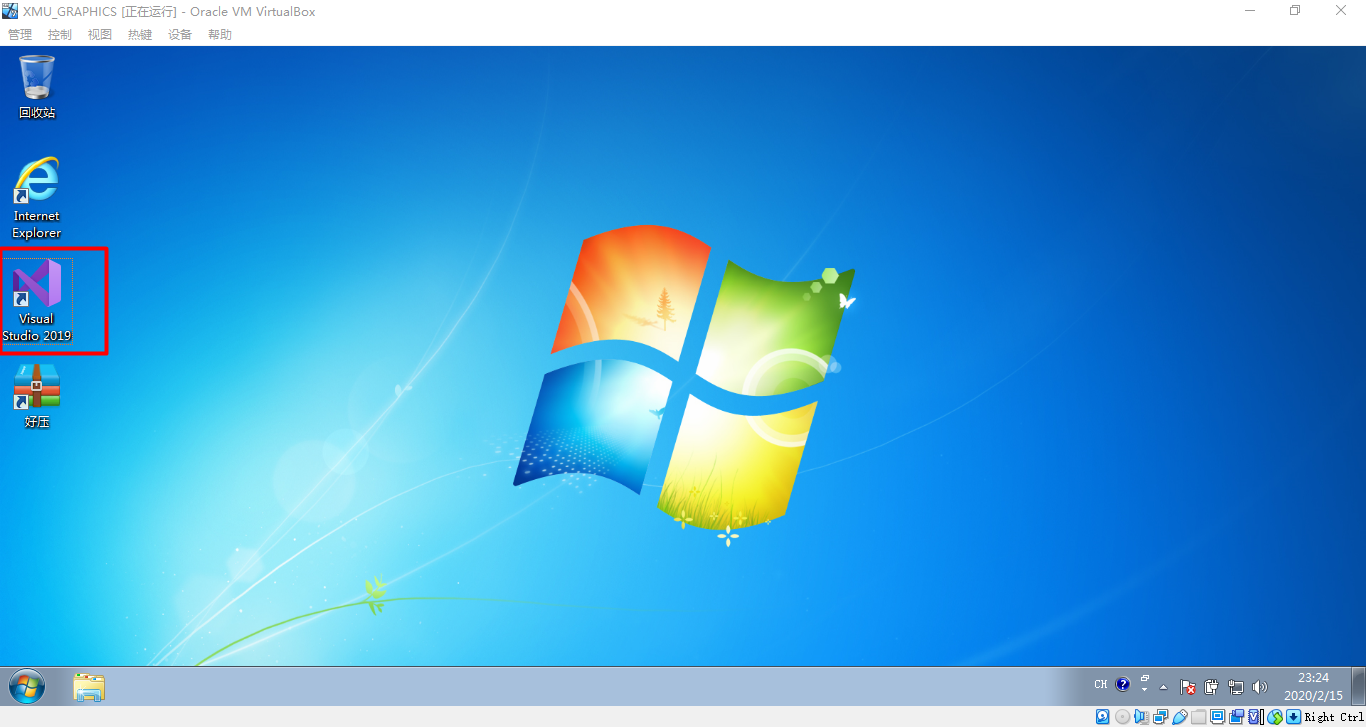
**实验0 环境配置**

**一、IDE的选择**

**我们已经提前在虚拟机中安装好Visual Studio 2019**，如果想要自行安装Visual Studio 2019可前往vs官方网站（https://docs.microsoft.com/en-us/visualstudio/releases/2019/release-notes）下载。



