

OpenGL with GLUT 的設定與執行

OpenGL Setup Tutorial 是在 Windows 10 作業系統並使用 Visual Studio 2017 版為開發環境下的設定。其他版本亦可嘗試使用本設定，若設定完不能執行，請聯絡助教。

1. 請至“課程網站 - 檔案下載”下載 OpenGL 環境建立所需檔案：openGL.zip 並將壓縮檔中的三個資料夾(dll, include, lib)內的所有檔案複製到以下資料夾

(1)將 lib 資料夾內的所有檔案放於以下資料夾中

C:\Program Files (x86)\Microsoft Visual Studio\2017\Professional\VC\Tools\MSVC\14.16.27023\include

(2) include 資料夾內包含一個資料夾 gl，將此資料夾放於以下資料夾中

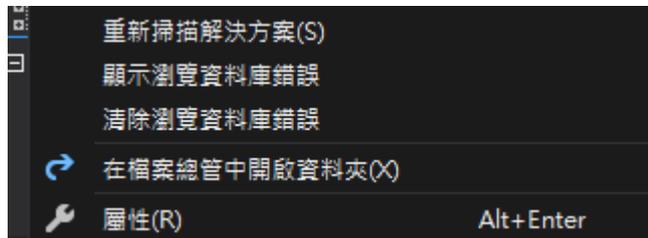
C:\Program Files (x86)\Microsoft Visual Studio\2017\Professional\VC\Tools\MSVC\14.16.27023\lib\x64

Tips：上面的(1)和(2)所展示的資料夾路徑，可能會根據你的 Visual Studio 安裝版本或設定而有所不同，請你照以下步驟(a~d)，確認你真正 include 和 lib 資料夾位置：

- a. 新增或開啟任意 C++專案，於專案(Project)右鍵選擇屬性(Properties)

