·luminació si ens apropem a un polígon gran → s'enfosqueix
- Efectes en cara d'un cub → discretitzant millor



IDI 2018-2019 2Q

12

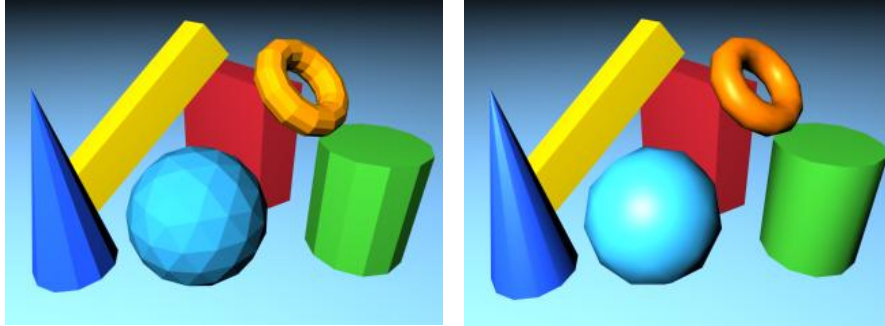
Classe 7: contingut

- Realisme: Il·luminació (2)
 - Breu recordatori de models empírics
 - Il·luminació en OpenGL 3.3 (1)
 - Càlcul de color en vèrtexs
 - Shading de polígons
 - **Suavitat d'arestes**
 - Il·luminació en OpenGL 3.3 (2)
 - Càlcul de color en fragments

IDI 2018-2019 2Q

13

Avantatge del Shading: Suavitzat d'arestes



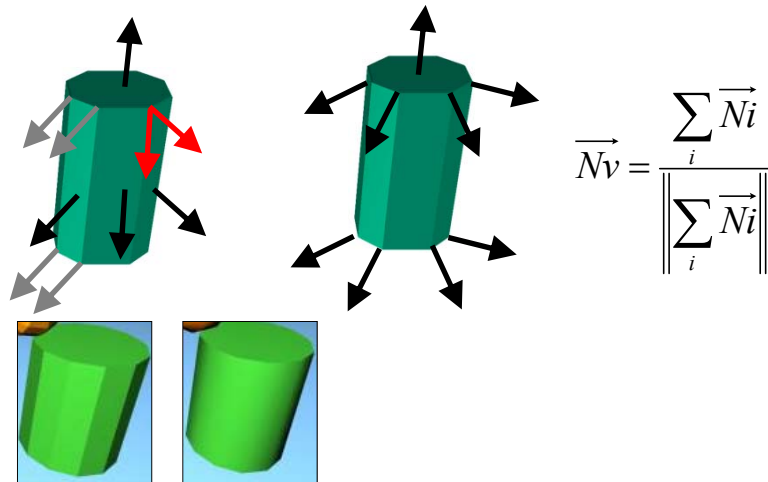
Quin model d'il·luminació i shading s'utilitza?
Per què no es veuen les arestes?
Noteu la forma de les siluetes

IDI 2018-2019 2Q

14

Suavitzat d'arestes

- Normal per cara vs normal per vèrtex

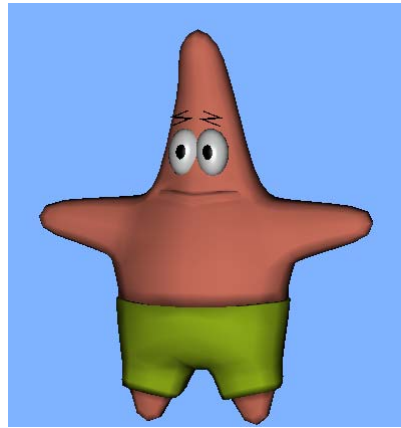
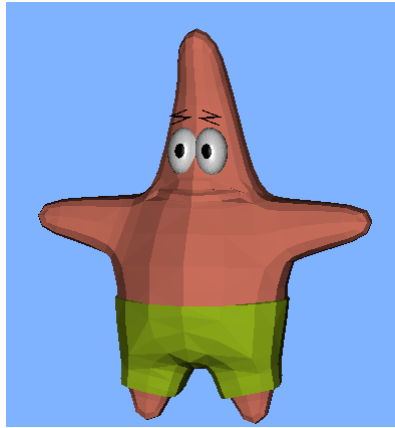


