

## 一、Visual Studio 2019 安装

下载网址：(下载 Community 2019 版本)

<https://visualstudio.microsoft.com/zh-hans/vs/>

## 二、Qt 5.13.0 安装

1、 下载地址：

[http://download.qt.io/official\\_releases/qt/5.13/5.13.0/](http://download.qt.io/official_releases/qt/5.13/5.13.0/)

2、 下载如下版本：

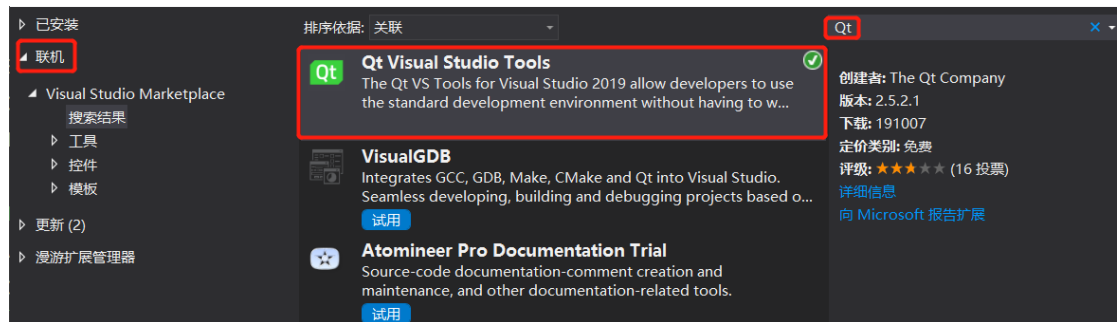
Name	Last modified	Size	Metadata
↑ Parent Directory		-	
submodules/	18-Jun-2019 13:34	-	
single/	18-Jun-2019 13:38	-	
qt-opensource-windows-x86-5.13.0.exe	18-Jun-2019 13:41	3.7G	<a href="#">Details</a>
qt-opensource-mac-x64-5.13.0.dmg	18-Jun-2019 13:39	3.3G	<a href="#">Details</a>
qt-opensource-linux-x64-5.13.0.run	18-Jun-2019 13:37	1.4G	<a href="#">Details</a>
md5sums.txt	18-Jun-2019 13:54	207	<a href="#">Details</a>

3、 双击安装，安装的组件全选(没有 qt 账号的注册一个)

## 三、Qt VS Tools 扩展安装（可能需要翻墙）

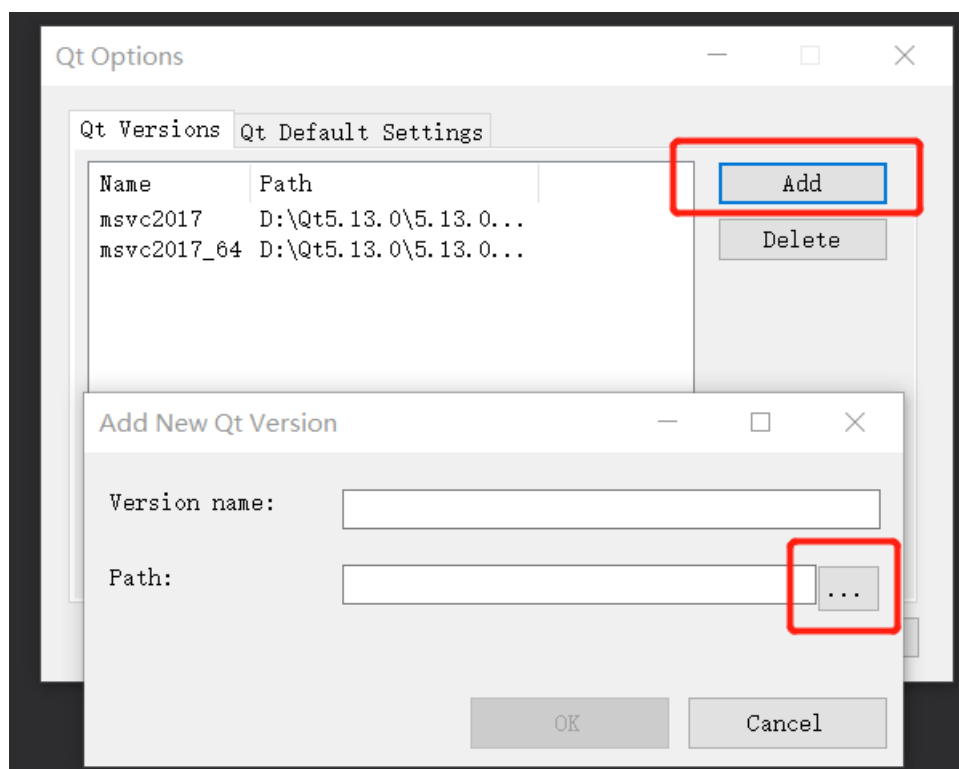
1、 启动 Visual Studio 2019，如需登录可跳过，选择“继续但无需代码”来启动。