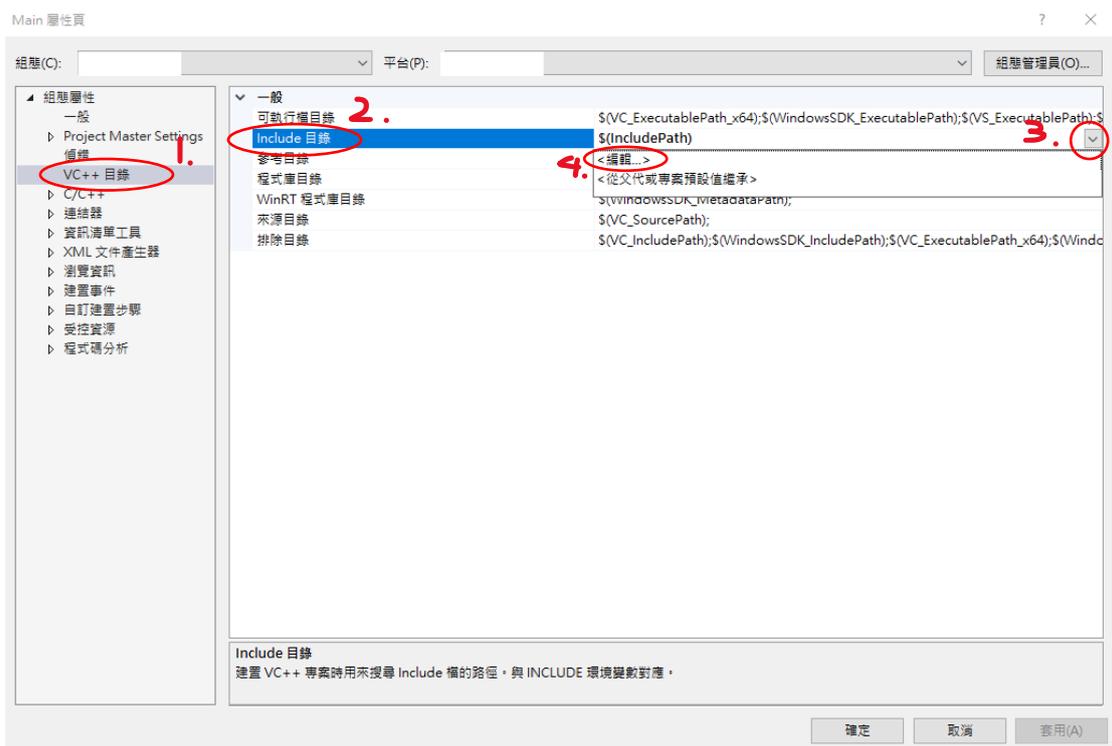


注意：上圖中，上方是你的“方案”，下方才是你的“專案”。兩者左邊的圖示不同，請留意。

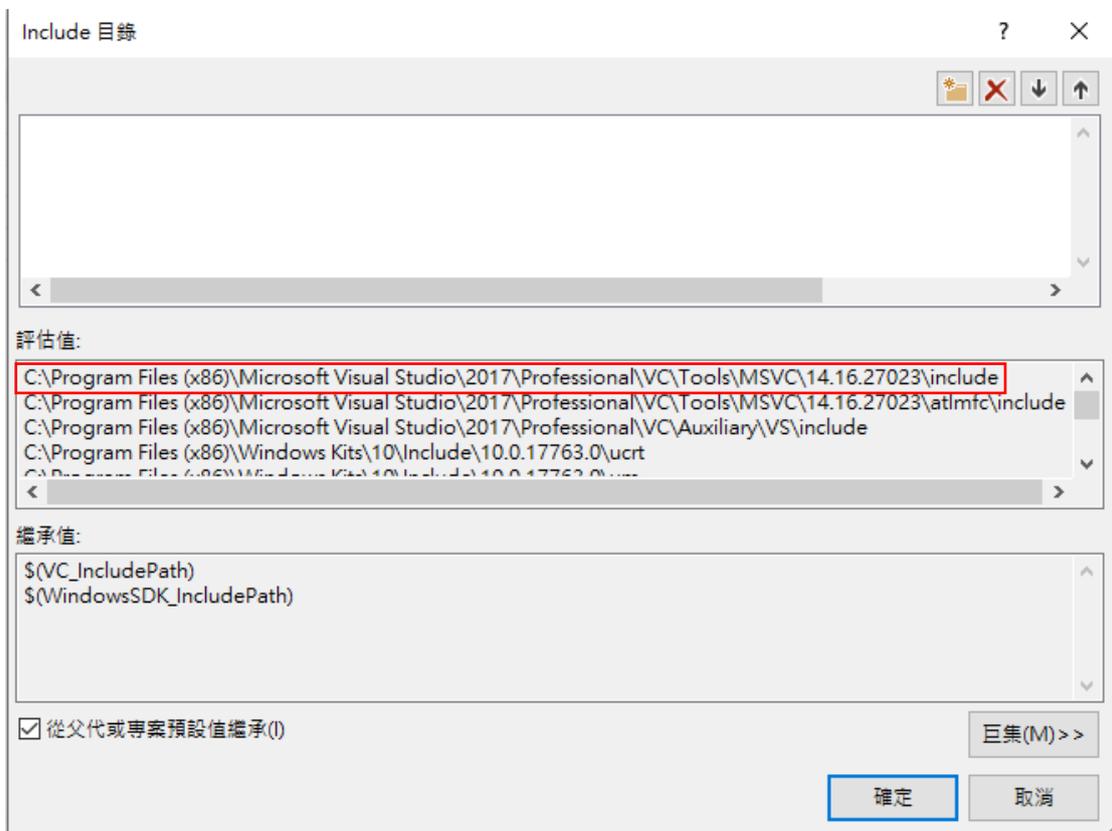


按下滑鼠右鍵後，最下一層應該就會是“屬性”

b. VC++ 目錄，Include 目錄->箭頭->編輯(<Edit>)



