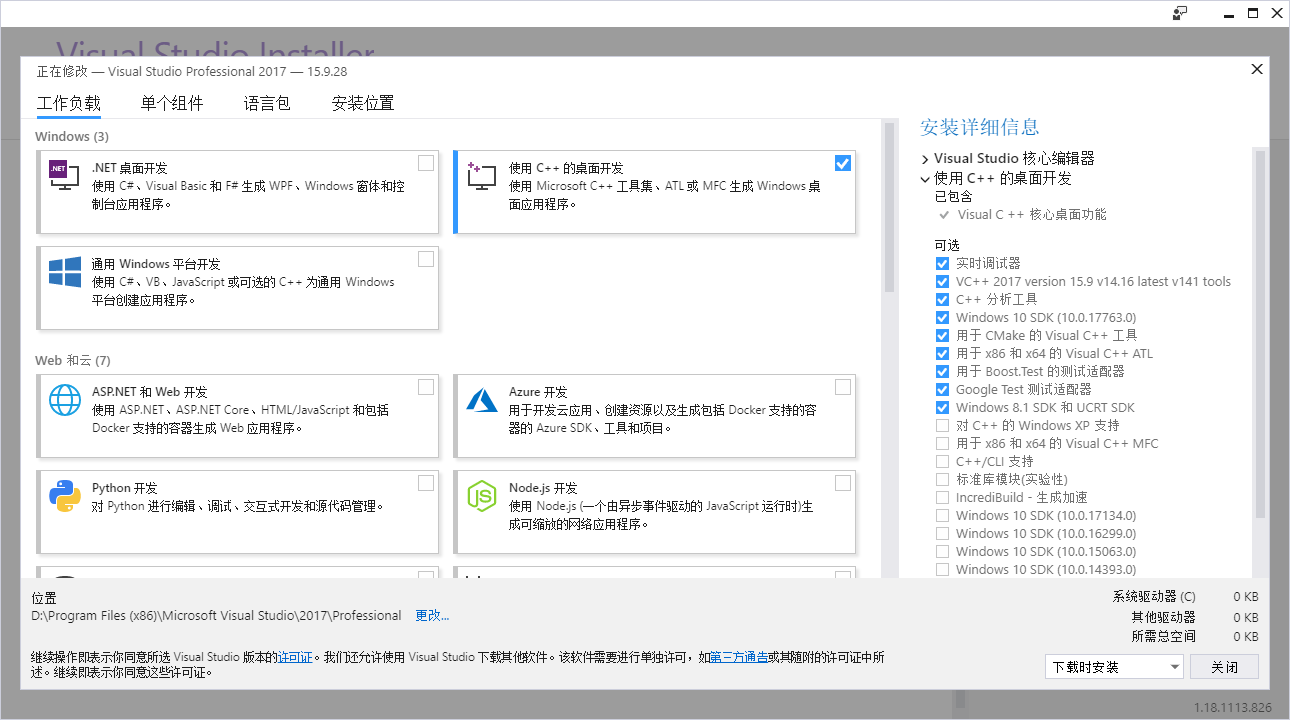
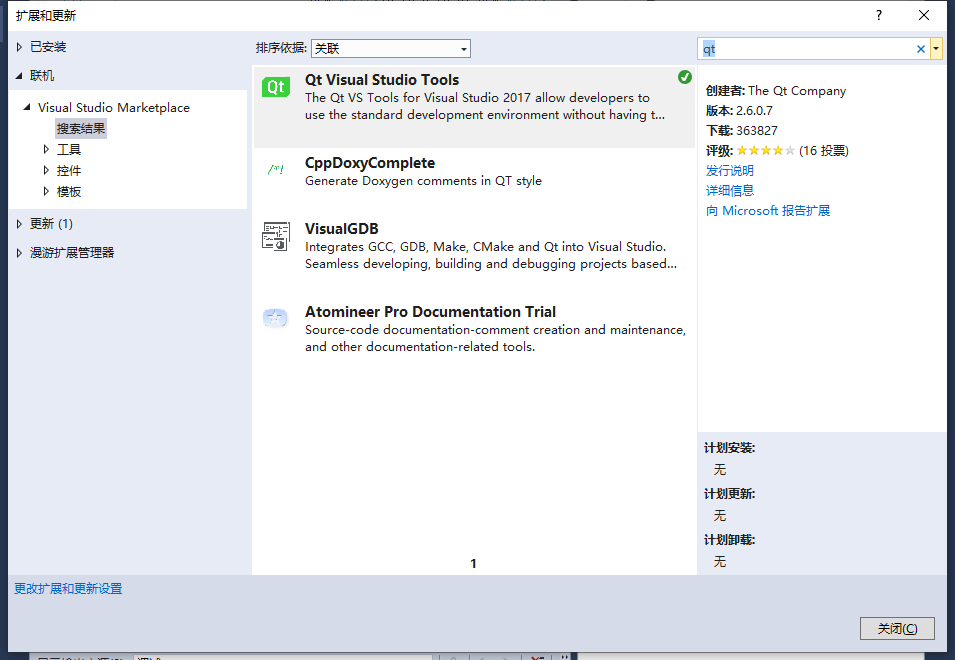
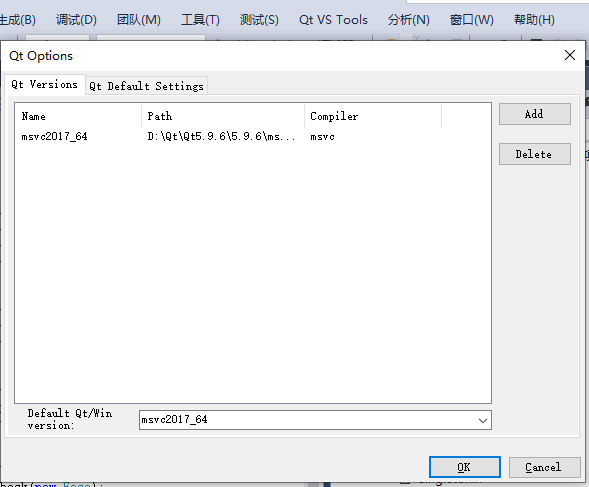