2、 菜单栏点击“扩展->管理扩展”，先点“联机”，再输入框搜索 Qt，安装 Qt Visual Studio Tools，安装完成后关闭 VS 以应用 Qt 扩展。

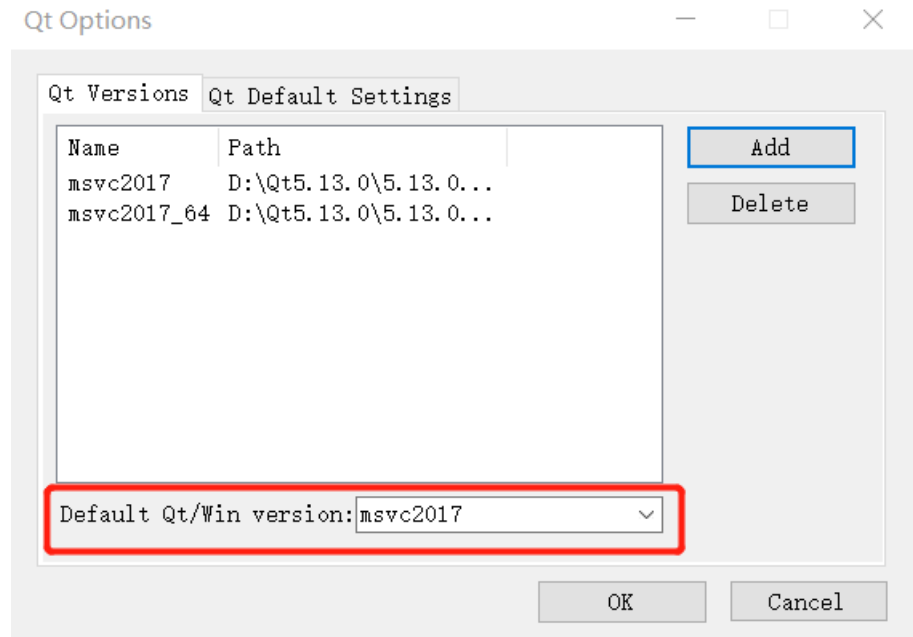


### 3、配置 Qt 插件

- 启动 VS 进入主界面，进入扩展中的 Qt VS Tools 点开 Qt Options。
- 进入以下界面，点击 Add，浏览到 Qt 编辑器的安装路径，默认为 C:\Qt\Qt5.13.0\5.13.0\msvc2017 和 C:\Qt\Qt5.13.0\5.13.0\msvc2017\_64（实际路径按照自己的安装路径为准），可自定义编辑器名称 (Version name)。