c. 第一欄即為 Include 目錄位置



d. lib 目錄則在步驟 b 選擇"include 目錄"下方兩格的"程式庫目錄"(Library Directories)，可用相同方法取得。

(3) dll 資料夾內應有三個檔案，全部放於以下資料夾中

若為 64 位元電腦則放在 C:\Windows\SysWOW64

若為 32 位元電腦則放在 C:\Windows\System32

Tips：通常現在電腦應該都是 64 位元，不放心的話可以查查看。

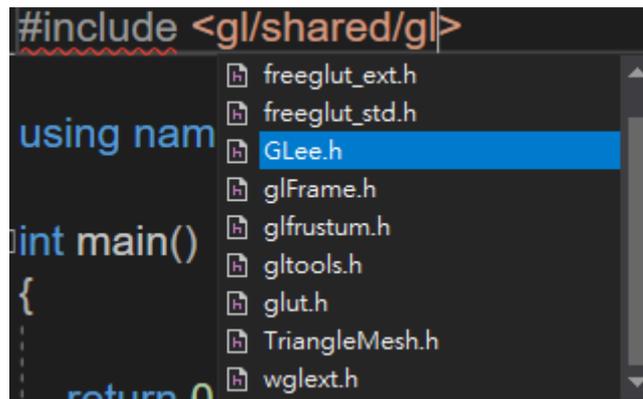
2. 在程式中引入標頭檔，有以下兩種做法，擇一使用：

```
#include <windows.h>
#include <gl/gl.h>
#include <gl/glu.h>
#include <gl/glaux.h>
#include <gl/glut.h>
```

or

```
#include <gl/shared/gltools.h>
#include <gl/shared/math3d.h>
```

Tips：請手動輸入，如果前面的設定正確的話，輸入到一半時編譯器應該會出現提示，告訴你哪些檔案是可以被 include 的，如下圖。



3. 連結函式庫 LIB 檔：opengl32.lib, glaux.lib, glu32.lib, glut32.lib

同樣有兩種做法，擇一使用：

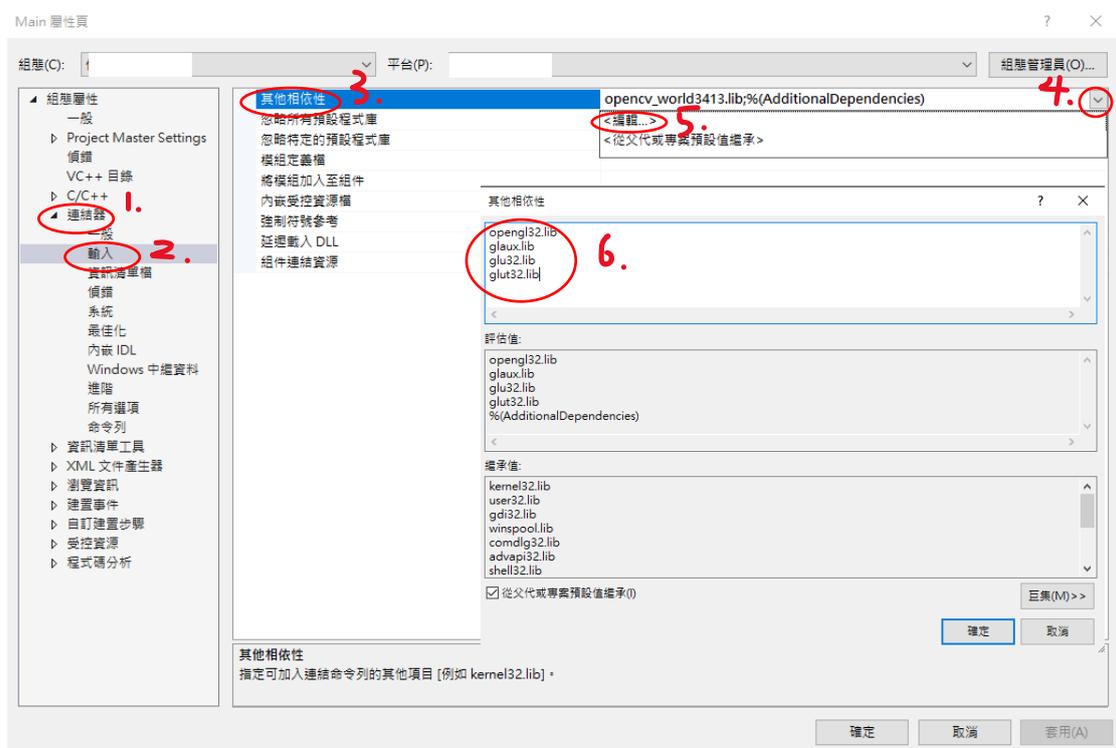
(1) 在方案總管視窗中，在目前的專案(Project)的標籤上按滑鼠右鍵，在 Context Menu 上選擇屬性(Property)。



