**实验1.1 OpenGL图形编程环境配置**

1. **实验目的**
2. 掌握Visual Studio Community 2015集成开发环境的安装。
3. 了解和掌握OpenGL的环境配置。
4. 掌握OpenGL工程项目的建立和基本设置。
5. **实验环境**

**硬件要求：**

主流配置，最好配备独立显卡。

**软件环境：**

操作系统：Windows 7/Windows 10。

集成开发环境：Microsoft Visual Studio Community 2015。

开发库：FreeGLUT库，GLEW库。

后续的实验环境与此相同，以后不再赘述。

1. **理论背景**

**OpenGL简介**

OpenGL（全称Open Graphics Library）是指定义了一个跨编程语言、跨平台的编程接口规格的专业的图形程序接口。它用于三维图像（二维的亦可），是一个功能强大，调用方便的底层图形库。OpenGL是行业领域中最为广泛接纳的 2D/3D 图形 API，其自诞生至今已催生了各种计算机平台及设备上的数千优秀应用程序。

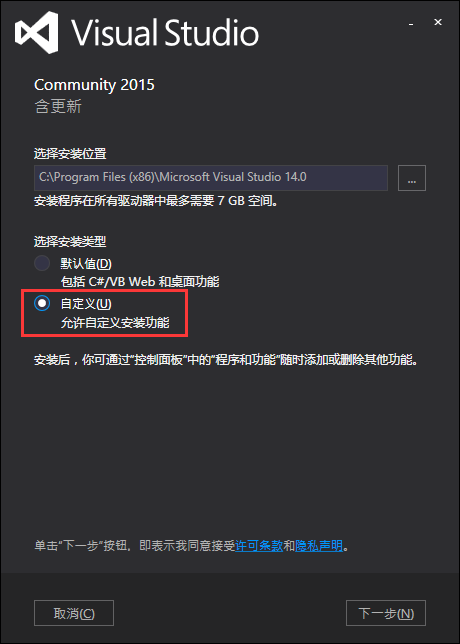
**FreeGLUT简介**

GLUT 的全称为 OpenGL Utility Tookit。最初 Mark Kilgard 为了使 OpenGL 应用程序结构能够真正独立于窗口系统而构思了 GLUT 库。但由于 GLUT 库很早就已经不再维护更新，便诞生了与 GLUT 库兼容的开源实现版本 FreeGLUT。FreeGLUT 库主要用来创建窗口，初始化 OpenGL 的环境，管理用户鼠标、键盘的输入，以及让 OpenGL 程序变得简单和提升可移植性。

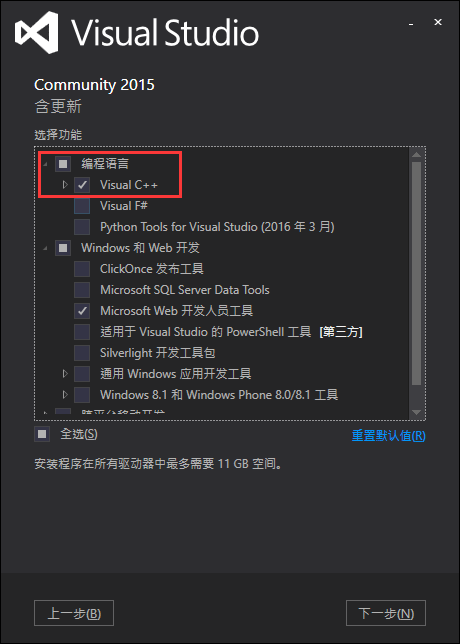
**GLEW简介**

GLEW 的英文全称为 The OpenGL Extension Wrangler Library，是一个基于 OpenGL、跨平台、开源的 C++ 扩展库。GLEW 提供了有效的运行机制，以寻找和识别你的平台所支持的全部 OpenGL 高级扩展函数。也就是说，只要包含一个 glew.h 头文件，你就能使用以 gl、glu、glext 等开头的全部函数。GLEW 已被测试能支持目前流行的各种操作系统，包括 Windows、Linux、Mac OS X、FreeBSD、Irix 和 Solaris。

1. **实验内容**
2. **Visual Studio Community 2015 的安装**
3. 打开网址<https://www.visualstudio.com/downloads/download-visual-studio-vs>，选择Visual Studio 2015，勾选ISO，点击下载。
4. 打开iso文件中的vs\_community.exe进行安装，好选择在默认的安装位置，并勾选自定义，点击下一步。



