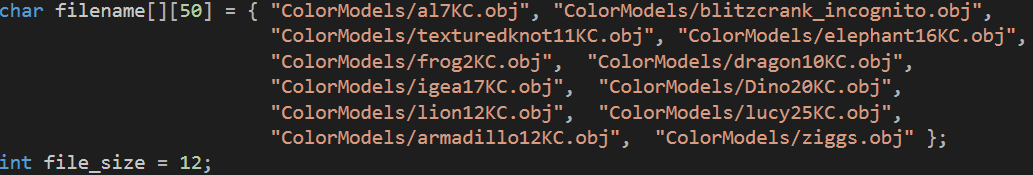
計算機圖學 作業一報告

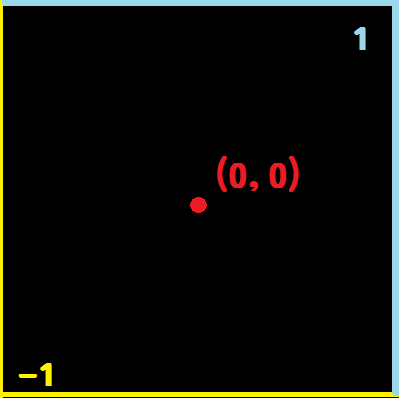
103062234 張克齊

一、 The method of operating program:

執行程式前，要先確定在CG\_HW1資料夾底下有ColorModels這個資料夾，裡面用來存放要讀取的.obj檔；接著，要修改程式中所開的filename陣列和file\_size，若filename中寫了12個路徑的名稱，則要將file\_size改成12，如下圖所示，這樣就能順利執行程式。



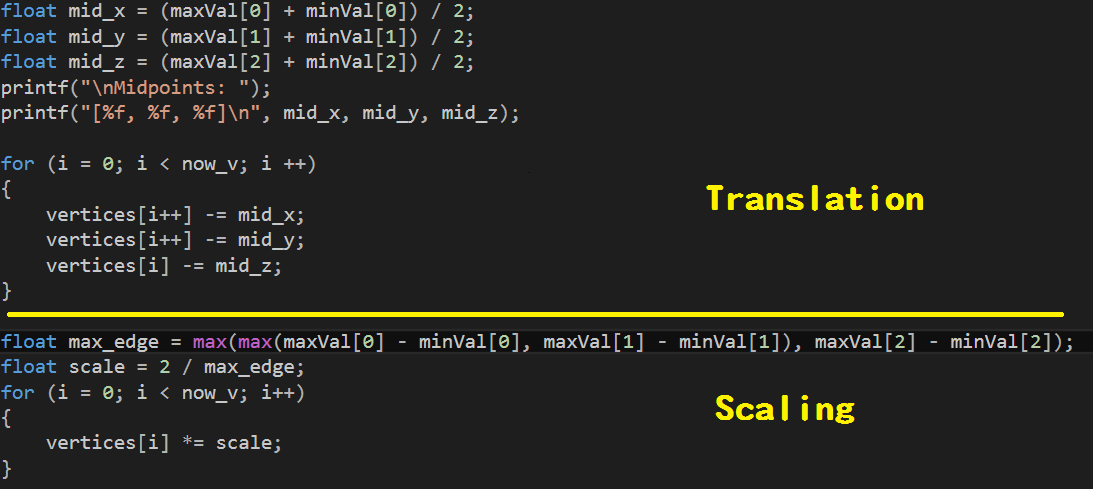
二、 The method of normalization:

**<- normalize to [-1, 1] 示意圖**

首先，我們要從traverseColorModel()這個function中進行修改，必須如同範例的三角形一樣，創建兩個GLfloat的array分別儲存vertices座標和colors座標(x, y, z)，而因為一張圖中是由很多三角形所構成，可以利用OBJ->numtriangles取得這張圖由多少個三角形組成，又因為每個三角形有三個點和三個座標，代表存一個三角形需要9個數值，因此至少要(OBJ->numtriangles)\*9大小的array才能存完所有的點，如下圖。