如同步驟 1.(2)a.，右鍵後選擇屬性



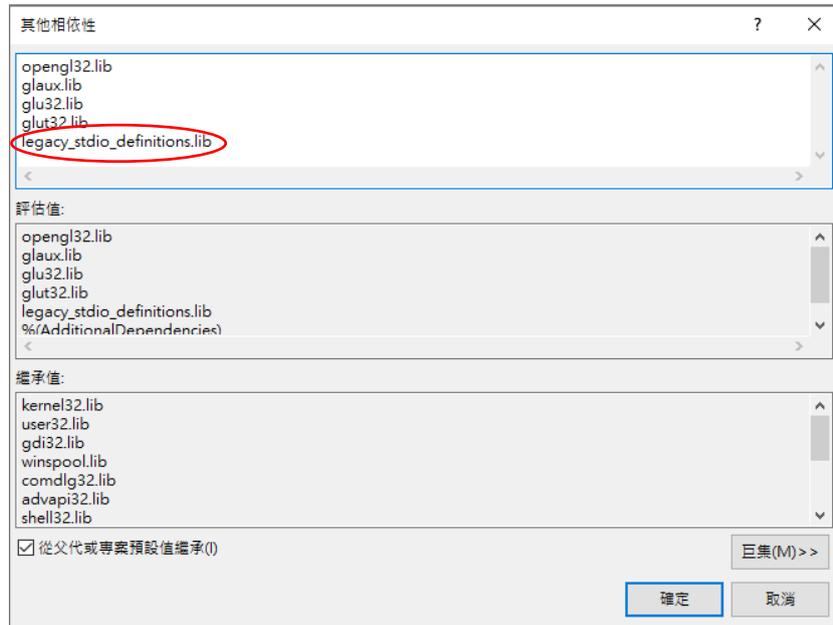
在對話盒的左邊，選擇「連結器(Linker)」，再選擇子項目「輸入」，然後在「其他相依性」中按右側箭頭，再按編輯進入其他相依性編輯框，”手動”輸入上列出的四個函式庫並用 Enter 隔開。



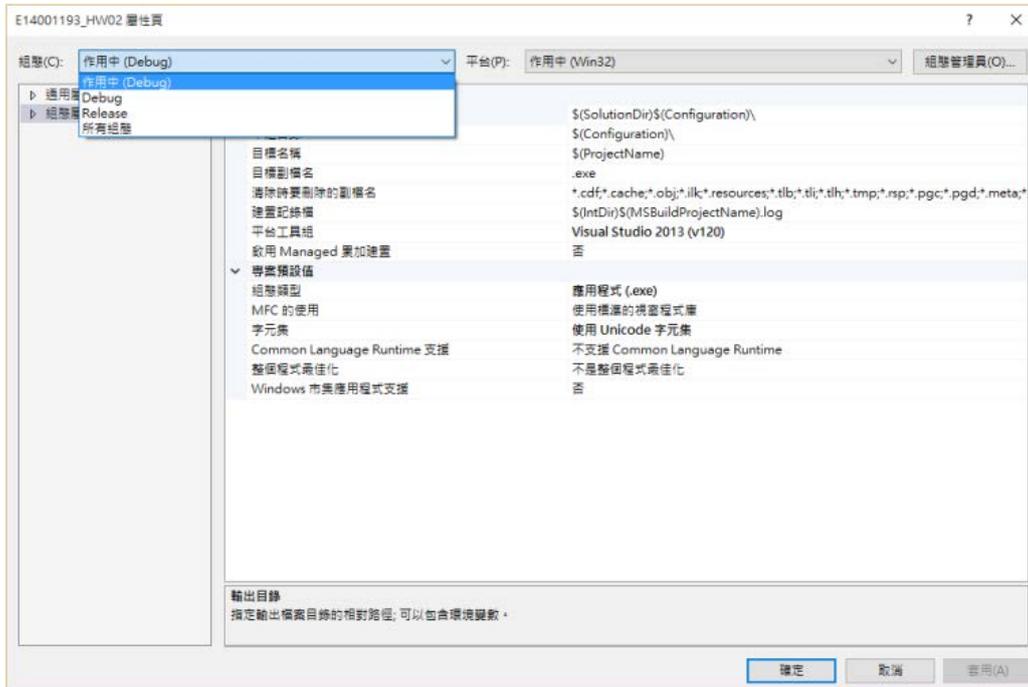
(2) 在原始碼檔案的開始部分(在引入標頭檔的部分之後)，寫入以下巨集指令：

```
#pragma comment(lib, "opengl32.lib")  
#pragma comment(lib, "glu32.lib")  
#pragma comment(lib, "glaux.lib")  
#pragma comment(lib, "glut32.lib")
```