打开vs，创建新的项目，选中**空项目**创建即可。

电脑屏幕的手机截图

描述已自动生成

图形用户界面, 文本

描述已自动生成

**二、OpenGL安装**

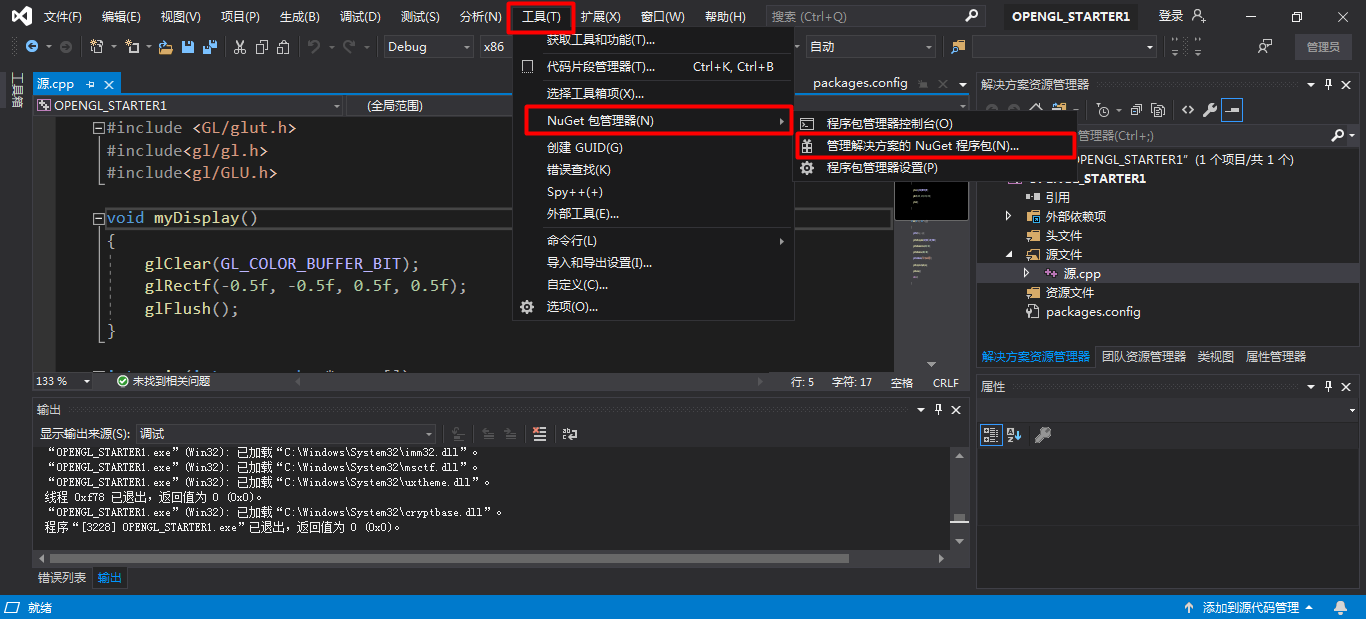
GLUT是什么：GLUT是OpenGL Utility Toolkit 缩写，用于实现在windows下进行opengl的窗口管理。freeglut是OpenGL Utility Toolkit（GLUT）库的免费软件/开源替代品。在后续的实验中，我们均使用freeglut。