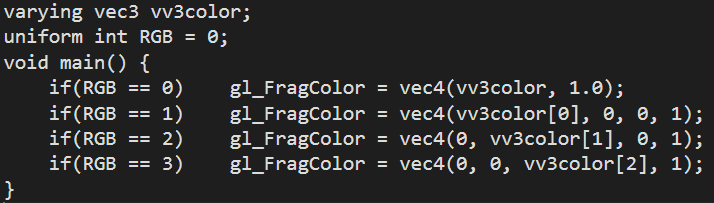
接著是normalization的部分，要分成translation和scaling兩個步驟來操作。第一個是translation的部分，我們要先找出讀進來OBJ的中心，方法是找出x, y及z軸上對於這個OBJ最大和最小的vertex，也就是這個OBJ的邊界，並利用code中所給的maxVal和minVal來儲存，而在讀每一個新的vertex時要進行max function做比較更新值；當我們取到三個軸的邊界後，就能利用(maxVal+minVal)/2找出OBJ的中心座標(Midpoint)，在利用一個新的for迴圈對所有的點(vertices array)進行平移，將OBJ移到視窗中心(0, 0)的位置，完成translation的動作。再來我們要處理scaling，要注意的是不能直接將整張圖縮放至[-1, 1]，因為會改變OBJ的比例；因此，我們也要利用剛剛得到的邊界maxVal, minVal來實現；利用max function找出x, y, z軸的最長邊界，並將這個邊界控制在[-1, 1]間，其他軸則是造著這個比例(下圖的變數scale)來進行scaling，這樣就能完成normalization的動作。下圖為找到x, y, z軸maxVal和minVal後的操作。



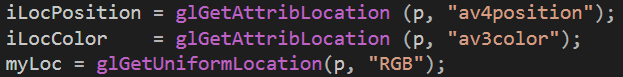
三、 Other implementation:

1. Shader:

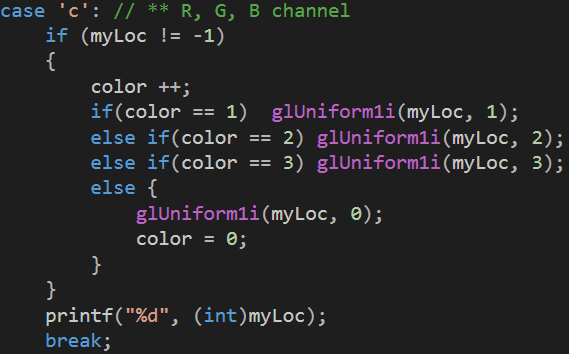
在處理只顯示R、G、B channel時，可以利用一個uniform variable連結main與shader，達到切換channel的效果。首先，我們可以先對shader.frag進行修改，創建一個uniform variable來控制我們想要的模式，並透過取出vv3color(OBJ的color)的[0](R channel)、[1](G channel)、[2](B channel)彆設其他為0來達到單色的效果，如下圖。



接著在main的部分，因為在setShaders()中已經將兩個shader彼此bind，所以我們可以直接建一個GLint來取出我們的uniform variable，所用的函式為glGetUniformLocation()，如下圖所示，第二個參數雙引號內的RGB就是我們在shader中創建的變數名稱，這樣就能取出該變數的位置。

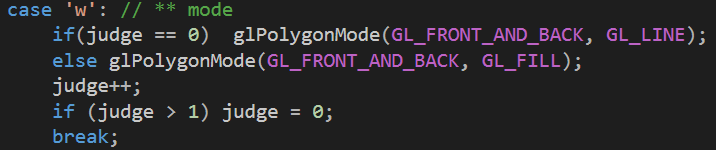


最後，在進行操作的部分，我們必須先確定myLoc不等於-1，才能確定有連接到shader的變數；確認後我們就可以利用glUniform1i來設定shader中uniform variable的值，1i的意思為一維的int，若想創建不同類型會有不同對應的glUniform函式。下圖為按鍵c的操作，順序為normal, R, G, B channel。



2. Wireframe/Solid mode:

要切換填滿和三角形模式，可以利用PDF中提示的glPolygonMode()來實現；只要設定其中的第二個參數，GL\_FILL代表會著色填滿，GL\_LINE代表指連點但不填滿顏色，就能達到三角形的效果，下圖為按鍵w的操作。



四、 Problem and efforts:

因為助教在一開始code架構和PDF中都給了很多的提示，所以在normalization的部分是很順利進行的，只有因為忘記要new float array而稍微卡住。而在改shader的部分就比較多挑戰，雖然一開始就有著要從main傳一個變數進shader，但試了一些方法都沒有成功；直到研究OpenGL官方tutorial中有關uniform variable的部分，才成功做出題目的要求。