4. 連結 legacy_stdio_definitions.lib，方法與步驟 3.的方法(1)相同



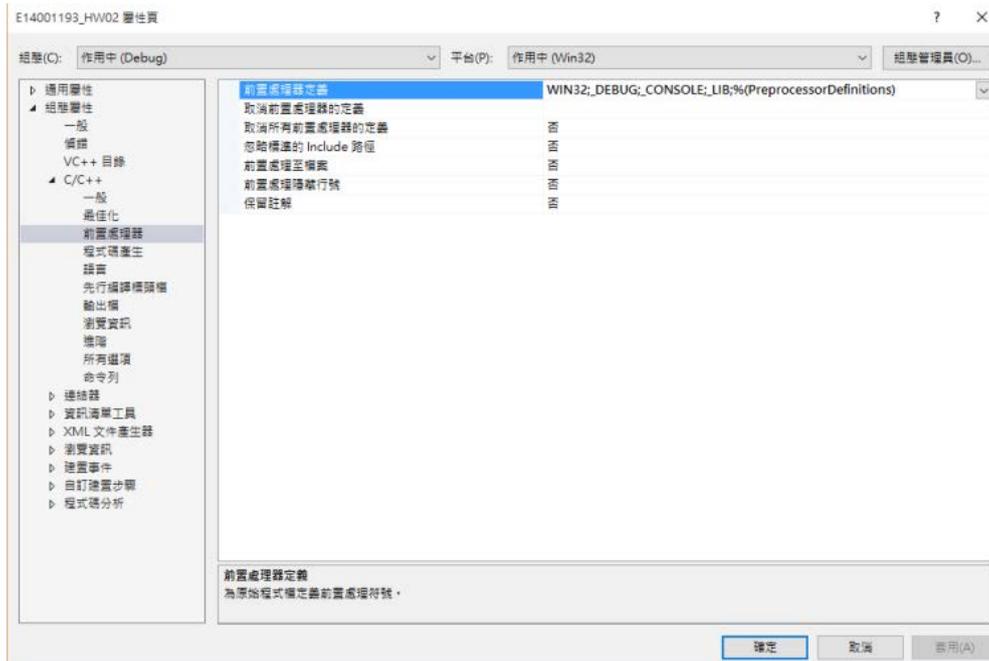
5. 在方案總管視窗中，在目前的專案(Project)的標籤上按滑鼠右鍵，在 Context menu 上選擇屬性(Property)，會產生屬性頁之對話盒。而將左上角【組態】的部分會依其模式調整成 Debug 或 Release，要記得兩個模式都要加入的部分。