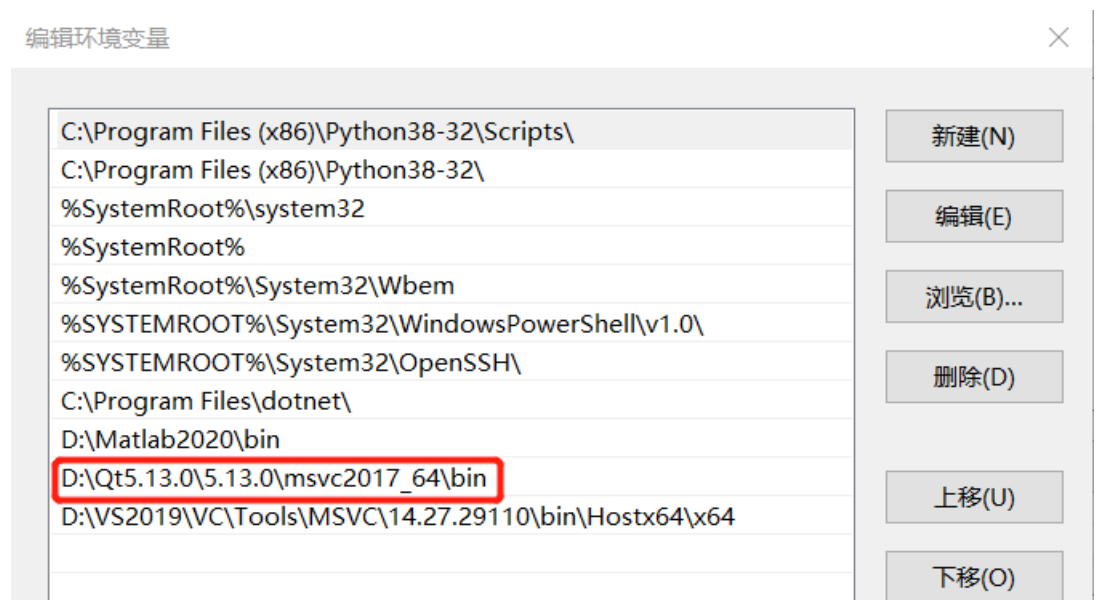
- 添加编辑器完成后可以设置默认的编辑器。



4、 安装完成

#### 四、 环境配置

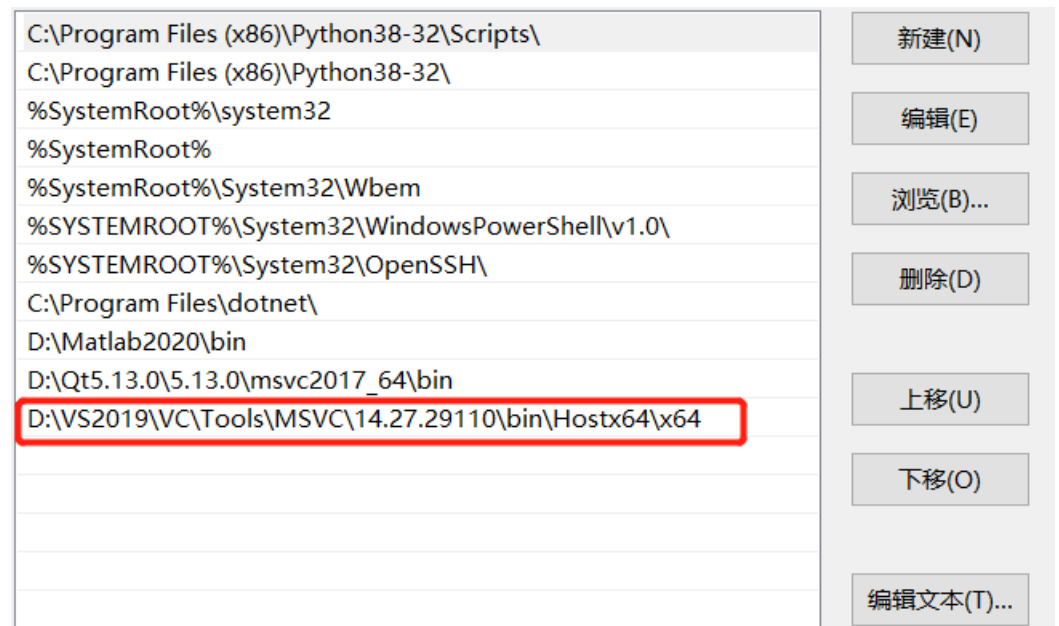
1、 qmake 使用： 新建系统环境变量



2、 启动 cmd 进入 Demo 文件夹， 键入命令： qmake -tp  
vc 生成 VS 工程文件， 可能会出现以下错误：

```
D:\skh\课程\计算机图形学 (TA) \project\Demo>qmake -tp vc
Project ERROR: Cannot run compiler 'cl'. Output:
=====
=====
Maybe you forgot to setup the environment?
D:\skh\课程\计算机图形学 (TA) \project\Demo>
```

解决办法：将 cl.exe 的路径加入系统环境变量



完成后建议关掉 cmd，重新打开后键入 qmake -tp vc  
命令操作

```
D:\skh\课程\计算机图形学 (TA) \project\Demo>qmake -tp vc
Info: creating stash file D:\skh\课程\计算机图形学 (TA) \project\Demo\qmake.stash
D:\skh\课程\计算机图形学 (TA) \project\Demo>