想要自行配置OpenGL环境可按照以下两种方法

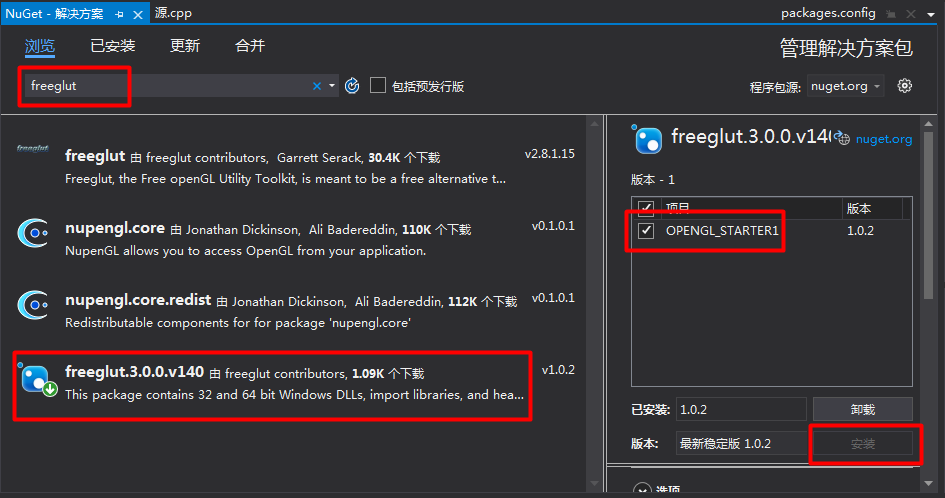
1. Nuget设置（推荐）

请注意，使用此方式配置时，每创建一个新项目都需要重新配置一次

在VS中选择工具 ->Nuget包管理器 ->解决方案的Nuget程序包



搜索freeglut->选择freeglut.3.0.0.v140 ->勾选OPENGL\_STARTER1（项目名）->点击安装



1. 手动安装

请注意，使用此方式配置时，创建一个新项目时无需重新配置。

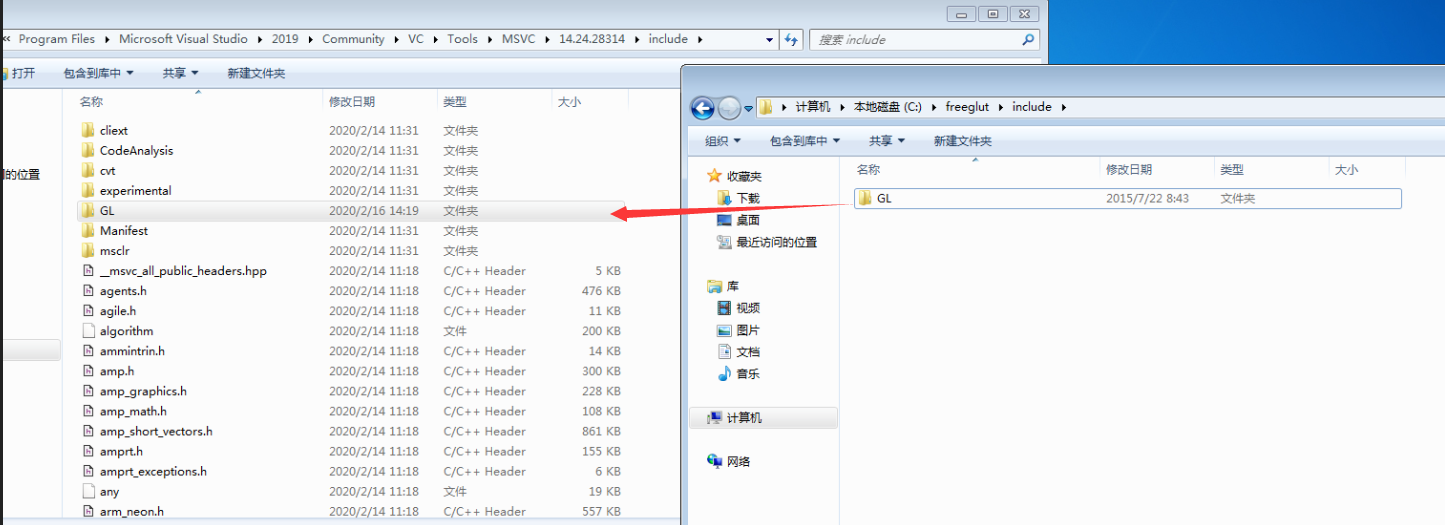
从网上/群文件/ftp下载freeglut包，并解压

下载地址为：<https://www.transmissionzero.co.uk/files/software/development/GLUT/freeglut-MSVC.zip>