(1)在【組態屬性】→【C/C++】→【前置處理器】內的【前置處理器定義】中加入屬性值：

【Debug】模式下：WIN32;_DEBUG;_CONSOLE

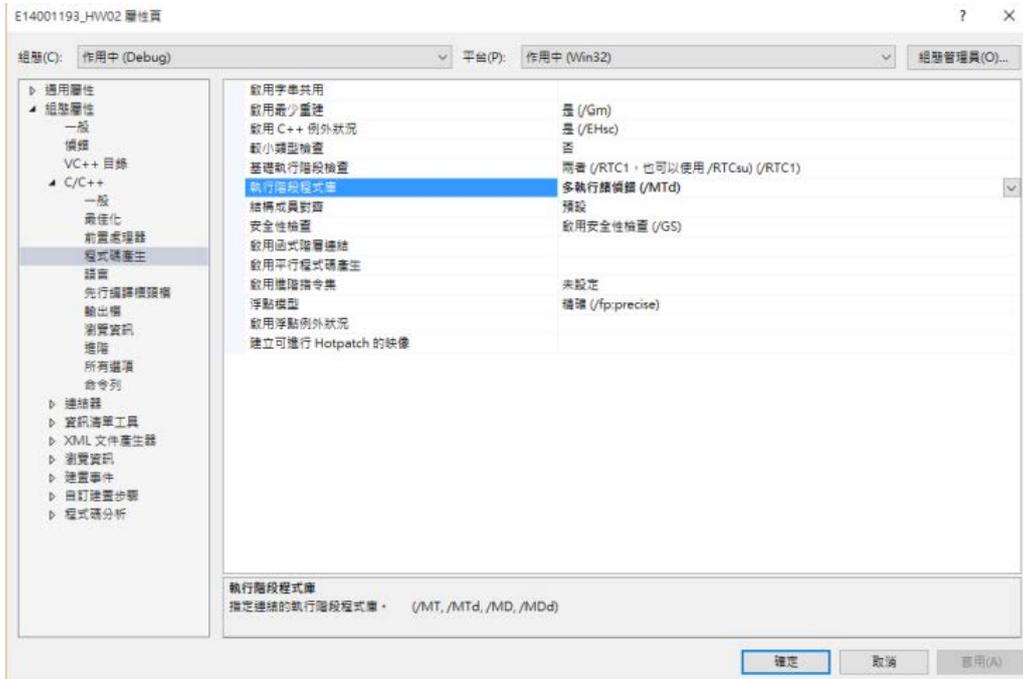
【Release】模式下：WIN32;NDEBUG;_CONSOLE



(2) 在【組態屬性】→【C/C++】→【程式碼產生】內的【執行階段程式庫】中的屬性改成：

【Debug】模式下：多執行緒偵錯 (/MTd)

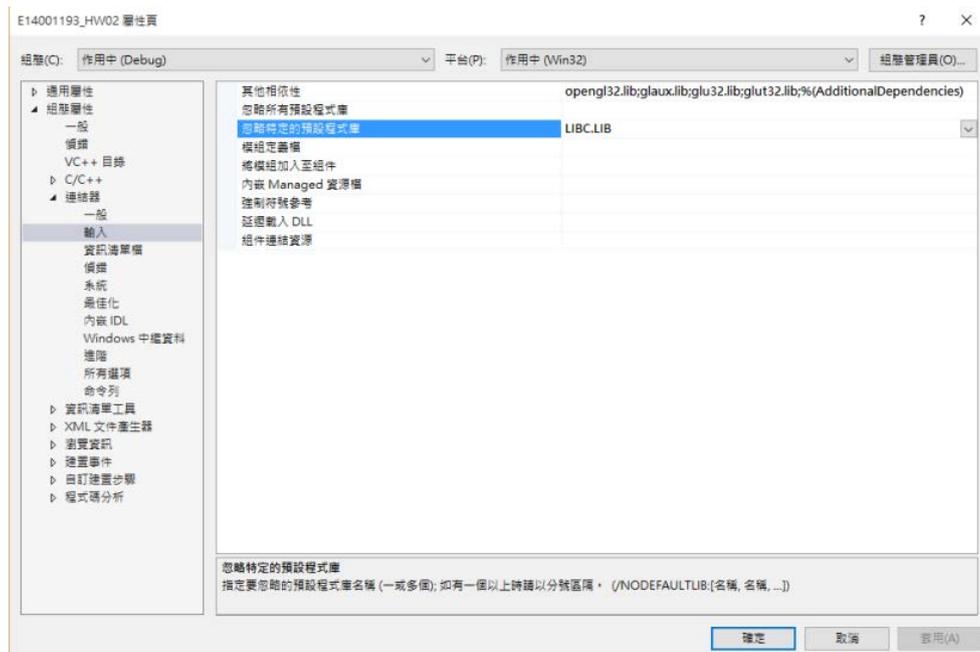
【Release】模式下：多執行緒 (/MT)