IDI 2018-2019 2Q

15

Suavitzat d'arestes: exemple

- Normal per cara vs normal per vèrtex



IDI 2018-2019 2Q

16

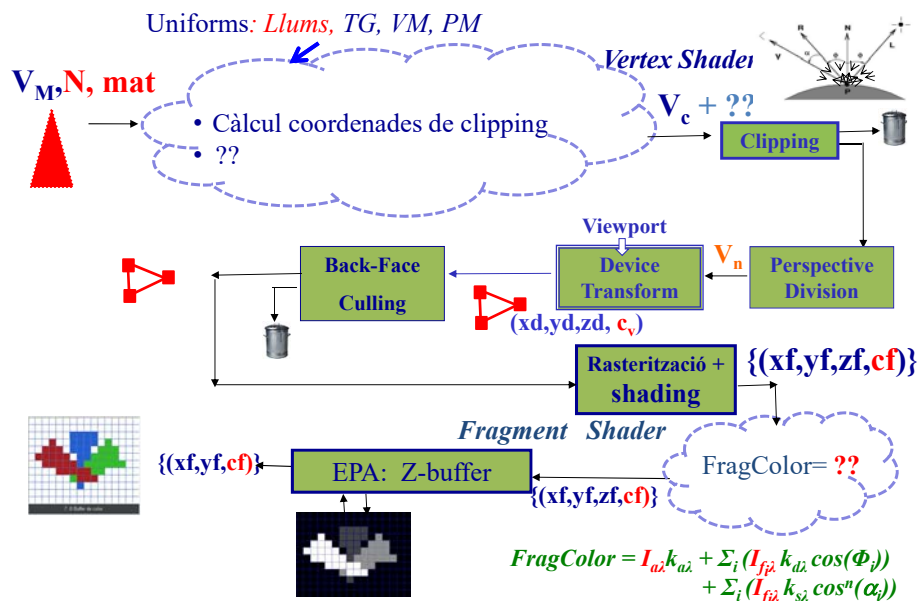
Classe 7: contingut

- Realisme: Il·luminació (2)
 - Breu recordatori de models empírics
 - Il·luminació en OpenGL 3.3 (1)
 - Càlcul de color en vèrtexs
 - Shading de polígons
 - Suavitzat d'arestes
 - **Il·luminació en OpenGL 3.3 (2)**
 - Càlcul de color en fragments

IDI 2018-2019 2Q

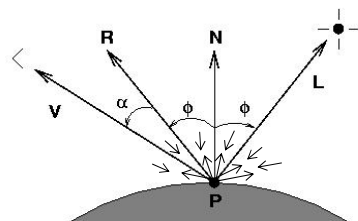
17

Procés de visualització: Shading (colorat) de polígons



18

Millor aproximació al càlcul del color en un punt: “Shading de Phong” en FS



Idea 2:

- Podem fer “out” del VS dels atributs associats a vèrtex com N, V (en SCO) i també de les constants de material.
- La rasterització aproximarà els seus valors pel fragment interpolant la informació dels vèrtexs del triangle ☺

Idea 1: Per cada píxel (fragment) càlcul del color

- Càlcul color per fragment:

