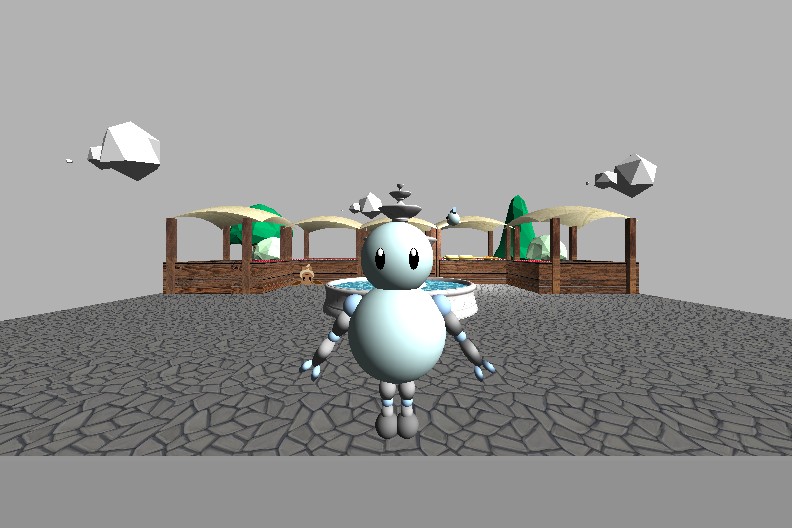
2023Computer Game Engine, Project #1, the shaders

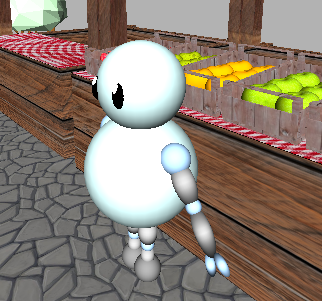
1. **介紹:**

這次作業的內容是要自己寫shader並且創作場景，我選擇小攤販市場作為我的主題，其實我原本的想法是中世紀貿易市場(就是異世界動漫常常出現的那種市場)，但是還不太擅長建模，所以先使用簡單的小攤販當作這學期的主題。