或者在这里



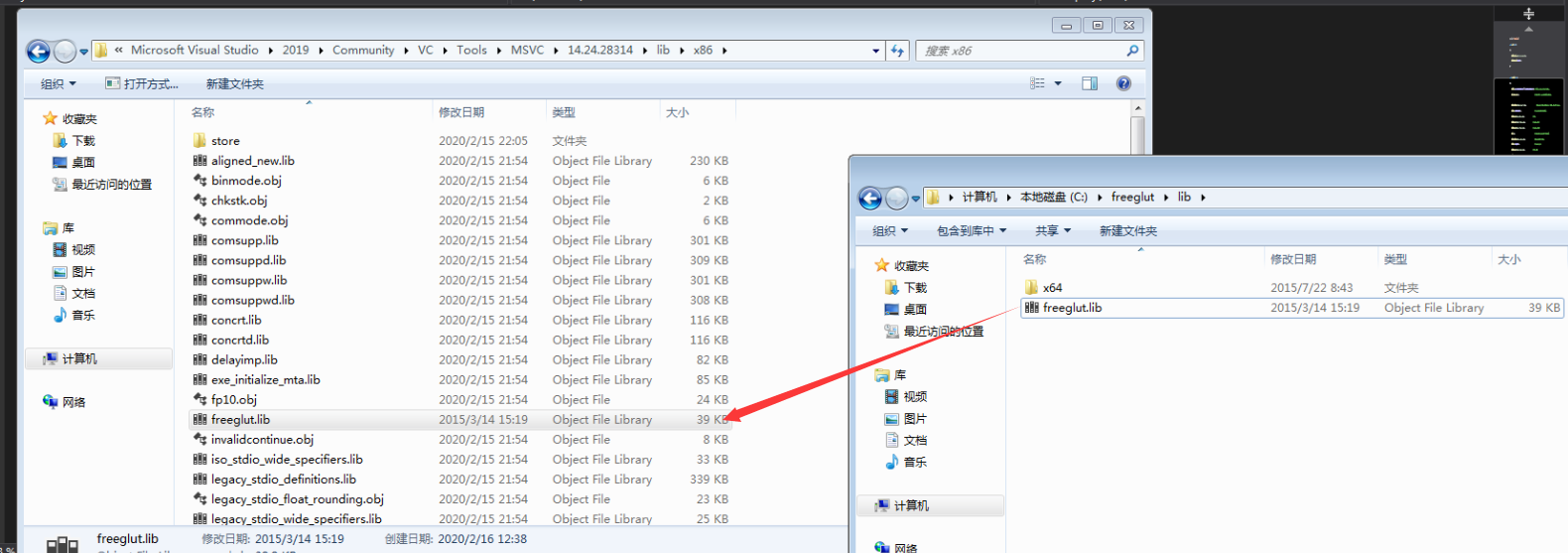
把freeglut的include文件夹下的GL文件夹复制到Windows SDK的Include目录下：(对于Visual Studio 2019该目录为 C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.24.28314\include)



说明：上述目录是Visual Studio查找头文件的默认目录。当然你也可以放在任意你感觉合适的位置，但是需要对Visual Studio进行设置，是它知道你存放的路径。

把freeglut库文件freeglut.lib复制到Visual Studio Windows SDK的Lib子目录

* 将lib/freeglut.lib 复制到C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.24.28314\lib\x86 下



说明：上述目录是Visual Studio查找库文件的默认目录。当然你也可以放在任意你感觉合适的位置，但是需要对Visual Studio进行设置，是它知道你存放的路径