```

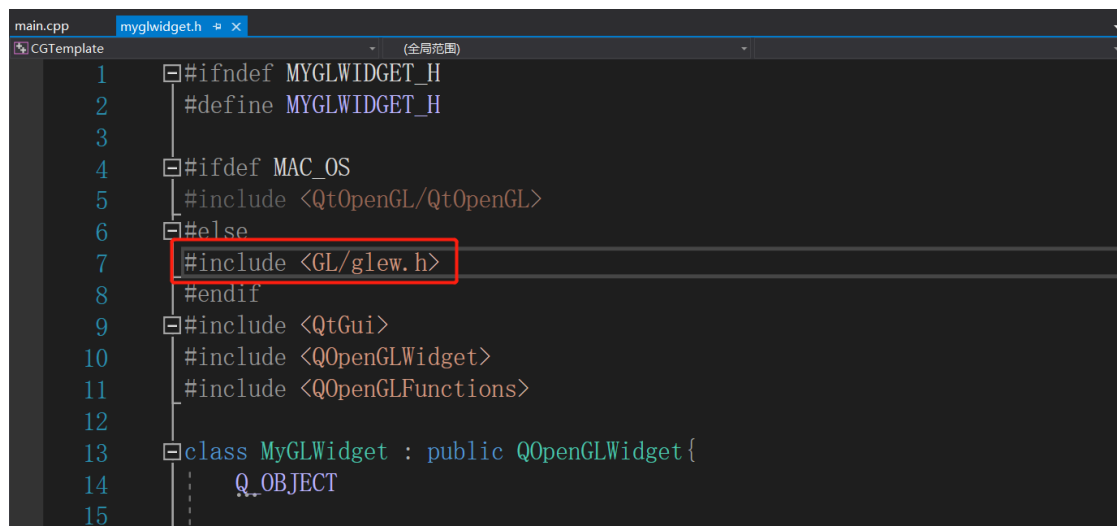
附：建议路径不要出现中文，否则会出现上图的？

## 五、Demo 运行

- 1、 qmake 后生成文件，在 VS 中打开

debug	2020/9/24 10:26	文件夹	
release	2020/9/24 0:35	文件夹	
.qmake.stash	2020/9/24 0:35	STASH 文件	1 KB
CGTemplate	2019/10/11 12:32	Qt Project file	1 KB
CGTemplate.sln	2020/9/24 11:42	Visual Studio Sol...	2 KB
CGTemplate.vcxproj	2020/9/24 10:21	VC++ Project	15 KB
CGTemplate.vcxproj.filters	2020/9/24 0:35	VC++ Project Fil...	3 KB
CGTemplate.vcxproj.user	2020/9/24 10:09	Per-User Project...	1 KB
main	2019/10/11 16:15	C++ Source File	1 KB
myglwidget	2020/9/24 10:25	C++ Source File	2 KB
myglwidget	2020/9/23 18:58	C Header File	1 KB