1. 必须勾上编程语言中的Visual C++，其余根据各自需求选择，并点击下一步，直到开始安装。

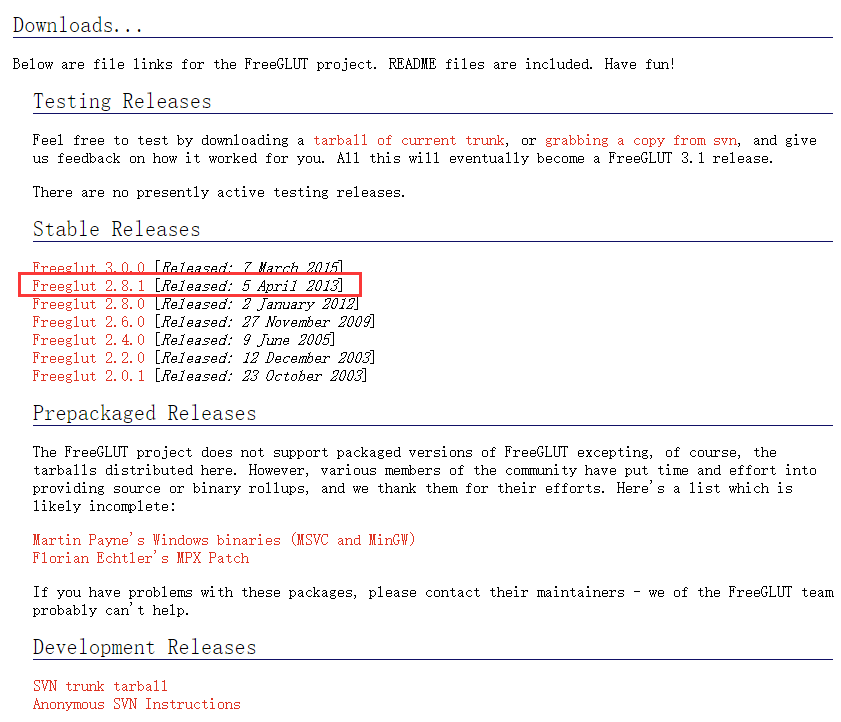


1. 安装完成。若安装过程中出现错误，可在控制面板中对VS Community 2015点击“更改”，并选择修复，继续安装直到完成。
2. **OpenGL的环境配置**

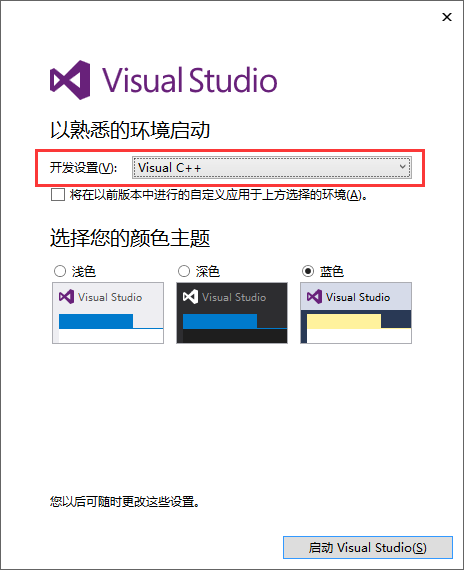
在实验课的上机实验中都依赖于FreeGLUT和GLEW这两个第三方库。因此需要分别针对自己的操作系统环境编译出这两个库。

* FreeGLUT库的编译

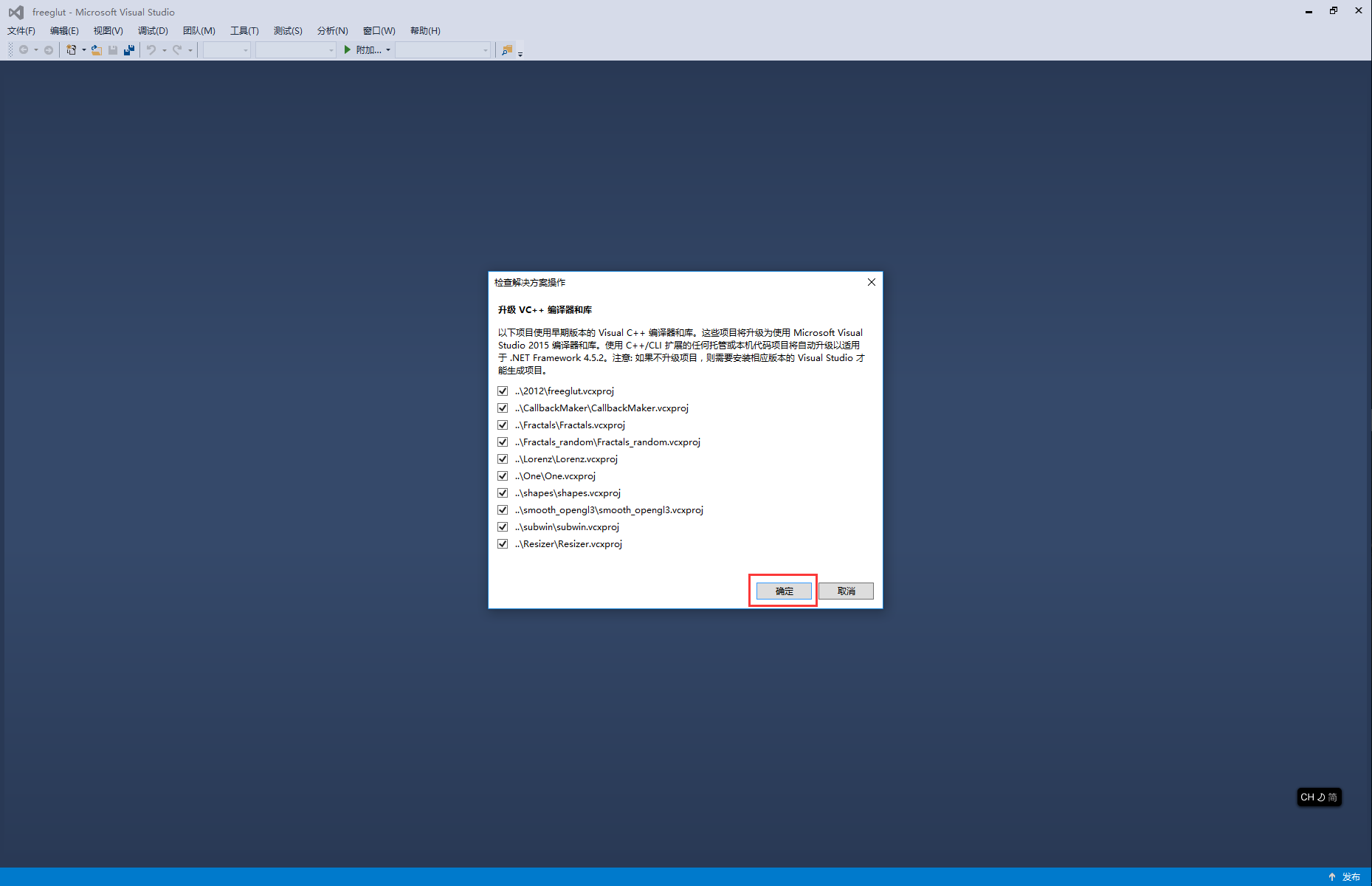
1. 打开网址 <http://freeglut.sourceforge.net/index.php#download>
2. 选择Freeglut 2.8.1版本下载。



