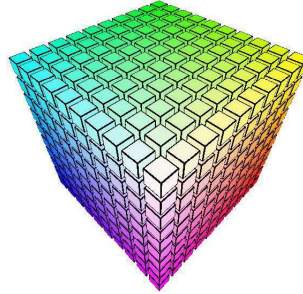


RGB color space (rgb-color-space.*)

Ureu: ~/assig/grau-g/Viewer/GLarenaSL

Escriu VS+GS+FS per representar, amb cubs de colors, l'espai de color RGB:



Considerarem que els valors (r,g,b) d'aquest espai tenen components en $[0,1]$. Treballarem amb l'objecte **filled-cube.obj** el qual té una col·lecció de triangles distribuïts per l'interior d'un cub amb extrems als punts (-1, -1, -1) i (1,1,1).

El **VS** farà les tasques imprescindibles, escrivint `gl_Position` en *object space*.

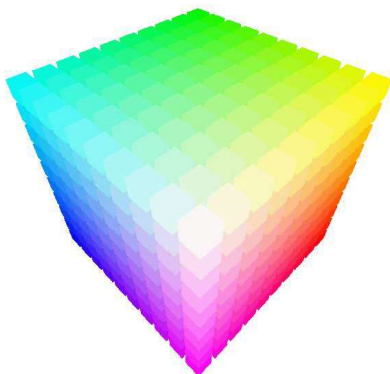
El **GS** crearà un cub per cada triangle, centrat al centre del triangle, i amb una mida de 0.16. El color de les cares del cub dependrà del `uniform int mode = 3`.

Si **mode = 1** [5 punts], el color de les cares del cub l'haureu de calcular a partir de les coordenades del centre C del triangle; aquest centre tindrà coordenades (x,y,z) en $[-1, 1]$, que haureu de transformar a coordenades (r,g,b) en $[0,1]$.

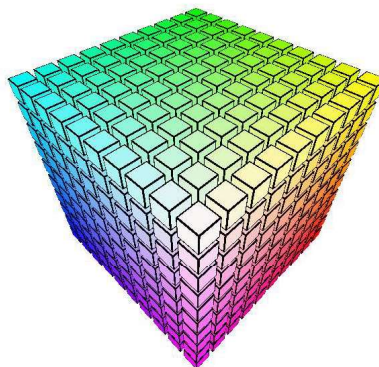
Si **mode = 2** [3 punts], el FS assignarà al fragment el color que li arribi del GS, tret d'un petit marge al voltant de cada cara, que serà de color negre. Per aquest marge, podeu fer que el GS generi coordenades de textura (s,t) per cada vèrtex, entre 0 i 1, i que el FS utilitzi el color de la cara si s, t estan dins $[0.05, 0.95]$; i que altrament assigni color negre.

Si **mode = 3** [2 punts], a diferència del mode 2, el GS només generarà un cub si el centre C té alguna coordenada més petita que `uniform float cut = -0.25`, de forma que seran visibles cubs de l'interior de l'espai RGB.

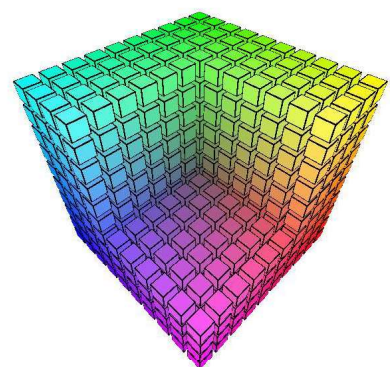
El FS farà les tasques imprescindibles, assignant el color que li arribi del GS, excepte pel petit marge en els modes 2 i 3.



Mode = 1



Mode = 2



Mode = 3

Identificadors obligatoris:

`rgb-color-space.vert`, `rgb-color-space.geom`, `rgb-color-space.frag` (en minúscules!)

La resta d'uniforms estàndard necessaris segons l'enunciat.