2、 开始运行 main.cpp 文件，可能会出现头文件缺失的问题，如下图所示：



```

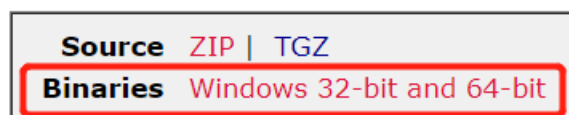
1  #ifndef MYGLWIDGET_H
2  #define MYGLWIDGET_H
3
4  #ifdef MAC_OS
5  #include <QtOpenGL/QtOpenGL>
6  #else
7  #include <GL/glew.h>
8  #endif
9  #include <QtGui>
10 #include <QOpenGLWidget>
11 #include <QOpenGLFunctions>
12
13 class MyGLWidget : public QOpenGLWidget{
14     Q_OBJECT
15

```

3、 配置 GL/glew.h

a) 下载 GLEW (Binaries Windows 32-bit and 64 bit)

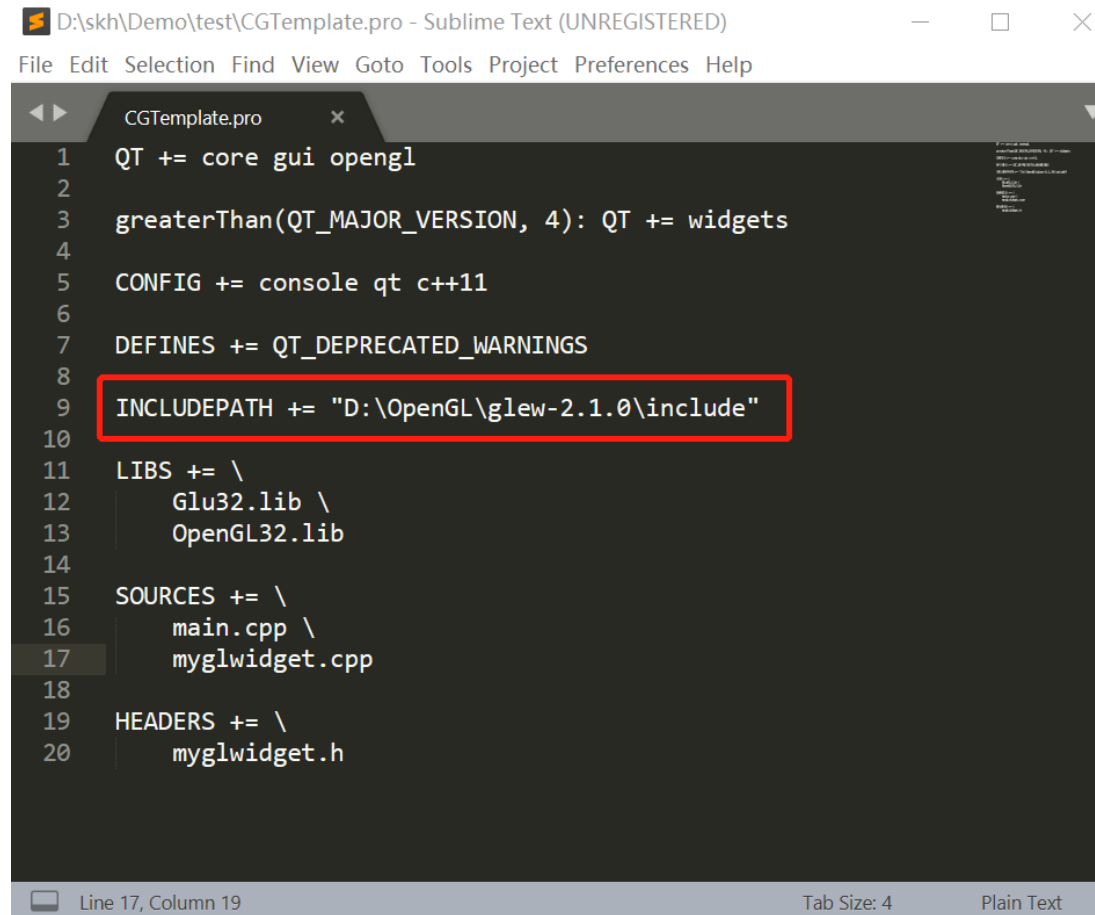
网址: <http://glew.sourceforge.net/>



下载后解压 (记住路径, 下面要用到)

b) 配置 OpenGL

- i. 打开相应的.pro 文件
- ii. 加入以下代码：（记得保存）



```