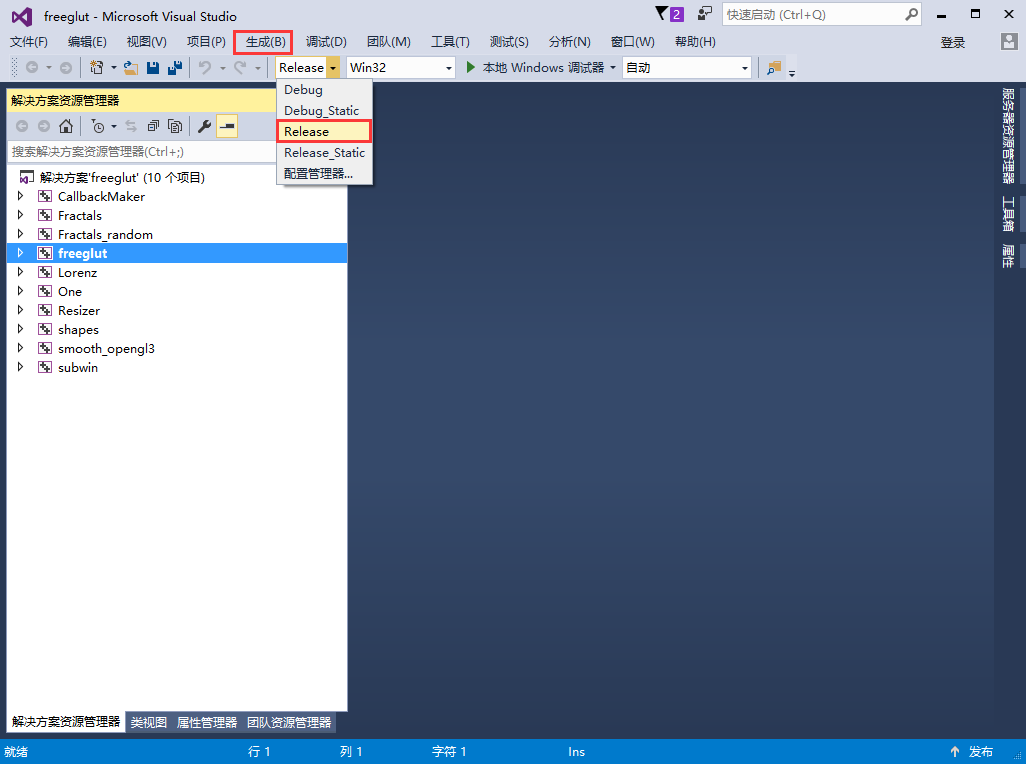
1. 解压后得到freeglut-2.8.1文件，进入VisualStudio，选择版本2012，打开解决方案freeglut.sln。
2. 初次打开，开发设置选择Visual C++。



1. 对所有由vs2012创建的freeglut项目，点击“确认”升级VC++编译器和库，以支持Visual Studio 2015的编译。



1. 选择Release版本，打开“生成”菜单，点击“生成解决方案”

