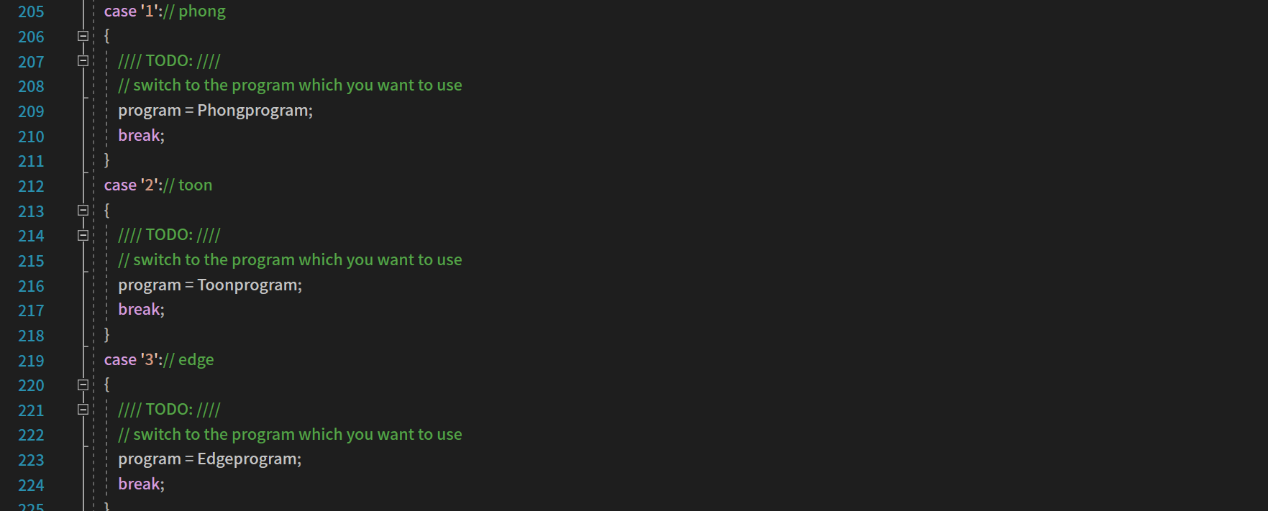
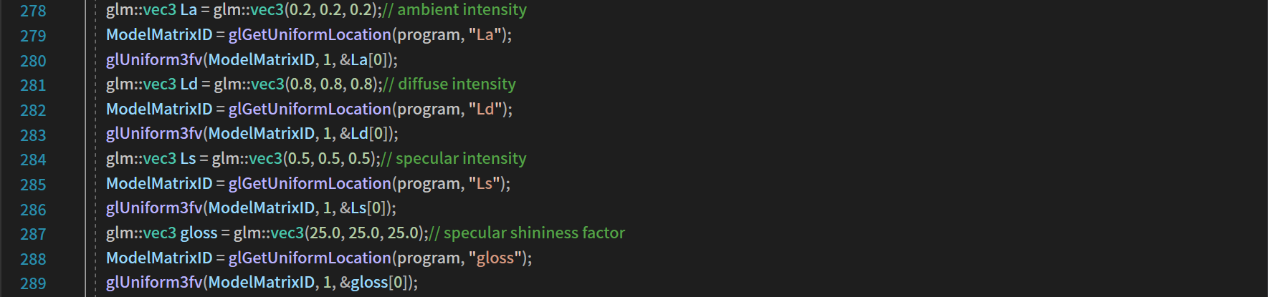
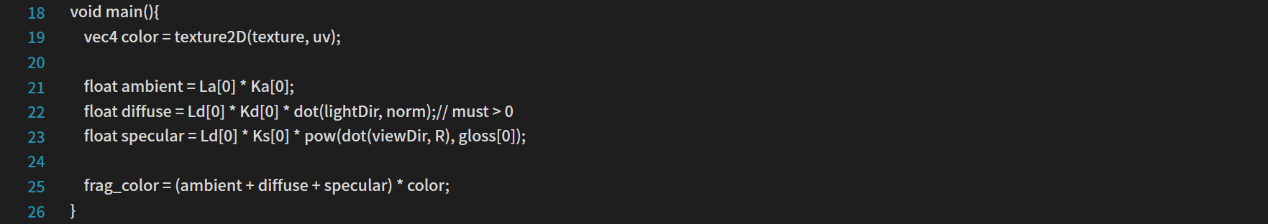
20201222 CG hw3 Report