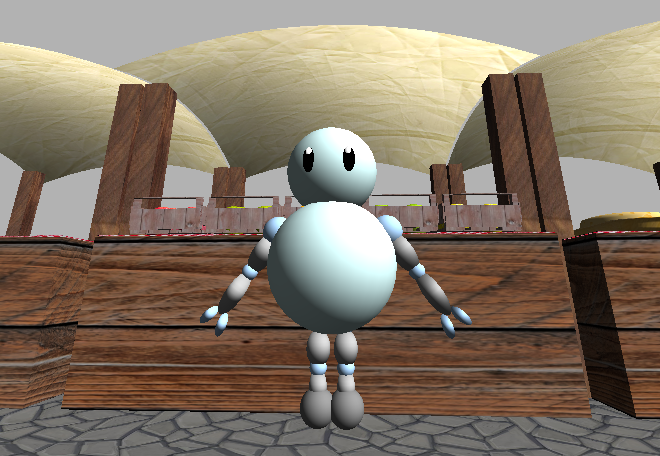


1. **使用方法:**

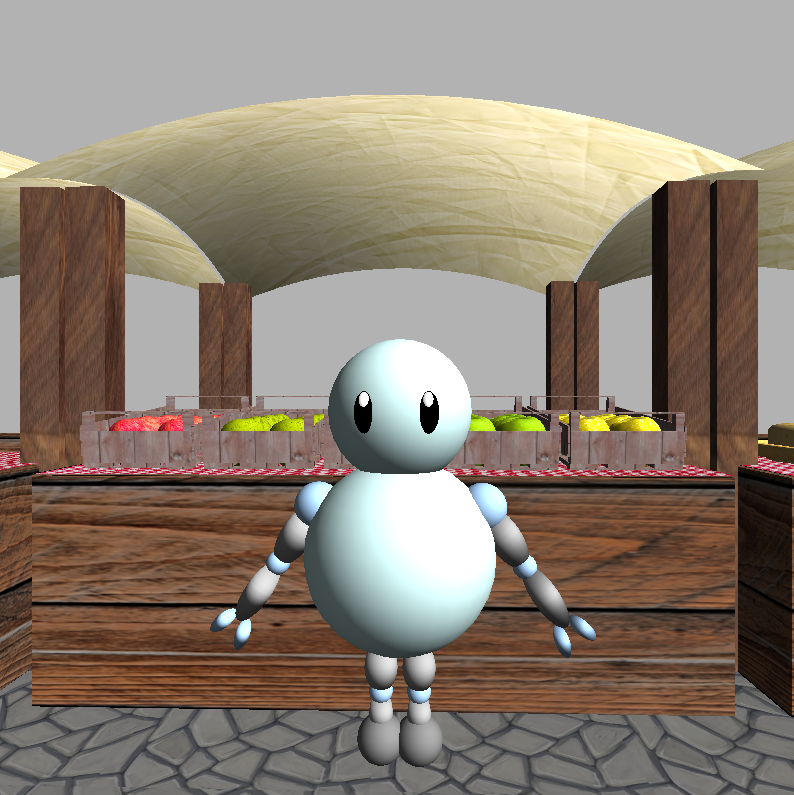
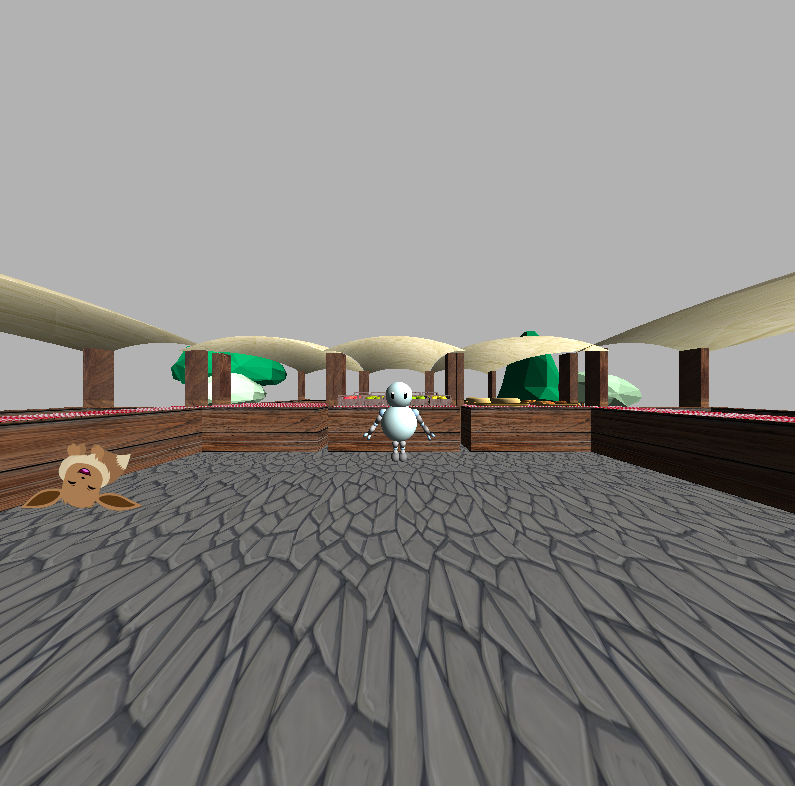
* 機器人走路: 機器人的前進方向比較特別，我是根據目前的視野去做方向的判斷(類似rpg遊戲)，連按2下會切換成跑步。
  + - W/w -> 遠離使用者
    - A/a -> 往左方
    - S/s -> 靠近使用者
    - D/d -> 往右
    - R/r -> 原地轉圈圈
    - 空白鍵 -> 跳躍
  + 視野:
    - 按著右鍵且往右滑 -> 視野向右
    - 按著右鍵且往左滑 -> 視野向左
    - 按著右鍵且往上滑 -> 轉為由上往下看
    - 按著右鍵且往下滑 -> 轉為由下往上看
    - 滾輪往上 -> zoom in
    - 滾輪往下 -> zoom out
    - Ctrl + back -> 初始化為預設視野