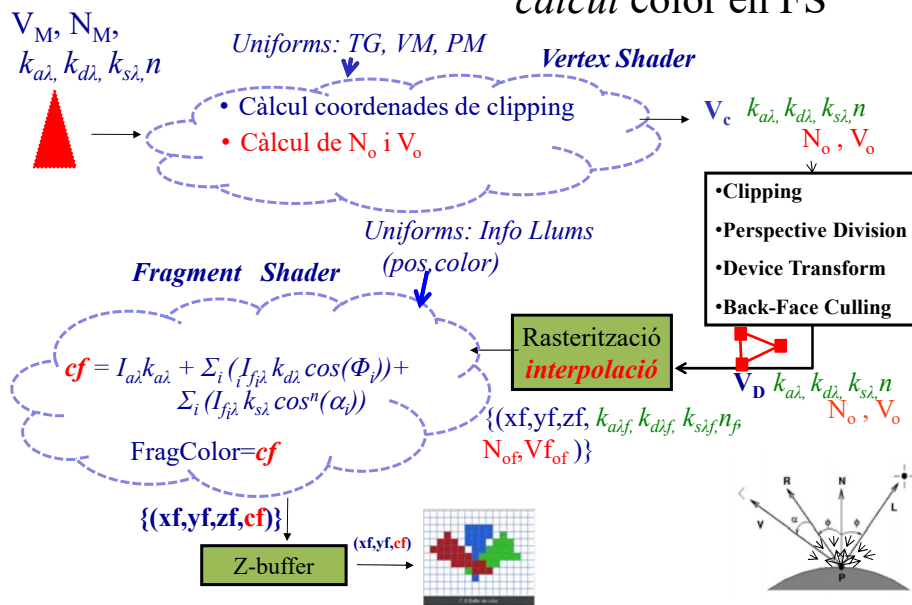
$$FragColor = I_{a\lambda} k_{a\lambda} + \sum_i (I_{f\lambda} k_{d\lambda} \cos(\Phi_i)) + \sum_i (I_{f\lambda} k_{s\lambda} \cos^n(\alpha_i))$$

$$\cos(\Phi) \Rightarrow \text{dot}(L, N) \text{ en SCO}$$

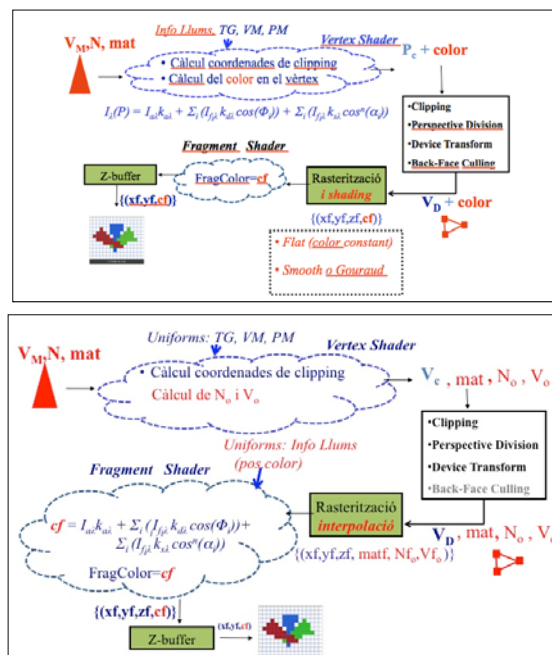
$$\cos(\alpha) \Rightarrow \text{dot}(R, V) \text{ en SCO}$$

- Requereix info de llums \Rightarrow uniforms
- Requereix el punt, altres vectors en SCO o SCA i les constants material
- Tenim el punt en SCD \Rightarrow podríem calcular les seves coordenades en SCO o SCA; però com podem saber N i les constants material?