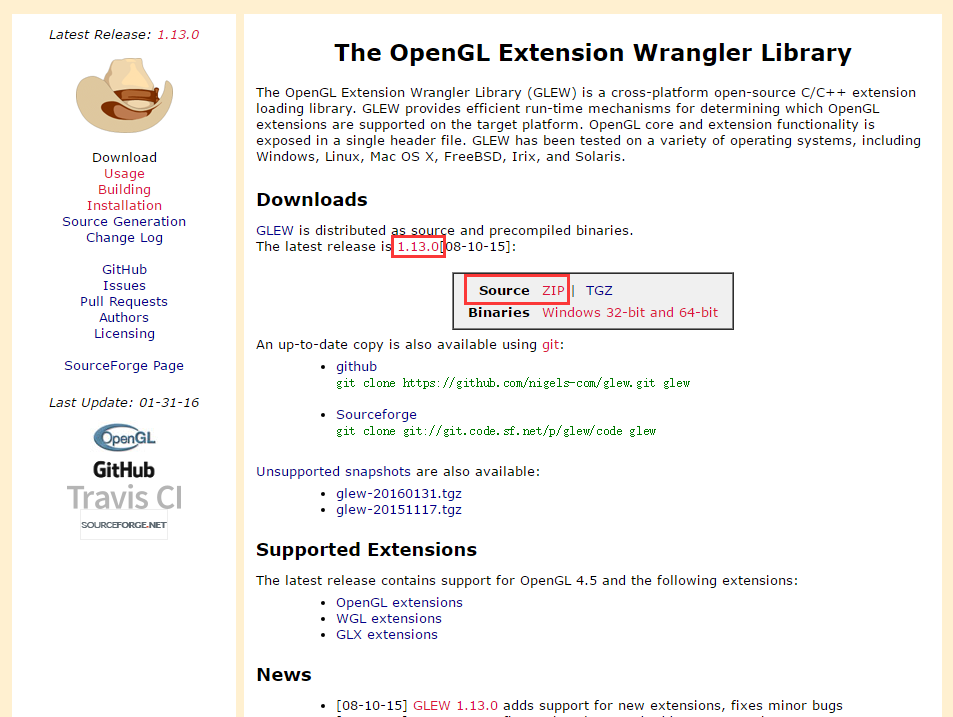


1. 编译成功后，返回到freeglut-2.8.1根目录，将include下的GL文件夹复制到C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\include文件夹。
2. 返回到freeglut-2.8.1目录，把lib\x86里面的freeglut.lib复制到C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\lib中。
3. 把lib\x86里面的freeglut.dll复制到C:\Windows\SysWOW64文件夹中。