▲ 視野向右/視野向左

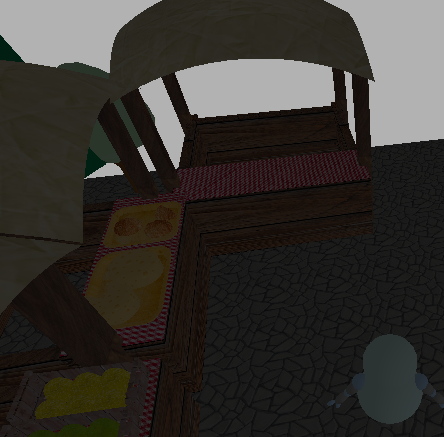
 

▲ 由下往上看/由上往下看

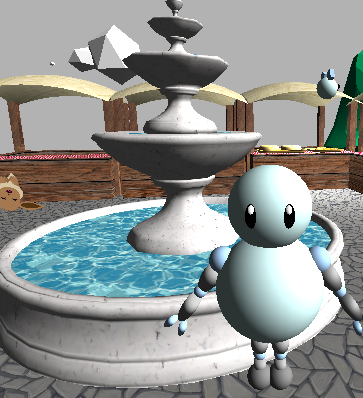
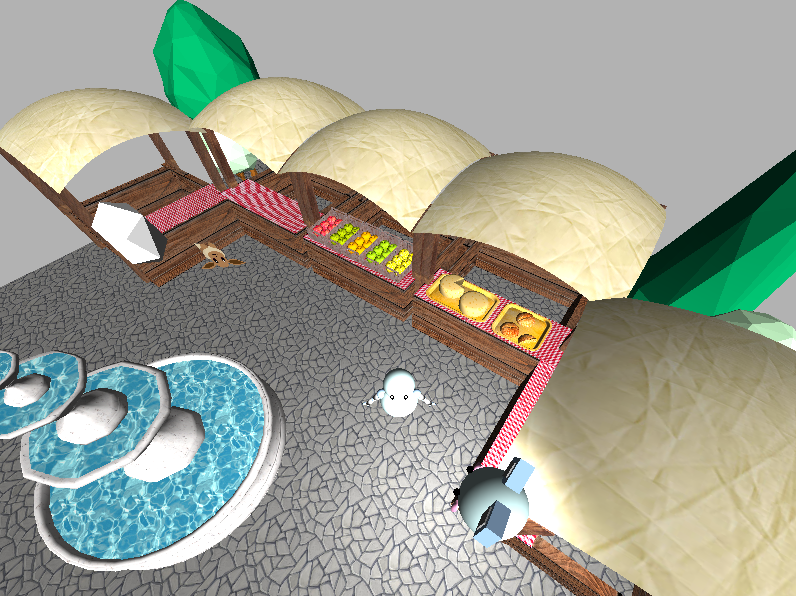
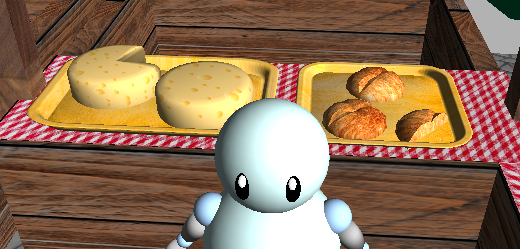
▲ (皆為整個view port截圖)zoom in/zoom out

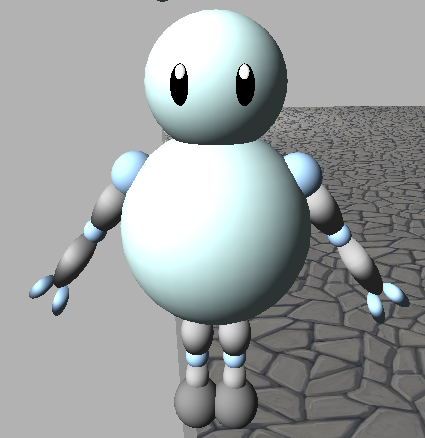
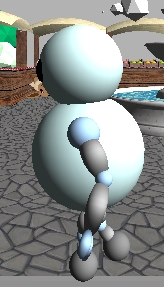
* + 燈光開關: 按下0可以對平行光做開關