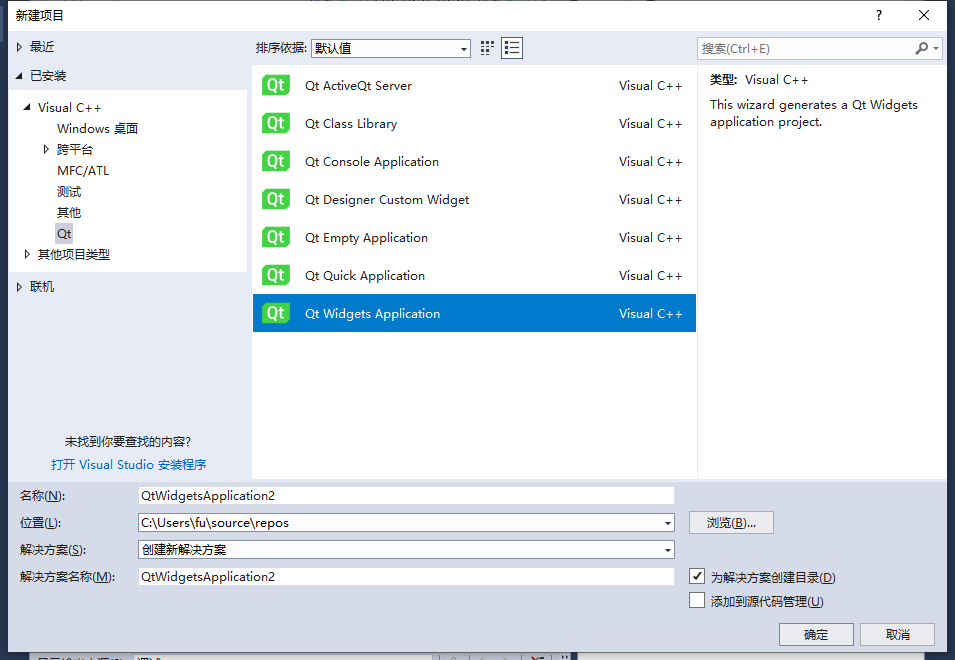
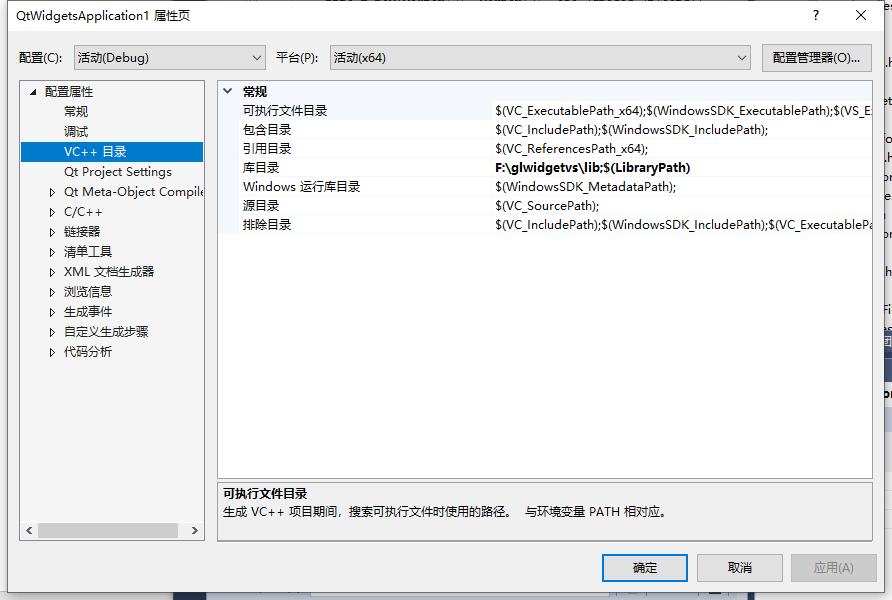
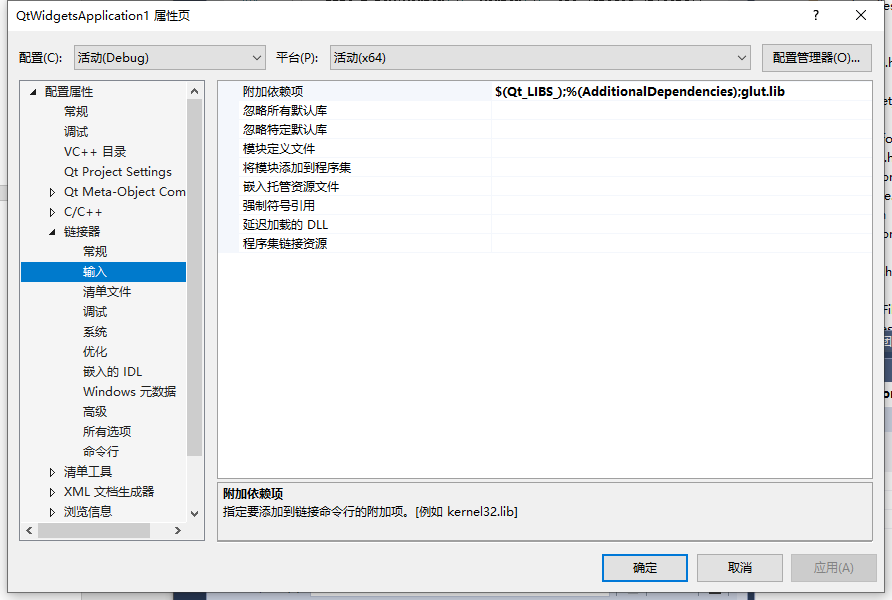
1. 环境配置，安装VS2017，如下图。