Procés de visualització: Phong Shading i càlcul color en FS

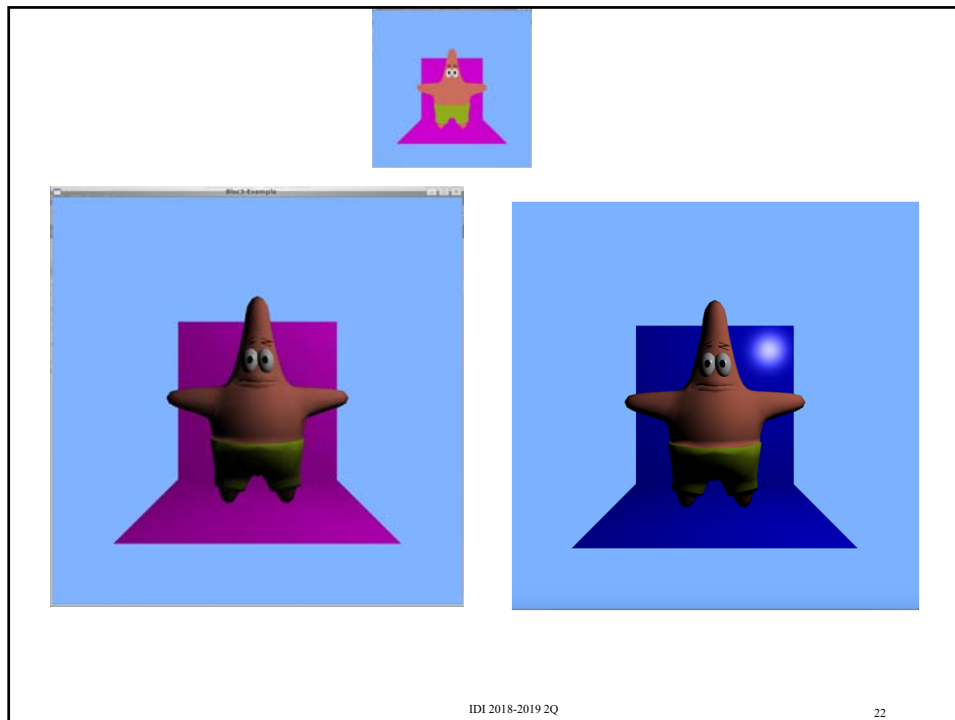


20



IDI 2018-2019 2Q

21



Exercici 6:

Una escena està formada per dos cubs amb les cares paral·leles als plans de coordenades. El CUB1 té aresta 20, el centre de la seva base en $(0,0,0)$ i és de color verd i mate; el CUB2 té aresta 20, centre de la seva base en $(30,0,0)$ i és del mateix color verd però brillant. Il·luminem l'escena amb un focus groc situat en $(50,10,0)$. L'observador es troba en una posició que pot veure les cares dels cubs ubicades en $x=10$ i $x=40$. Si es pinta l'escena amb OpenGL utilitzant model d'il·luminació de Phong en VS i Smooth shading (Gouraud Shading), de quin color es veuran aquestes cares? No hi ha llum ambient.

- La cara en $x=10$ és veurà de color verd constant, la cara en $x=40$ també és veurà de color constant però d'un verd més fosc.
- La cara en $x=10$ és veurà de color verd constant, la cara en $x=40$ també és veurà de color constant però d'un verd més clar.
- La cara en $x=10$ és veurà de color verd constant, la cara en $x=40$ també és veurà de color constant però d'un verd més clar i amb una taca especular groga en mig de la cara.
- La cara en $x=10$ és veurà amb diferents tonalitats de verd, la cara en $x=40$ també és veurà amb diferents tonalitats de verd però més clars i amb una taca especular groga en mig de la cara.

Exercici 7:

Un cub amb constants de material $K_d=(0.8,0,0.8)$ i $K_s=(1,1,1)$ i $N=100$, és il·luminat amb un focus que emet llum de color $(1,1,0)$. No hi ha llum ambient. La càmera (correctament definida) és axonomètrica i l'observador i el focus estan a una distància 10 d'una cara (i mirant cap a ella) sobre una recta que és perpendicular a la cara i que passa pel seu centre. Indica, raonant la resposta:

- a) quins colors observa l'observador en el cub si s'utilitza *FLAT shading* (colorat constant)? Indica els colors dels vèrtexs.
- b) quins colors observa l'observador en el cub si es pinta amb *SMOOTH shading* (colorat de Gouraud)?