0716207 呂思函

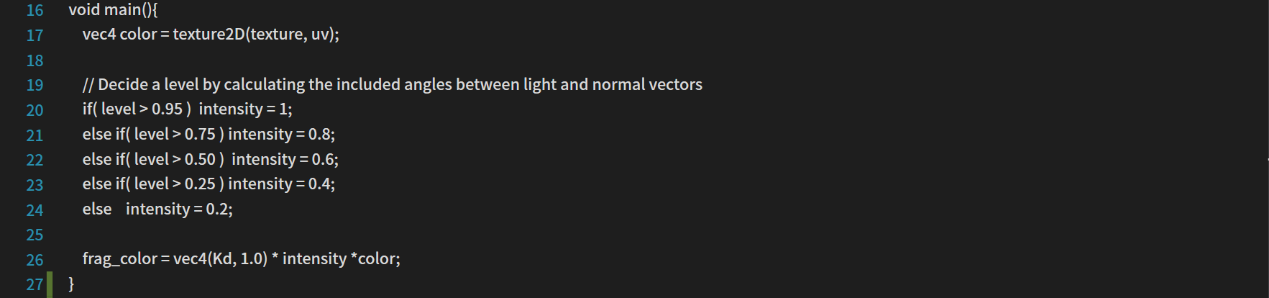
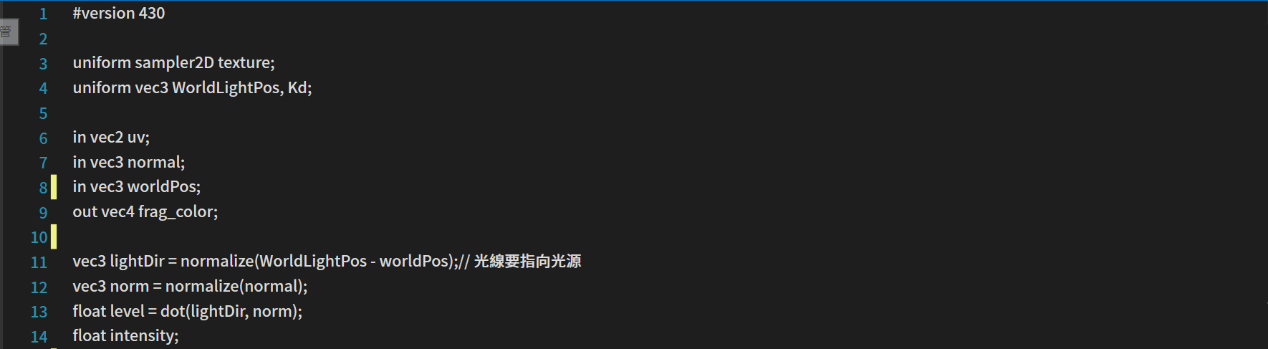
* Shader初始化 : shaderInit()
  + 創建三個shading program : Phong Shading, Toon Shading, Edge Effect
  + 並且初始化的program是Phong Shading，讓程式一執行的時候就有模型
* Keyboard control
  + 利用鍵盤切換不同的program
* 畫伊布 : DrawUmbreon()，初始化各個參數
  + 需要注意的點是WorldLightPos和WorldCamPos是3d vector
  + 所以要用glUniform3fv傳參數到shader裡



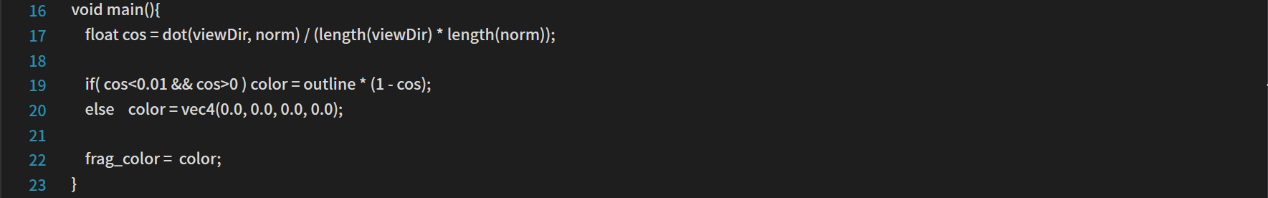
* 建立各種3d vector參數
* 因為找不到float要怎傳，所以gloss的部分還是用3d vector去傳
* Vertex shader
  + 三個shading的vertex shader都相同
  + 用location拿main的位置、法向量、材質的參數
  + 利用uniform傳入model(M)、model view(V)、projection(P)
  + gl\_Position用來設定model的位置
  + 要傳到fragment shader的normal必須先經過模型變換(先inverse再transpose)，因為要讓light和model同在world coordinate或model coordinate才不會有衝突
  + worldPos這個變數用來儲存模型的世界座標，之後傳給fragment shader
* PhongShading(fragment shader)



* + 用uniform去接要使用的各種參數
  + 發現的問題 : fragment shader的out只能有一個，不然會有衝突
  + main裡的是計算Phong Shading各種因子的公式 : ambient、diffuse、specular(因為數值預設的關係所以dot不會小於0，否則應該會有更進一步的判斷=>要取>0的結果，且gloss必須>0)
  + 要把各種參數normalize的原因是 : 這些要計算的向量必須換成單位向量，否則計算結果可能會>0(燈光各種參數加起來應該會是1，否則會爆亮過曝)
* ToonShading(fragment shader)



* + 上半部傳參數和變數宣告與Phong Shading類似
  + 底下main的地方是將顏色分成五層，利用公式將單位向量的light vector和normal vector內積
  + 得到對應的光強度之後，再與diffuse係數和texture color相乘
* EdgeEffect(fragment shader)



* + 上半部傳參數和變數宣告與Phong Shading及Toon Shading類似
  + 設定一個邊框顏色，在這邊我使用青色讓邊框清楚一點
  + 要取得圖案邊框，要從三角函數去取，這邊使用單位向量的光線和法向量的內積來得到cos
  + 將cos值接近0，盡量逼到圖案邊緣、以取得外框
  + 其他區塊就設成黑色