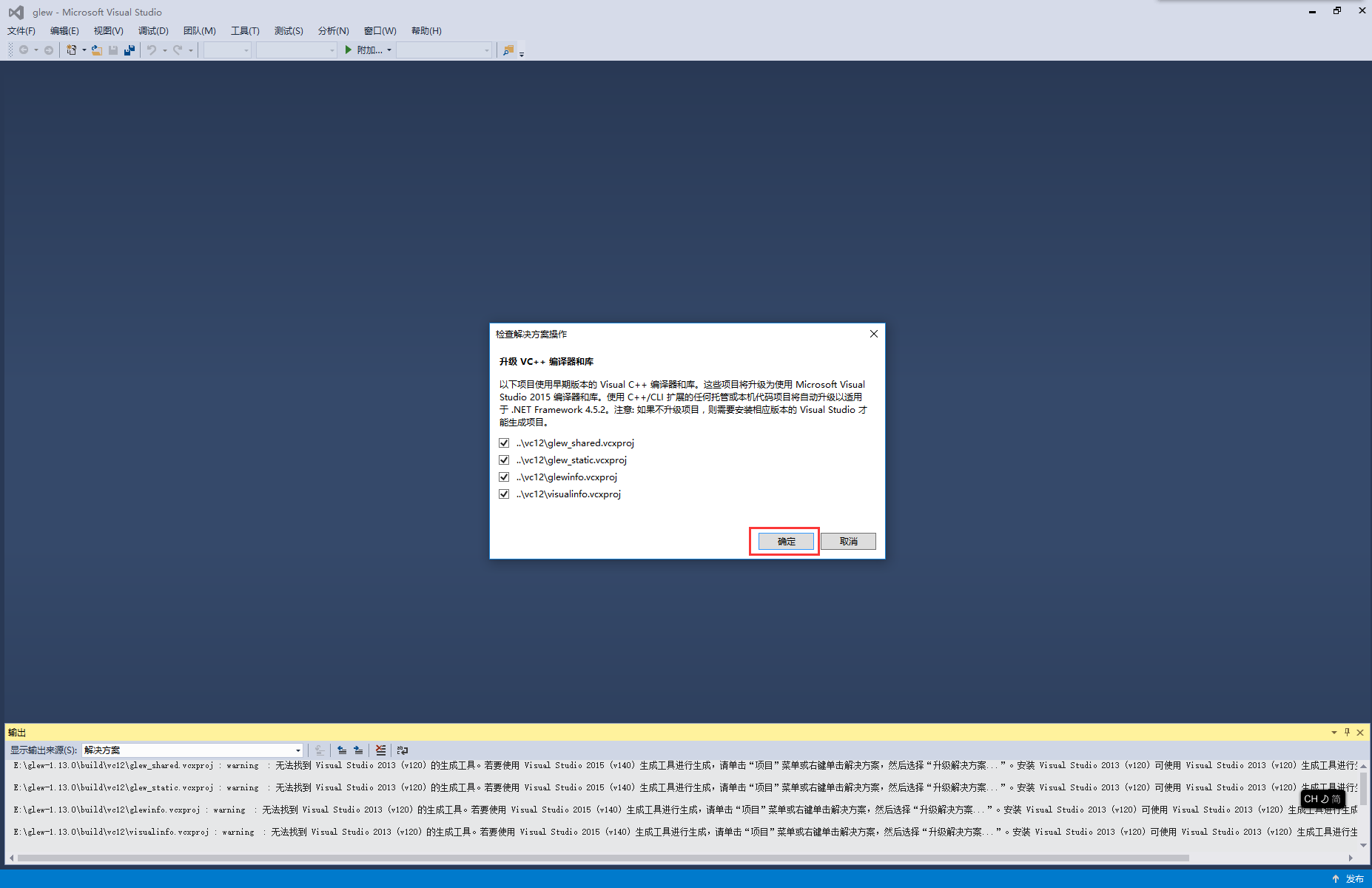
至此，FreeGLUT的配置完成。

* GLEW库的编译

1. 打开网址 http://glew.sourceforge.net/
2. 选择1.13.0版本的源代码ZIP格式压缩包对库进行编译（也可以选择下方针对Windows平台已编译好的库文件，则可以跳过编译阶段。但为了兼容稳定性，推荐下载源代码编译）。



1. 解压后进入glew-1.13.0文件夹。在build文件夹中选择vc12，打开glew.sln解决方案。
2. 下面与配置freeglut的过程类似。会提示对由更低版本Visual Studio创建的glew项目进行升级，点击“确认”升级VC++编译器和库，以支持Visual Studio 2015的编译。



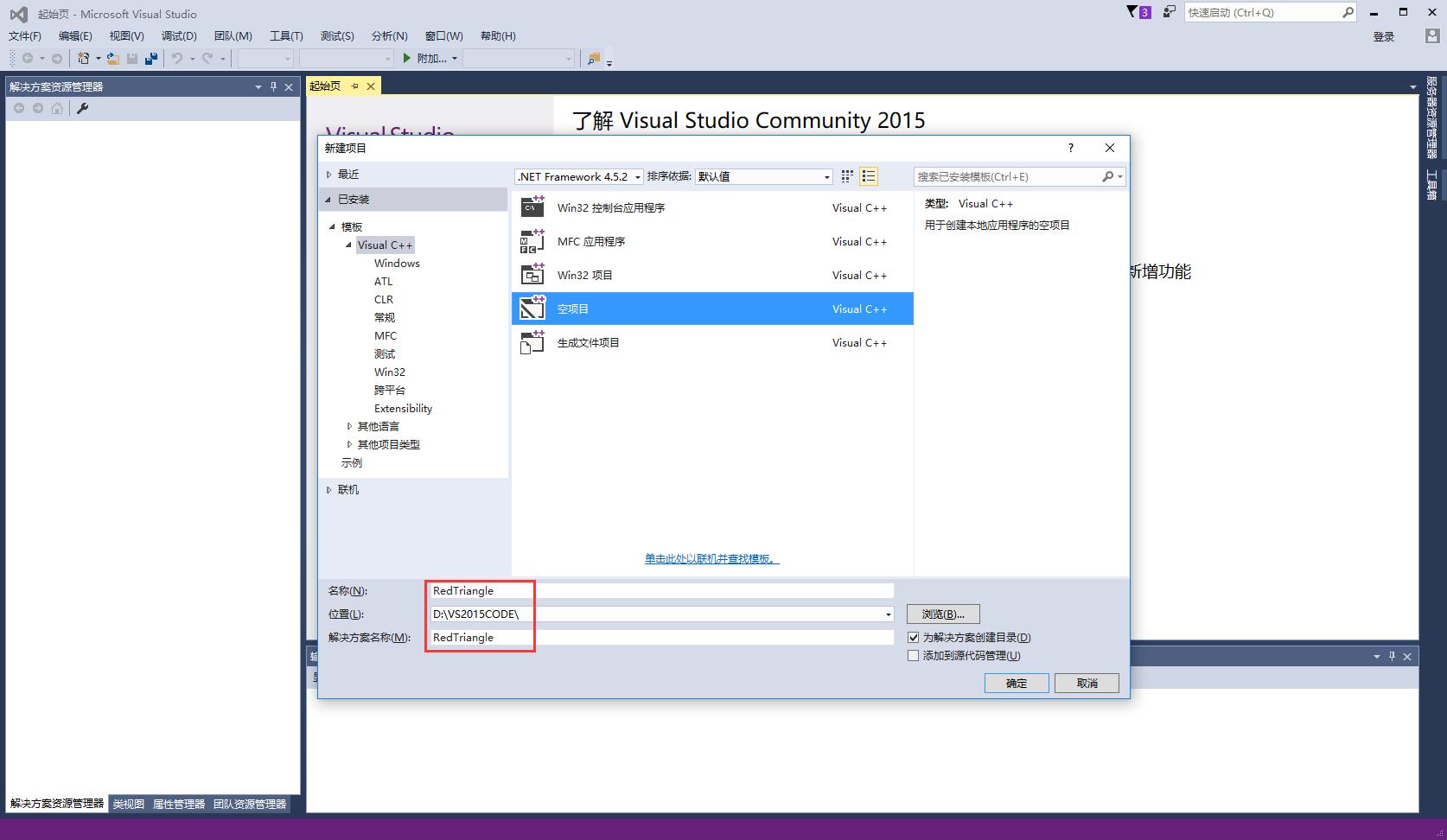
1. 选择Release版本，打开“生成”菜单，点击“生成解决方案”。
2. 编译成功后，返回到glew-1.13.0根目录，将include下的GL文件夹整个复制到C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\include文件夹，与前面的GL文件夹合并。
3. 返回glew-1.13.0目录，把lib\Release\Win32中的glew32.lib复制到C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\lib里。
4. 把bin\Release\Win32里面的glew32.dll复制到C:\Windows\SysWOW64文件夹中。