(3) 在【組態屬性】→【連結器】→【輸入】內的【忽略特定程式庫】中加入需忽略的 lib

【Debug】模式下：LIBC.LIB

【Release】模式下：LIBC.LIB



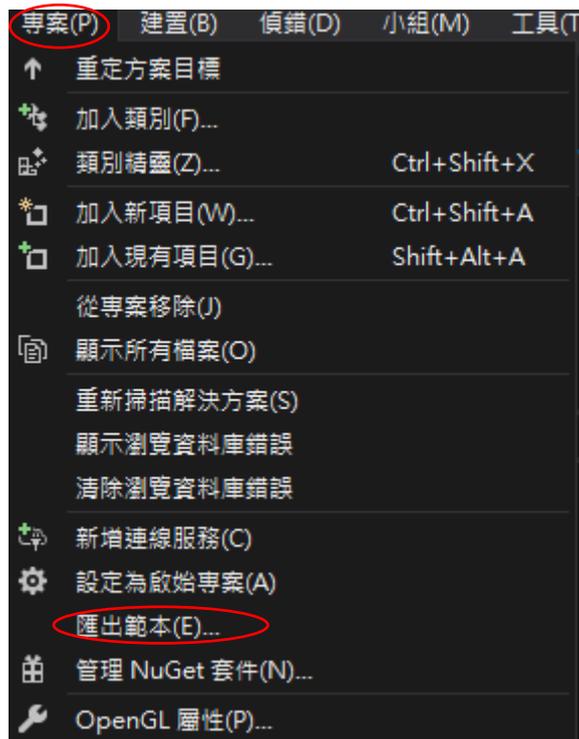
到此環境建立完畢，可以開始練習程式了

其他注意事項：

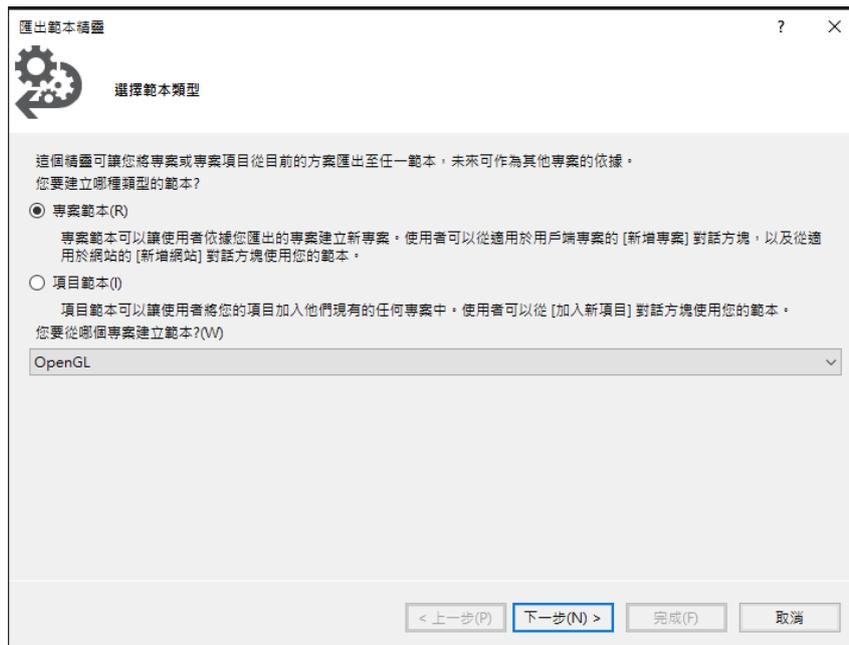
1. 每次新開一個“專案”，這些環境就必須重新建立一次(第 1 步不用，因為那是把檔案放入你的電腦裡，只需做一次)。