1. 安装Qt5.9.5，添加MSVC2017（64bit）组件，安装Qt\_creator。
2. 在VS中安装Qt Visual Studio Tools。
3. 程序安装完成后重启VS。在Qt VS Tools中添加Qt编译器。
4. 新建Qt widgets Application 项目，删除原有的杂项。
5. 添加压缩包中src文件夹下的代码文件，在项目属性页中添加库目录，地址为压缩包中lib文件夹。最后添加附加依赖项glut.lib。





7.可进行调试运行。

程序简介：

cad.h中包含程序所用到的数据结构。在main.cpp中包含程序所用的欧拉操作以及扫成扫做。其余为图形显示的代码，来自曾寅家同学。

1. Solid\* mvsf（）无传入参数，生成一新实体、新面、新环，返回实体对象的指针。
2. Halfedge\* mev（Vertex\* v1，Vertex\* v，loop\* lp），传入三个参数分别为一已有点、新点、与已有点所在的环，生成一条新边，返回环中的半边指针。
3. Face\* mef（Vertex\* v1，Vertex\* v2，loop\* lp1），给定同一外环中的两个点，构造一条新边，同时生成一个新环、新面,返还以v2为半边起始点的面指针。
4. Face\* kemr（Vertex \*v1,Vertex \*v2, Loop\* lp），消去环中的一条边，在同一个外环里，构造一个内环，此处输入要删除边的起始点和终止点，起始点在外环上，终止点在内环上，返回内环所在的面指针。
5. Face\* kfmrh（Face\* f1, Face\* f2），删除一个面f2，并将其定义为一个内环，进而在体中生成一个柄或将两物体合成一个物体，返回内环所在的面指针。
6. Face\* sweep（Face\* f, Vec\* vec, int L），给定要扫成的面、方向以及长度，进行扫成操作，返回扫成后的面。

程序中可直接通过欧拉操作进行实体构建（被注释），也可通过扫成算法进行实体构建，可通过循环进行多次扫成（被注释）。