至此，glew库的配置完成。

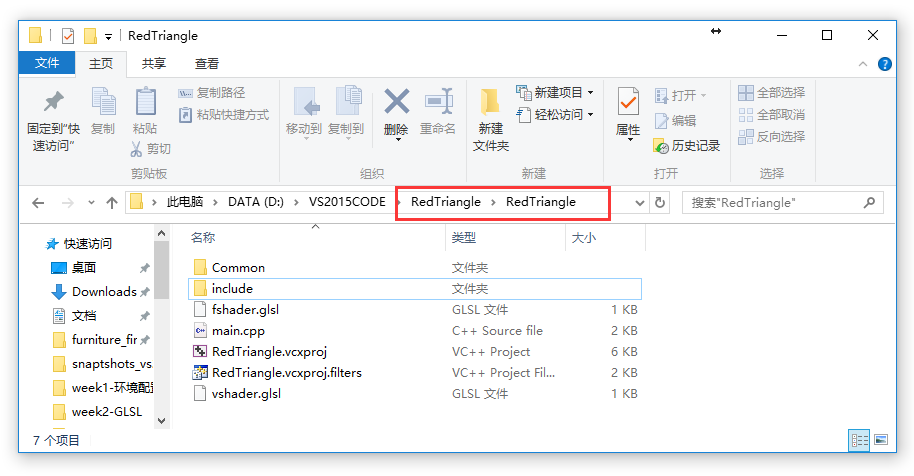
1. **OpenGL基本工程项目的建立和设置**

上机课程中的项目都同样用到了课本教材中所使用的基础工具类头文件，因此在建立项目前，需要提前准备好教材所提供的Common和Include这两个文件夹。

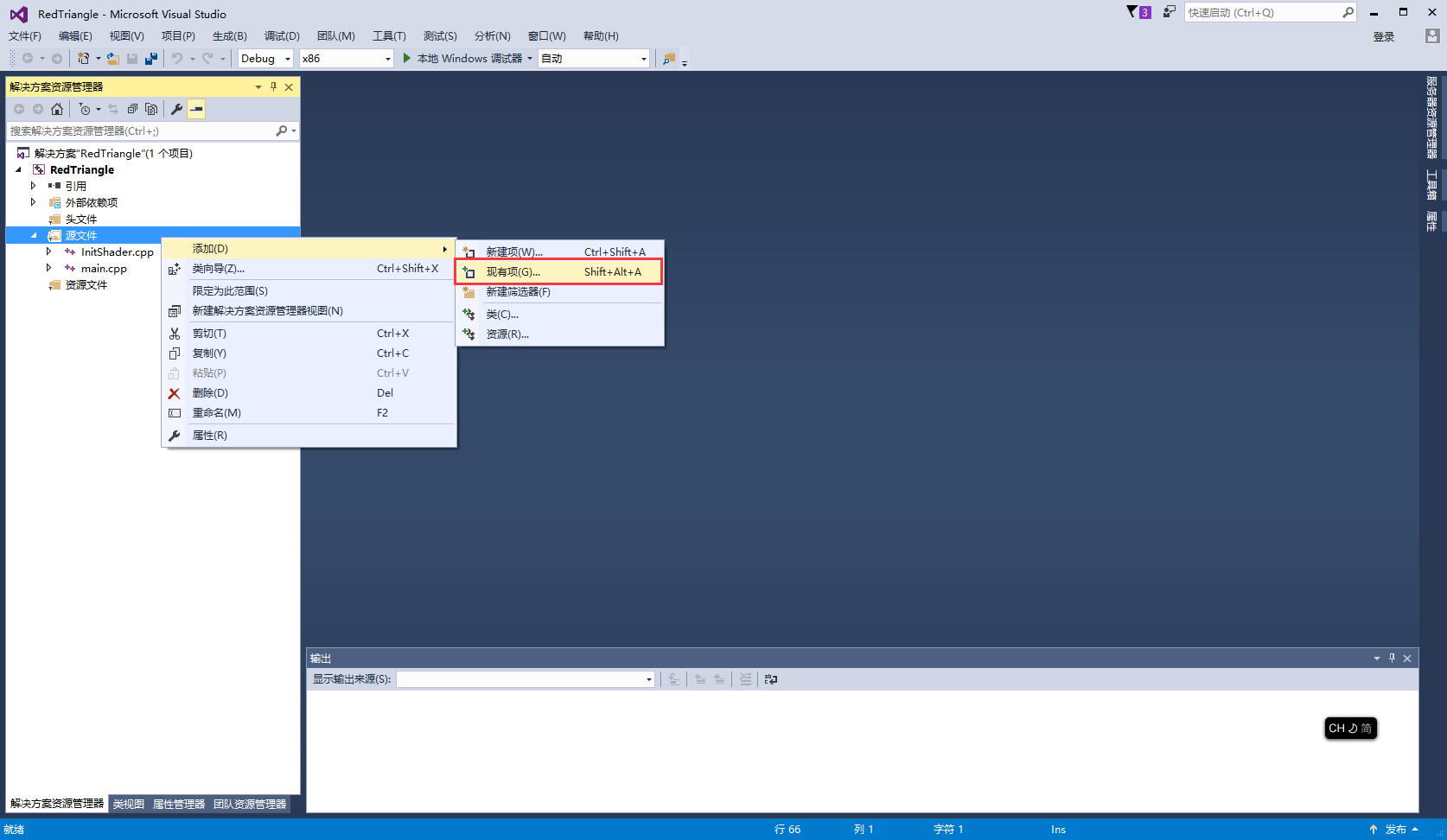