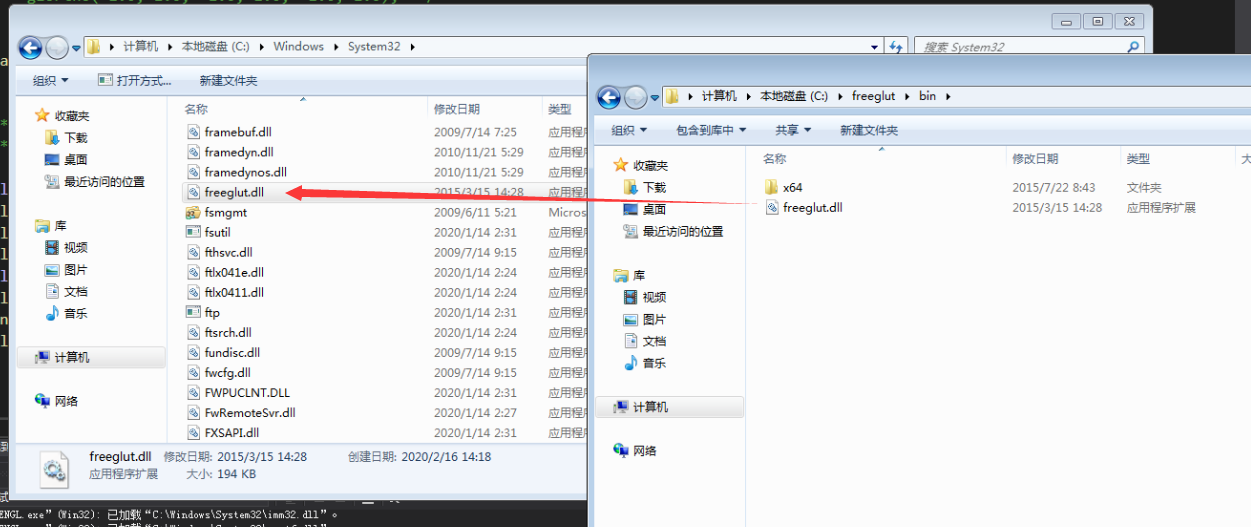
④ 把freeglut的dll文件freeglut.dll复制到系统目录：

将bin/freeglut.dll复制到系统动态链接库目录

* 若系统是64位，将bin/freeglut.dll 复制到“C:\Windows\systemWOW64”

将bin/x64/freeglut.dll 复制到” C:\Windows\system32”

* 若系统是32位，将bin/freeglut.dll 复制到" C:\Windows\System32"



**说明：以上请搞清楚.h, .lib, .dll头文件的作用，如果搞不清楚，后面的课程可能会有难，请与周围同学请教。**

⑤ 配置完成，启动VS，无需其他额外配置，即可使用。

**三、运行第一个OpenGL程序**

**在启动页的左侧可以看到我们已经编写好的一个名为OPENGL\_STARTER1的OpenGL程序**，这个项目是使用第1种方式配置的。打开该项目并运行，测试OpenGL环境是否正确安装。

或直接使用以下代码：

#ifdef \_\_APPLE\_\_

#include <GLUT/glut.h>

#else

#include <GL/glut.h> /\* glut.h includes gl.h and glu.h\*/

#endif

void display(void)

{

/\* clear window \*/

glClear(GL\_COLOR\_BUFFER\_BIT);

/\* draw unit square polygon \*/

glBegin(GL\_POLYGON);

glVertex2f(-0.5, -0.5);

glVertex2f(-0.5, 0.5);

glVertex2f(0.5, 0.5);

glVertex2f(0.5, -0.5);

glEnd();

/\* flush GL buffers \*/

glFlush();

}

void init()

{

/\* set clear color to black \*/

/\* glClearColor (0.0, 0.0, 0.0, 0.0); \*/

/\* set fill color to white \*/

/\* glColor3f(1.0, 1.0, 1.0); \*/

/\* set up standard orthogonal view with clipping \*/

/\* box as cube of side 2 centered at origin \*/

/\* This is default view and these statement could be removed \*/

/\* glMatrixMode (GL\_PROJECTION);

glLoadIdentity ();

glOrtho(-1.0, 1.0, -1.0, 1.0, -1.0, 1.0); \*/

}

int main(int argc, char\*\* argv)

{

/\* Initialize mode and open a window in upper left corner of screen \*/

/\* Window title is name of program (arg[0]) \*/

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(500, 500);

glutInitWindowPosition(0, 0);

