

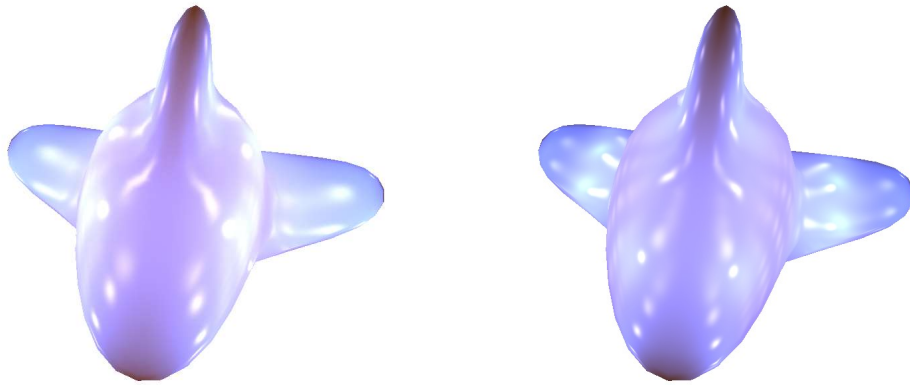
---

## ManyLights (manylights.\*)

useu ~/assig/grau-g/Viewer/GlarenaSL per resoldre aquest exercici.

---

Escriu VS+FS per il·luminar el model amb un nombre arbitrari de fonts de llum distribuïdes de manera uniforme dins la capsa englobant del model:



Concretament, volem aplicar **el model d'il·luminació de Phong per fragment** (com va fer a l'exercici lighting 4):

$$K_a I_a + K_d I_d (N \cdot L) + K_s I_s (R \cdot V)^s$$

on

$K_a$ ,  $K_d$ ,  $K_s$  = reflectivitat ambient, difosa i especular del material

$s$  = shininess del material

$I_a$ ,  $I_d$ ,  $I_s$  = propietats ambient, difosa i especular de la llum

$N$  = vector normal unitari

$L$  = vector unitari cap a la font de llum

$V$  = vector unitari del vèrtex cap a la càmera

$R$  = reflexió del vector  $L$  respecte  $N$ . Es pot calcular com  $R=2(N \cdot L)N-L$

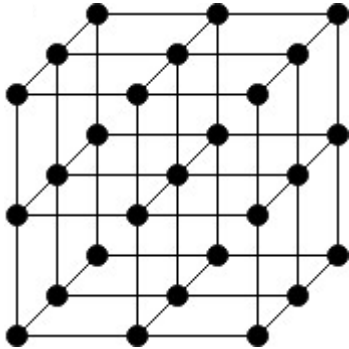
Pel que fa al material, feu servir els uniforms usuals (matAmbient, matDiffuse, matSpecular, matShininess). Pel que fa al color de la llum, useu també els uniforms habituals (lightAmbient, lightDiffuse, lightSpecular).

En comptes d'una única font de llum a lightPosition, n'hi haurà (NUM+1)\* (NUM+1)\* (NUM+1) fonts de llum, on

**uniform int NUM = 5;** // no proveu amb valors més grans que 5

Les fonts de llum (en object space) les heu de situar en una graella regular, cobrint la capsa englobant del model (donada per boundingBoxMin, boundingBoxMax).

Per exemple, si  $NUM = 2$ , les fonts de llum estaran en les posicions de la figura (una d'elles coincidirà amb `boundingBoxMin` i una altra amb `boundingBoxMax`):

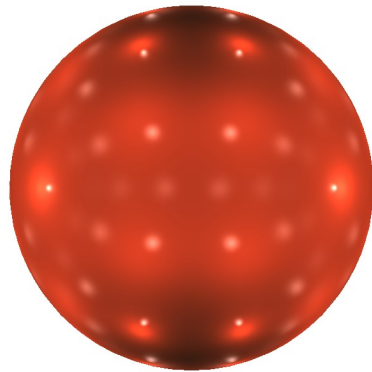


Òbviament, pels càlculs d'il·luminació, les llums hauran d'estar en l'espai adient.

El color final del fragment serà la suma de les contribucions de totes les fonts de llum. Com que n'hi ha moltes, per tal de no cremar el model, a la contribució de cada llum caldrà multiplicar-li un decreixement exponencial en funció de la distància  $d$  (en eye space) entre el punt i la llum,

$$\exp(-\text{decay} * d);$$

amb **uniform float decay = 6.0**, de forma que cada llum només il·luminarà la geometria més propera.



**Identificadors obligatoris:**

`manylights.vert`, `manylights.frag`

Els uniforms de l'enunciat.