1. 新建项目，选择即空项目，将项目命名为RedTriangle。



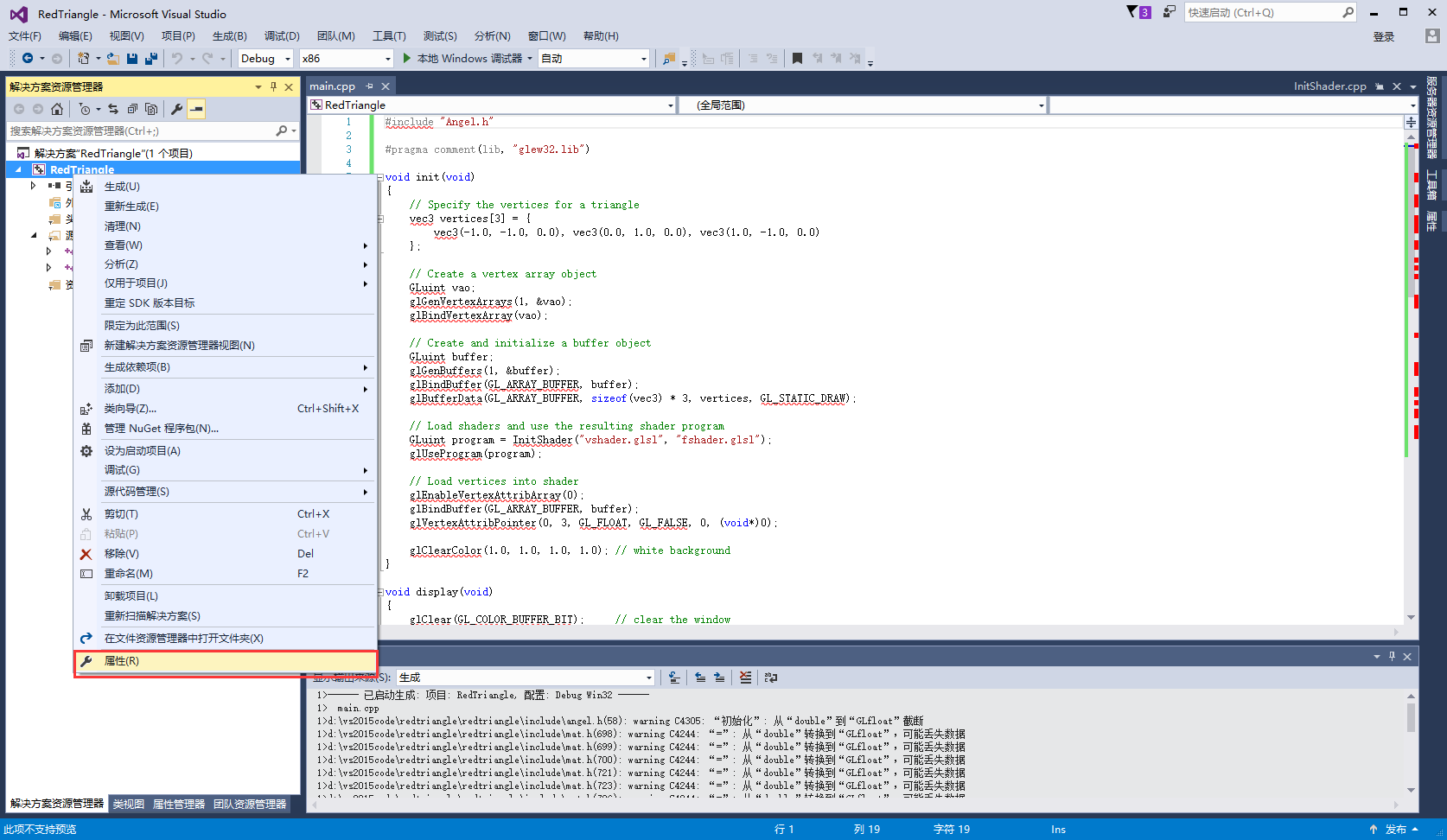
1. 拷贝包含基本工具类函数的Common和Include这两个文件夹，以及本次实验所提供的main.cpp，vshader.glsl，fshader.glsl，将这5项放入RedTriangle的子文件夹RedTriangle中。此时的目录结构如下。



