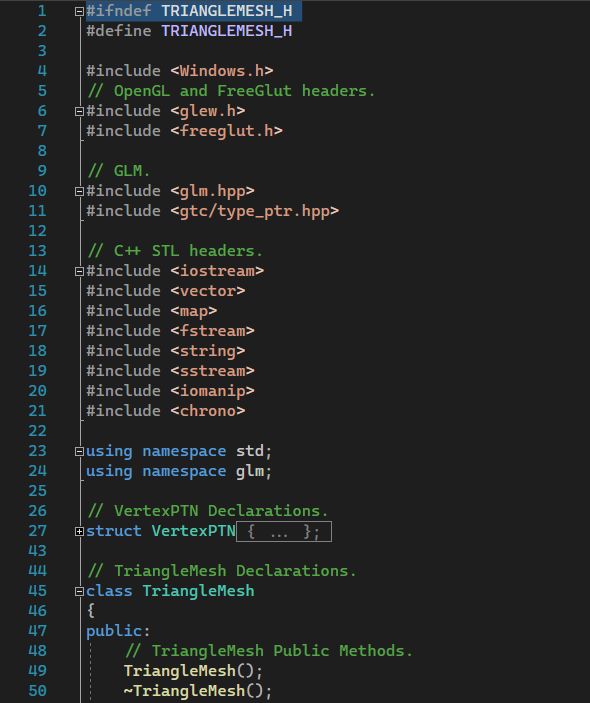
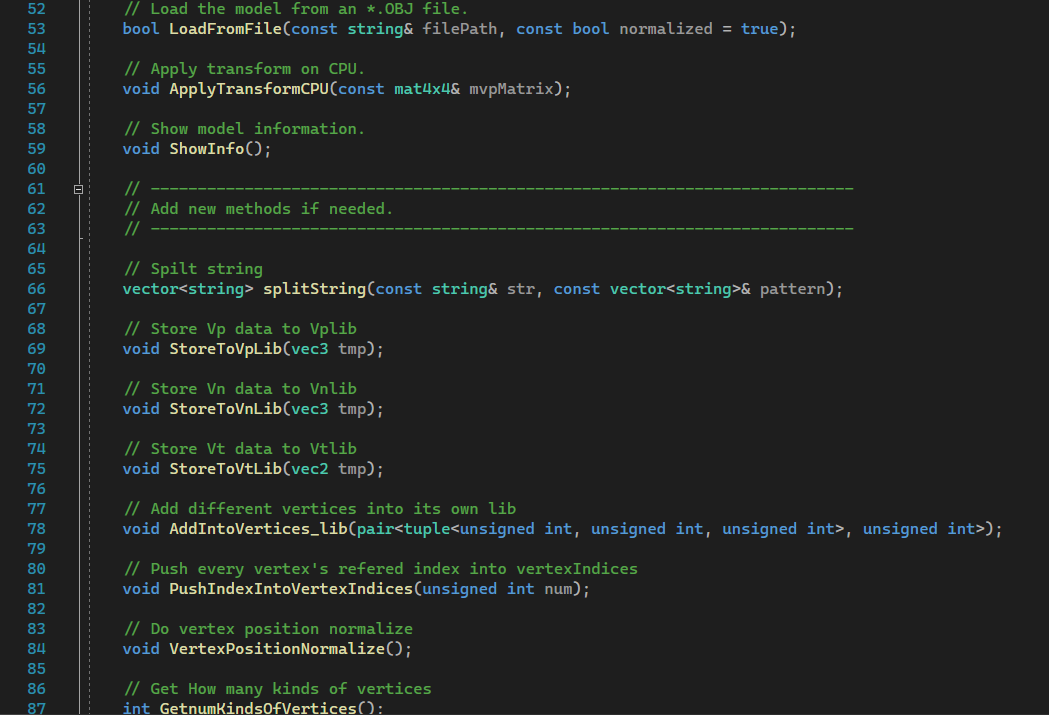
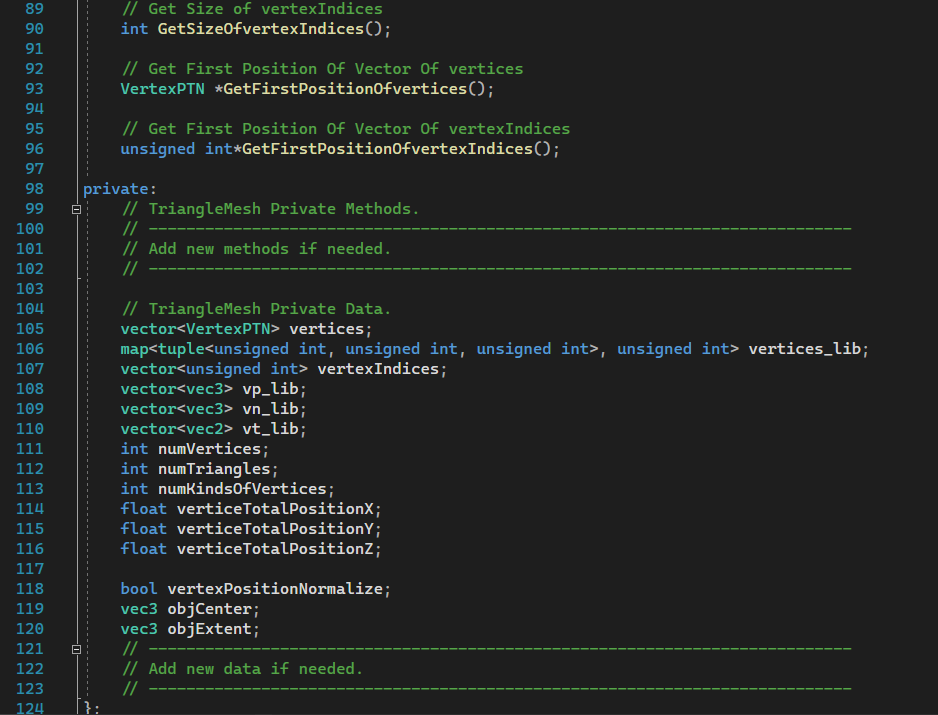
程式註記