或是你也可以將專案匯出成範本，以後可以直接新增 OpenGL 專案：

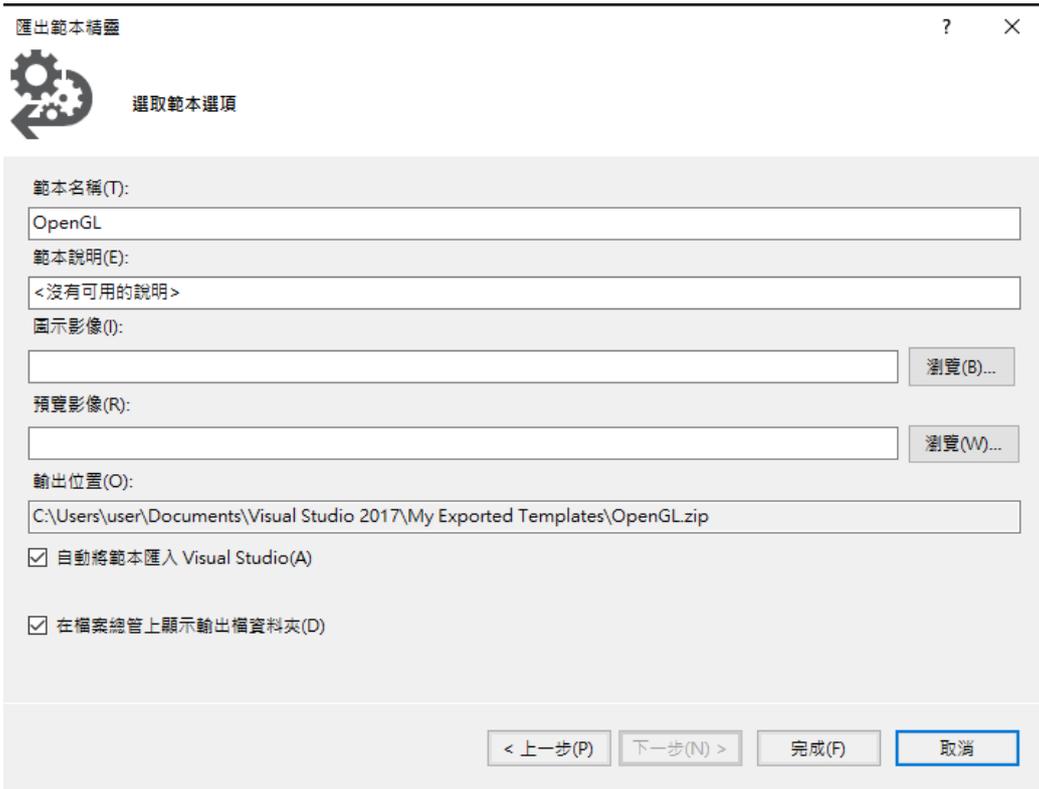
(1)確認環境安裝完畢之後，點選左上方「專案」，點選「匯出範本」。



(2)選擇“專案範本”，確認你要從哪個專案匯出範本，按下一步



(3)自己取範本名稱，範本說明可有可無，其他選項不需更動



匯出範本精靈

選取範本選項

範本名稱(T):
OpenGL

範本說明(E):
<沒有可用的說明>

圖示影像(I):
瀏覽(B)...

預覽影像(R):
瀏覽(W)...

輸出位置(O):
C:\Users\user\Documents\Visual Studio 2017\My Exported Templates\OpenGL.zip

自動將範本匯入 Visual Studio(A)

在檔案總管上顯示輸出檔資料夾(D)

< 上一步(P) 下一步(N) > 完成(F) 取消

(4)按下完成後，以後建立新專案時，此範本就會出現，可直接沿用已經設定好的環境和已打上的程式碼

2. 將範例程式碼複製到你自已建的專案時，請捨棄原本範例的標頭檔：

```
#include "../..../shared/gltools.h" // OpenGL toolkit
```

如上圖那行，請不要把原本的 include 帶到新的專案去