在這次的作業中，前面的部分讓我瞭解了整份code的架構，後面則是對shader有非常深刻的認識，是一個很棒的學習！

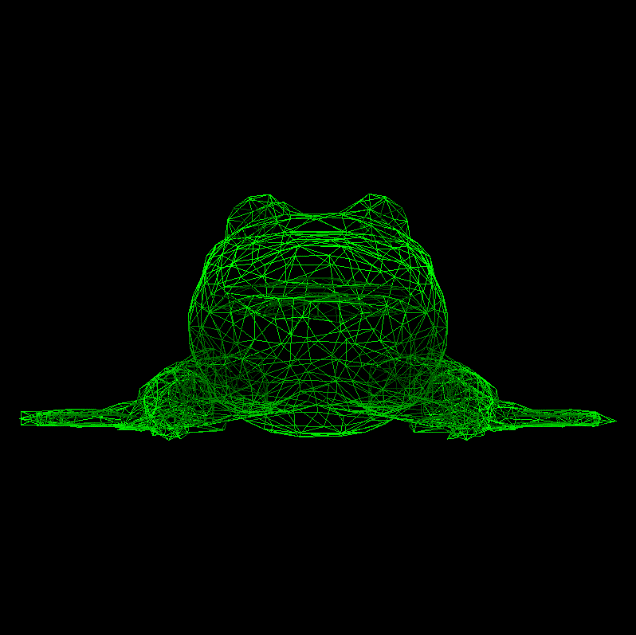
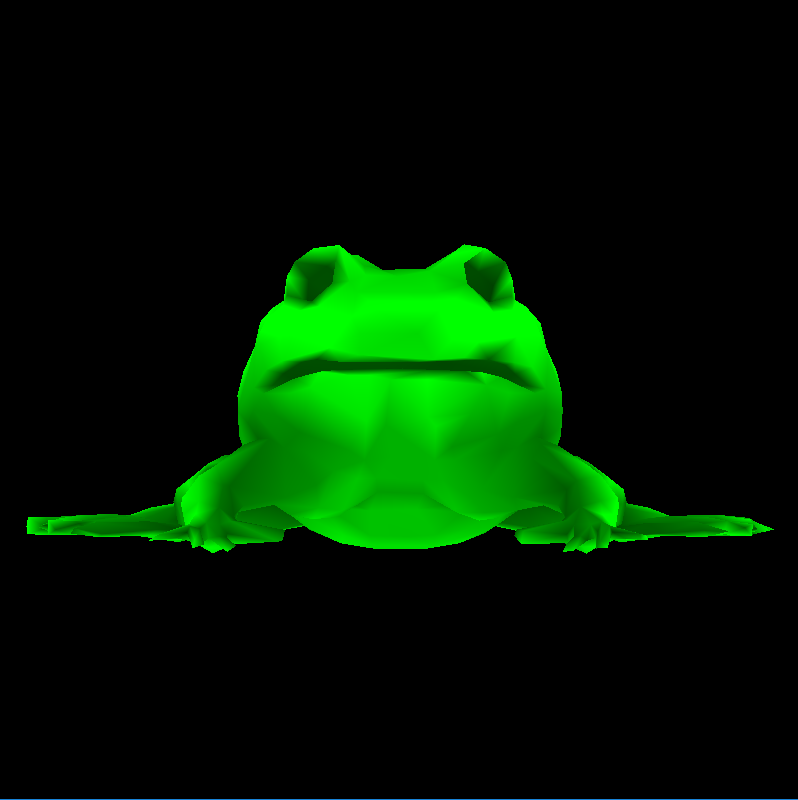
五、 Results:



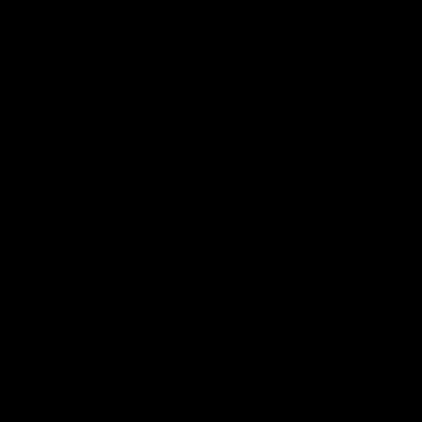
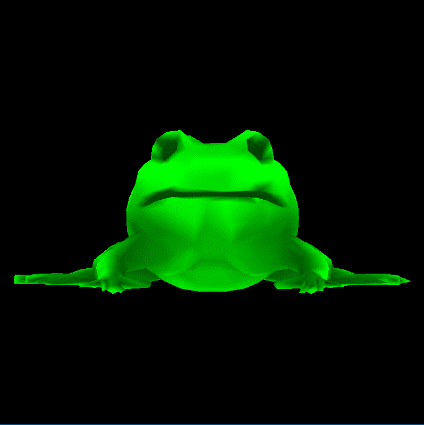
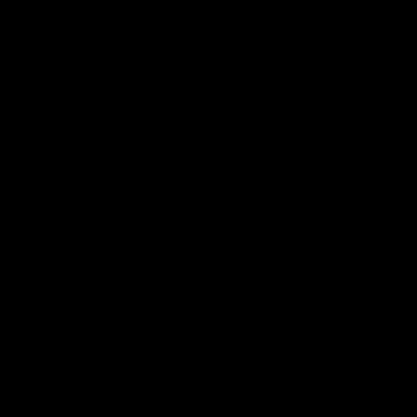
**Solid mode Wireframe mode**



**R channel G channel B channel**



**Solid mode Wireframe mode**



**R channel G channel B channel**