D:\skh\Demo\test\CGTemplate.pro - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

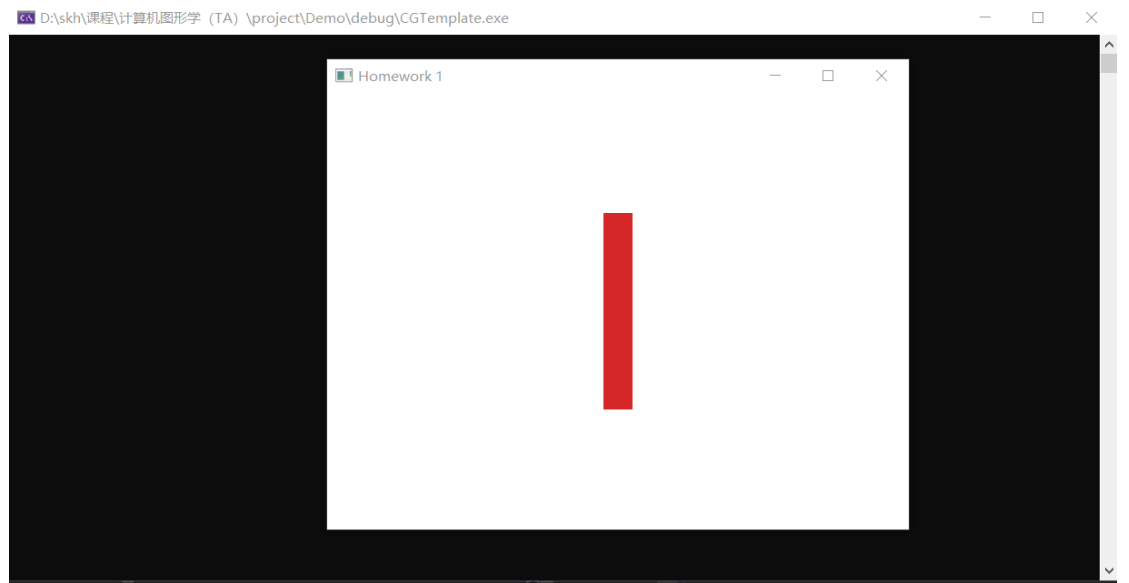
CGTemplate.pro
1  QT += core gui opengl
2
3  greaterThan(QT_MAJOR_VERSION, 4): QT += widgets
4
5  CONFIG += console qt c++11
6
7  DEFINES += QT_DEPRECATED_WARNINGS
8
9  INCLUDEPATH += "D:\OpenGL\glew-2.1.0\include"
10
11  LIBS += \
12      Glu32.lib \
13      OpenGL32.lib
14
15  SOURCES += \
16      main.cpp \
17      myglwidget.cpp
18
19  HEADERS += \
20      myglwidget.h

Line 17, Column 19    Tab Size: 4    Plain Text
```

切记：双引号内的路径要替换成在第 a) 步的解压  
路径

- iii. 重新 qmake 生成项目

#### 4、 重新运行 main.cpp 文件



六、 开始完成你的第一次作业吧