

利用球型的函數畫出地球、太陽、月亮在使用平移及旋轉跟時間函數的控制做出模擬太陽、月亮、地球的運行情況。

方法:

第一步:

尋找出我所需要的函數，像是畫出球體的函數、平移及旋轉的使用及利用時間函數讓我的圖以動畫的方式移動。

第二步:

球體繪製的部分很快的完成了，但卡在旋轉的軌道上如何運作，這部份則是透過在現實裡實際利用桌面的xyz來發想加上觀察網路上有人完成的程式碼，所以透過時間函數讓我完成了旋轉的動畫呈現。

第三步:

開始把所有球體繪製出來及運行的軌道，跟修改大小調整公轉、自轉的

速度，一個簡單的模擬就完成了。

第四步:

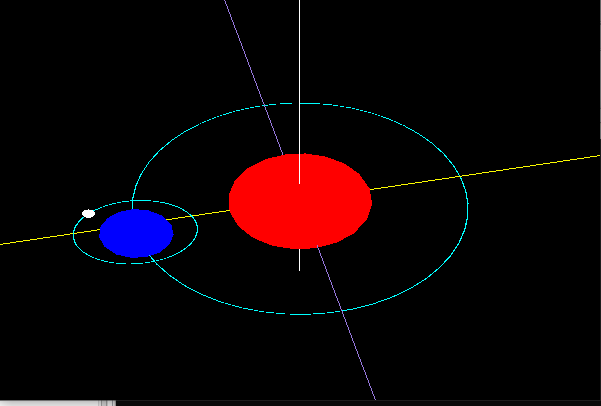
當我以為完成時，回去看到作業的第二頁才發現需要製作xyz軸以及兩個事件處理函式作互動，所以開始加工，由於xyz感覺比較快速做完，所以我做兩3條線來顯示xyz。

第五步:

最後剩下兩個事件處理函式作互動，我就裡用了鍵盤讀取來實作，當我按鍵盤1或2時調整x加或減, 3或4時調整y加或減, 5或6時調整z加或減

利用gluLookAt(x,y,z,0,0,0,0,1,0)在xyz的位置上觀看原點且頭朝上，跟滑鼠的左鍵加快地球公轉，右鍵減慢地球公轉，中鍵回復預設值。

結果:



圖中紅色的球體為太陽、藍色的球體為地球、白色的球體為月亮，淺藍色為進行公轉的軌道、白色線為z軸、黃色線為x軸、紫色線為y軸。

結論:

此次的作業透過旋轉和平移來完成動畫的呈現地球、太陽、月亮間的運行，大部分的內容都不會太難，只有關於旋轉和平移的部分需要全部從頭開始思考他們是如何運動的，因為是三維座標系統，對我來說不向平面的情況容易一看就了解需要多多思考，除了這部分外最讓我感到失敗的地方有關於紋理的部分，這部份是我從還沒開始實作就想做出來的東西，所以從一開始就研究如何使用紋理，自己定義結構並綁定到茶壺有做成功，但是在讀取外部圖片時遇到了問題，需要額外的.h檔運行，所以最終選擇直接實作。