▲關平行光/在關平行光下被spot light照到

1. **實作要點 & 程式內容:**
   * 場景:
     + 場景內有五個攤販(水果 & 麵包)、四棵樹、噴水池、三朵雲、spot light小精靈還有一隻伊布(他是動態billboard，由55張圖片組成)。

* + - 小精靈spot light，會在天上移動，碰到障礙物會隨機朝某個方向轉彎。
    - 平行光是沿著(1,1,1)的方向射來。
    - 機器人不得超過邊界，也不得穿越物體(有做障礙物偵測效果)
    - 幾乎都有做texture mapping(攤販、水果、麵包和噴水池)
  + 機器人:
    - 機器人是之前圖學的作業，他在做運動時(跑走跳)，關節都會一起動。 
    - 比較不一樣的是，我之前圖學都是把東西全部放到main.cpp裡，但這次我嘗試拆成很多檔案，有物件導向的感覺(?不然之前寫到最後好痛苦，一個main有4000多行，變數名稱到最後都改不定了。

 ◂ 寫成class

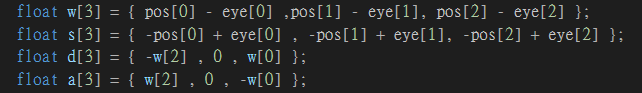
* + View & projection:
    - 視野的部分我是參考一款叫托蘭異世錄的視野移動方式。他的往前是往場景的前方，然後可以用滑鼠配合W/A/S/D去做同步的移動，不需要先調到好位置在移動。



* + - 實作的原理是，攝影機固定看著機器人的xz座標，攝影機本身是在機器人**後方30單位**處。因為畫面轉動後，機器人的**前方就會改變**，所以抓攝影機位置時必須\*上旋轉矩陣。底下的eye是攝影機位置，pos是機器人位置，eyeAngY是畫面旋轉的角度。



* + - 有了這些資訊，就可以透過**機器人位置 - 眼睛位置**去算往前方的向量，再透過往前方的向量外積(0,1,0)得到左方和右方的向量。
    - 機器人移動時，就根據這4個向量的單位向量\*移動篇幅就可以了!

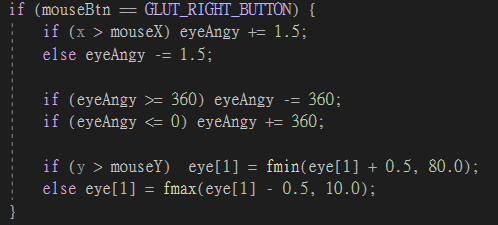


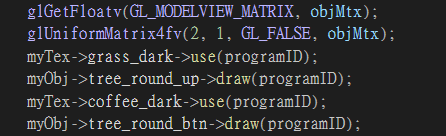
▲w/a/s/d方向向量的算法(d/a是w和(0,1,0)做外積)