Exercici 8:

Volem il·luminar un polígon de 10×10 ubicat sobre el pla XZ i centrat en l'origen, amb un focus de llum blanca ubicat en la posició $(0, 2, 0)$. No hi ha llum ambient. La normal del polígon és $(0, 1, 0)$. Les constants de material del polígon són: $K_d = (0, 0.8, 0)$, $K_s = (1, 1, 1)$ i Shininess = 100. Indica quina de les següents afirmacions és la correcta:

- a) Com la llum ha d'estar fixa en l'escena, el càlcul de la il·luminació s'ha de fer obligatòriament en el vèrtex shader per a cada vèrtex del polígon.
- b) Si el càlcul de la il·luminació es realitza en el fragment shader, cal passar la posició de la llum i la normal a coordenades de dispositiu.
- c) Si el càlcul de la il·luminació es realitza en el vèrtex shader, cal que les posicions del vèrtex, del focus i la normal estiguin referenciades totes respecte al sistema de coordenades de l'aplicació o de l'observador.
- d) La imatge –acoloriment- que s'obtindrà del polígon serà la mateixa tant si els càlculs es realitzen en el vèrtex com en el fragment shader, sempre que es realitzin en el sistema de coordenades adient.

Exercici 8:

Volem il·luminar un polígon de 10×10 ubicat sobre el pla XZ i centrat en l'origen, amb un focus de llum blanca ubicat en la posició $(0, 2, 0)$. No hi ha llum ambient. La normal del polígon és $(0, 1, 0)$. Les constants de material del polígon són: $K_d = (0, 0.8, 0)$, $K_s = (1, 1, 1)$ i $\text{Shininess} = 100$. Indica quina de les següents afirmacions és la correcta:

- a) Com la llum ha d'estar fixa en l'escena, el càlcul de la il·luminació s'ha de fer obligatòriament en el vèrtex shader per a cada vèrtex del polígon.
- b) Si el càlcul de la il·luminació es realitza en el fragment shader, cal passar la posició de la llum i la normal a coordenades de dispositiu.
- c) Si el càlcul de la il·luminació es realitza en el vèrtex shader, cal que les posicions del vèrtex, del focus i la normal estiguin referenciades totes respecte al sistema de coordenades de l'aplicació o de l'observador.
- d) La imatge –acoloriment– que s'obtindrà del polígon serà la mateixa tant si els càlculs es realitzen en el vèrtex com en el fragment shader, sempre que es realitzin en el sistema de coordenades adient.

Exercici 9:

Una escena està formada per dos cubs d'aresta 2 amb cares paral·leles als plans coordenats i centres als punts $(0, 1, 0)$ i $(3, 1, 0)$. El primer és vermell i el segon verd. Ambdós són mats.

Per error s'ubica a l'usuari a la posició $(0, 1, 0)$ amb VRP al $(3, 1, 0)$. L'òptica és axonomètrica amb un $\text{window} = (-4, 4, -4, 4)$, $zN = -1$, $zF = 6$. S'ubica una llum blanca a $(8, 1, 0)$. Si no hi ha llum ambient, i el *background* és blau, indica què es veurà en funció del mètode d'eliminació de parts amagades que s'utilitza:

- a) Si només s'empra *back-face culling*: un quadrat de color negre
- b) Si tenim *zbuffer* i *back-face culling* activats: un quadrat de color verd
- c) Si només tenim el *zbuffer* activat: un quadrat de color vermell
- d) Si només tenim el *back-face culling* activat: un quadrat de color verd

Classe 7: contingut

- Realisme: Il·luminació (2)
 - Breu recordatori de models empírics
 - Il·luminació en OpenGL 3.3 (1)
 - Càlcul de color en vèrtexs
 - Shading de polígons
 - Suavitzat d'arestes
 - Il·luminació en OpenGL 3.3 (2)
 - Càlcul de color en fragments