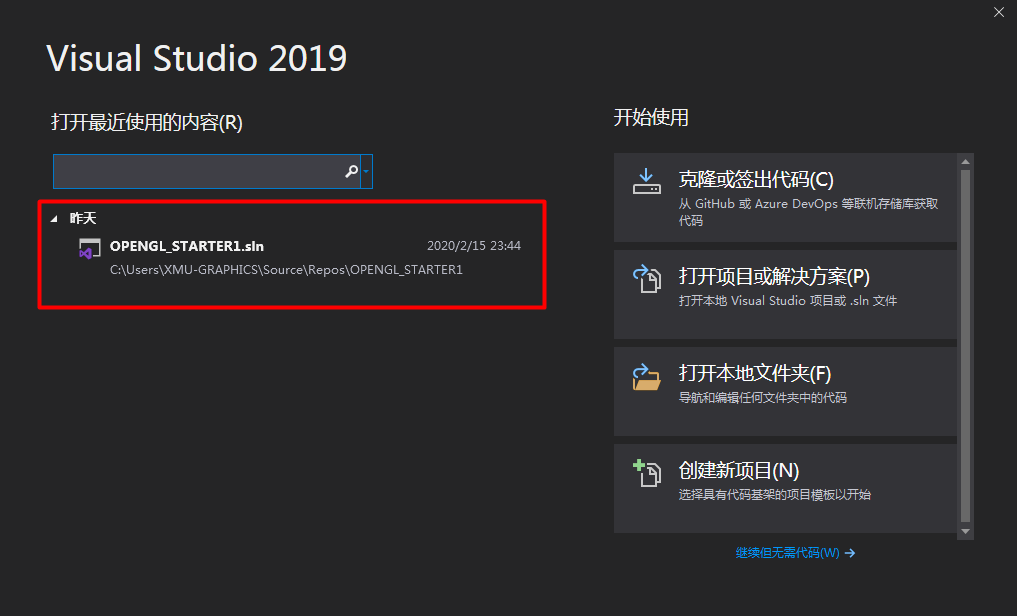
glutCreateWindow("simple");

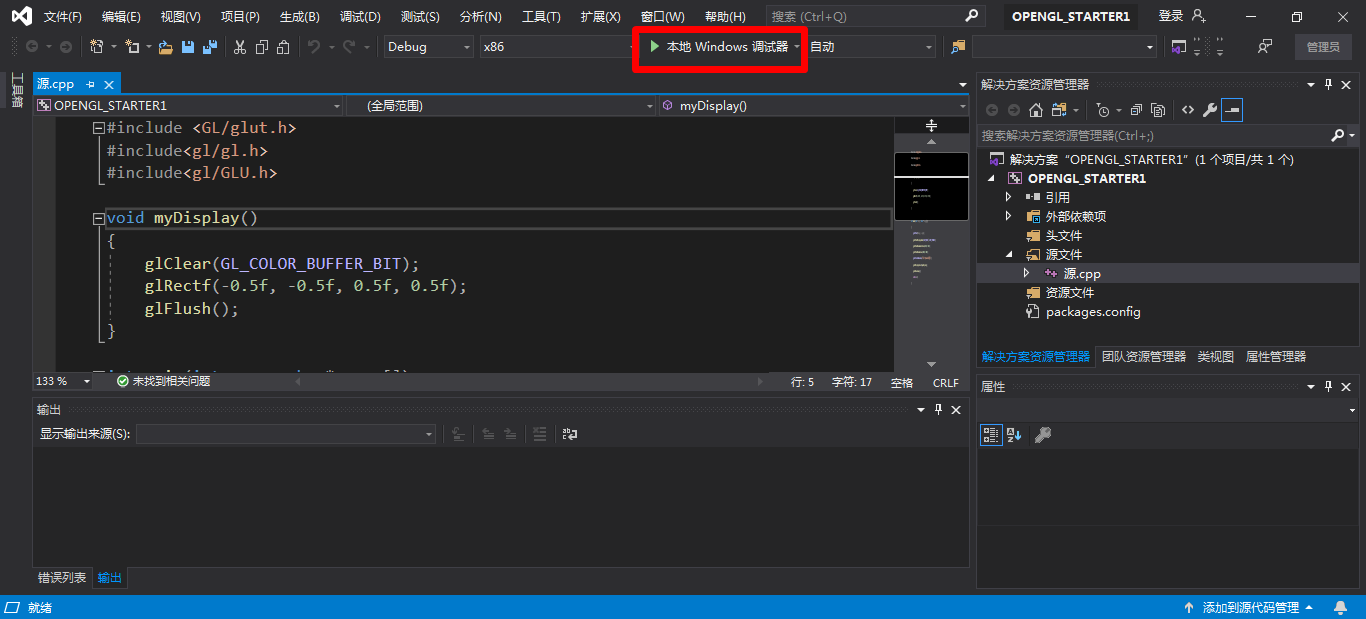
glutDisplayFunc(display);

init();

glutMainLoop();

}





运行结果如下