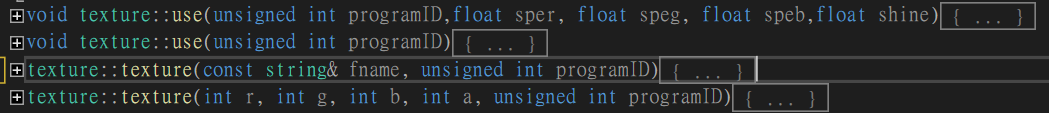


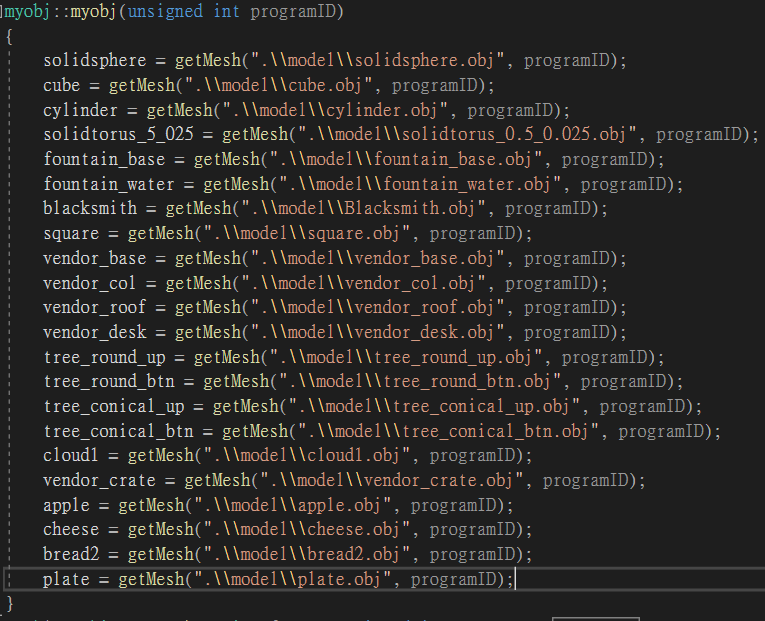
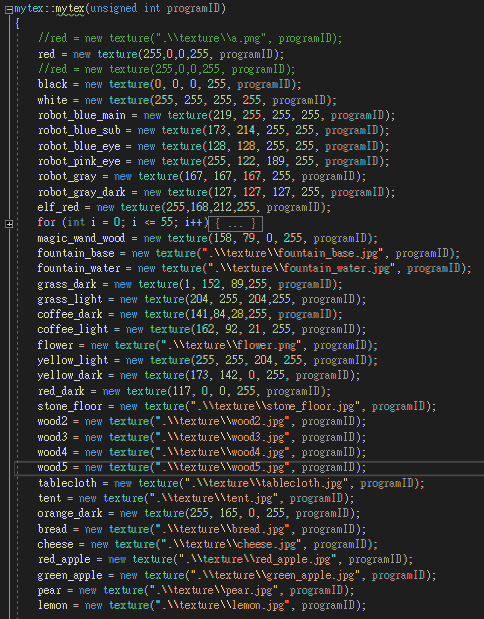
▲移動的時候，加上位移量\*方向向量就行了

* + - 至於方向移動鏡頭的部分，我是判斷滑鼠移動的方向，去決定要往左/右/上/下，xy為當前滑鼠的位置，mouseXY是之前滑鼠的位置。每次移動都去改eyeAngY的角度，或是眼睛位置的高低。



* + shader:
    - 在寫shader架構時，我想了很久，因為每次畫東西，都需傳入點座標，所以我把它寫成一個叫mesh.cpp的檔案，裡面的函式會讀入點座標，然後要畫圖時呼叫mesh類別的物件 -> draw()，即可渲染畫面，但這樣的話我每次要畫都需要重新傳入會很麻煩，所以我又使用一個obj.cpp去存入各種圖案，然後要用時再呼叫，有點像原生glut的感覺。Textute的部分也是一樣，所以我在使用的時候需要先選texture，在選model。
    - 不過缺點時，每次要加入新的圖案，我都必須去改mytex.cpp或是 myobj.cpp的程式碼。到最後就覺得好煩q。mytex.cpp提供四種函式，use是使用那個圖案，可以傳入光罩係數改他的material，或是不傳入，那就是default = 0。至於創造一個texture有2種方法，一種是給rgba，他會生成一塊1\*1的顏色方塊，或是給檔案名，他會去讀圖片。





* + 每次要用都需要去加入q

1. 心得:

經過這一次作業更加了解老師上學期圖學課介紹的內容了，不管是Phong模型，還是物體轉換的pipeline流程。我程式碼的設計是，將要畫的物體切成很多三角形，然後讀入三角形座標。因為這個做法的關係，所以我有使用blender輔助我建物體，blender可以直接將物體輸出成三角形的座標，讓我讀入。不過，因為是第一次用，所以非常不擅長。希望可以開一堂專門教建模的課(x。然後這次作業要特別感謝冠宏，沒有他的幫忙我可能一行也寫不出來，謝謝冠宏。

冠宏除了教我怎麼寫shader外，還教我怎麼寫物件導向和使用blender，冠宏根本是大神。再次謝謝冠宏!

1. 未來展望:

還有三個店鋪沒有賣商品，而且也沒有商人，可能下一次會把他補完。