1. Trianglemesh.h標頭檔

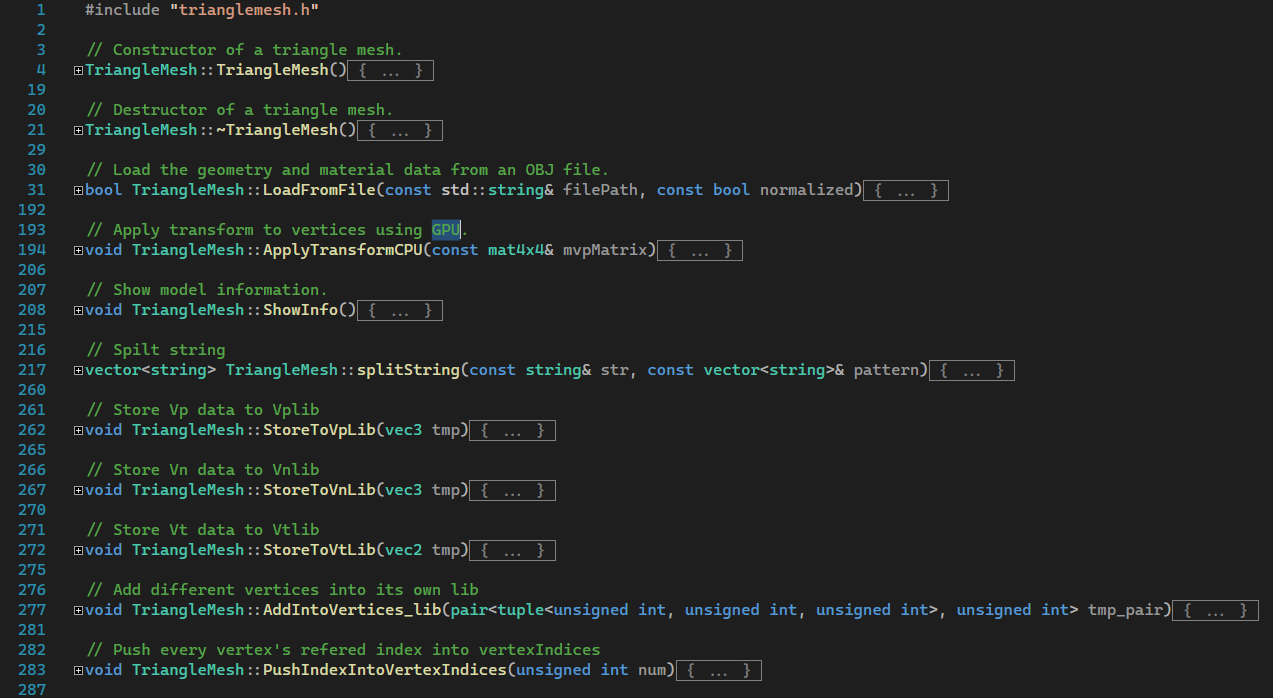


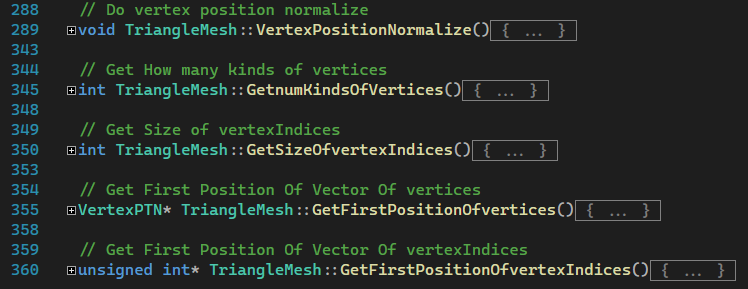




標頭檔主要分為四個部分:

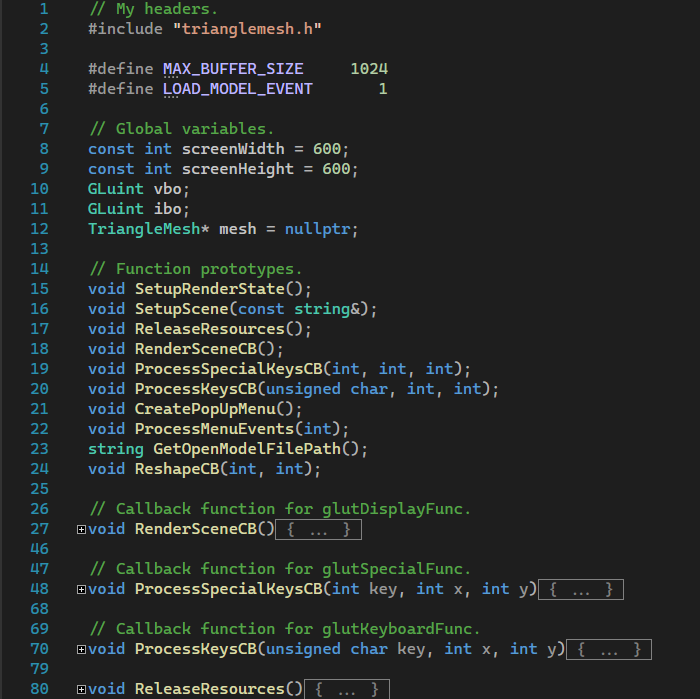
1. 將所有會使用到的library寫入標頭檔中。
2. 建立Vertex的struct，裡面分別記錄每個Vertex的頂點位置、紋理座標以及法向量。
3. 建立Trianglemesh的class，在public中，主要建立各物件所需要的方法，各函式的功能都已標住在程式碼中。
4. 在private中，主要負責建立供各物件存取的變數。vp\_lib、vn\_lib、vt\_lib分別將讀入v、vn、vt分別存入各自的vector。vertices\_lib；vertices\_lib存入不同的vertex避免相同的vertex不斷塞入記憶體造成浪費；vertexIndices紀錄所存入的vertex其對應到vertices\_lib的哪一個位置。verticeTotalPositionX、Y、Z分別記錄所有vertex的頂點在各三軸中的總和(作正規化時會用到)。vertexPositionNormalize用來判斷是否已經做過正規化。
5. Trianglemesh.cpp實作檔

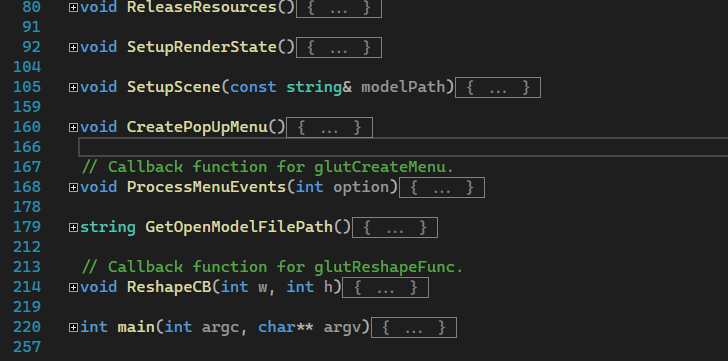




1. LoadFromFile -> 實作將Obj檔讀入
2. ApplyTransformCPU -> 實作vertex在CPU和GPU之間的轉換
3. ShowInfo -> 顯示物件的相關資訊
4. splitString -> 實作檔案內容的分割處理
5. StoreToVpLib -> 實作將vp內容存入vp\_lib
6. StoreToVnLib -> 實作將vn內容存入vn\_lib
7. StoreToVpLib -> 實作將vt內容存入vt\_lib
8. AddIntoVertices\_lib -> 實作將不同的vertex存入library中
9. PushIndexIntoVertexIndices -> 實作將每一個vertex其所對應到library位置的index存到vector中
10. VertexPositionNormalize -> vertex頂點的正規化
11. GetnumKindsOfVertices -> 得到有幾種不同的vertex
12. GetSizeOfvertexIndices -> 得到有多少個vertex
13. GetFirstPositionOfvertices -> 得到vertex library(vertices\_lib)的第一個陣列位置
14. GetFirstPositionOfvertexIndices -> 得到第一個儲存vertex index的陣列位置

ICG2022Hw1.cpp主程式





1. RenderSceneCB -> 繪圖
2. ProcessSpecialKeysCB -> 按下指定鍵，使模型產生相對應的結果
3. ProcessKeysCB -> 按下esc終止程式
4. ReleaseResources -> 終止程式時，釋放資源的實作
5. SetupRenderState -> 初始化繪圖的設定
6. SetupScene -> 初始化場景的設定
7. CreatePopUpMenu -> 建立彈出的列表
8. ProcessMenuEvents -> 實作處理列表事件的程序
9. GetOpenModelFilePath -> 得到物件開啟檔案的路徑
10. main -> 主函式