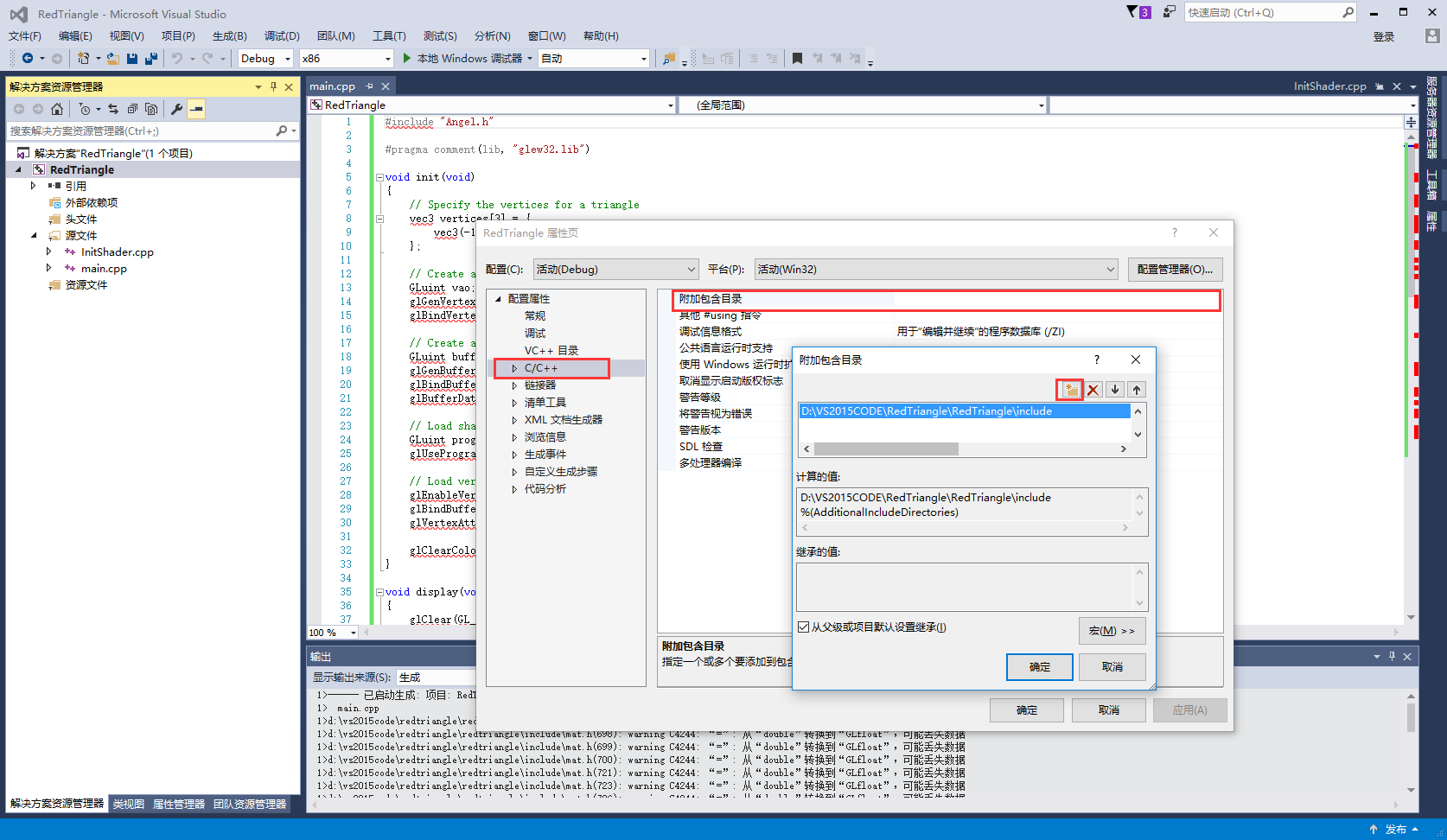
1. 对项目中的源文件点击右键，选择“添加”，点击“现有项”。选择main.cpp和Common文件夹中的InitShader.cpp，将它们添加到工程项目中。



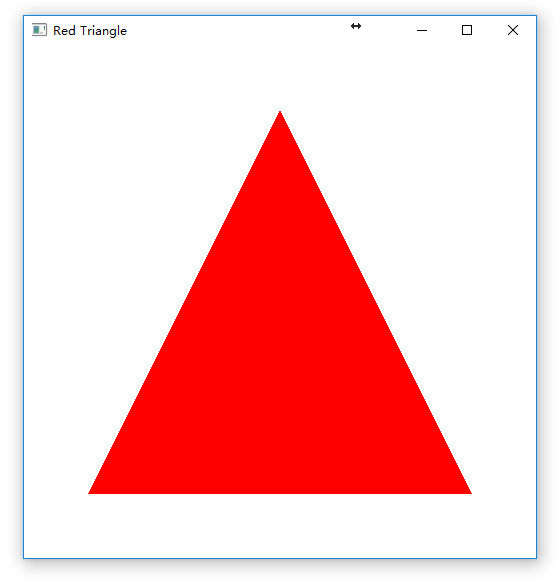
1. 右键点击RedTriangle项目，选择“属性”。



1. 在左侧选择“C/C++”，在“附加包含目录”中添加include文件的路径（本例为D:\VS2015CODE\RedTriangle\RedTriangle\include）。



1. 点击“本地Windows调试器”，运行即可获得下面的结果。



如果程序编译成功，并显示出一个红色的三角形，则代表OpenGL程序和环境的配置正确！OpenGL程序中各文件的含义和具体绘制实现将在下一次实验中进行讲解。