**实验1、OpenGL初步**

建议阅读资料：

（1）OpenGL编程基础 （第三版）第1、2章

要求：

（1）了解OpenGL [OpenGL是什么? OpenGL编程涉及哪些库，这些库的作用是什么？

（2）学会OpenGL在Windows环境下的配置（如有需要，Mac和Linux环境请上网搜索）

（3）编写、运行第一个OpenGL程序

（4）尝试理解、运行红宝书或者教材中其他OpenGL程序

作业提交说明：

本次实验共有３个任务。Task2、3需要提交。

请根据实验结果，提交word版本实验报告，实验报告写明学号姓名，陈述、展示实验结果。

提交方式为：*将代码源文件、可执行文件、实验报告*放到一个文件夹中，文件夹命名格式为：姓名\_学号 ，上传到ftp服务器中相应目录下（/上传作业/曾鸣/2022软工（或数媒）图形学/exp1）。*请确保提交的可执行文件可以运行（*打分的重要依据）。注意代码提交源文件即可（.h, .cpp文件），不*需要提交整个工程项目*。每次实验作业的提交截止日期为下一次实验课前一天晚上。

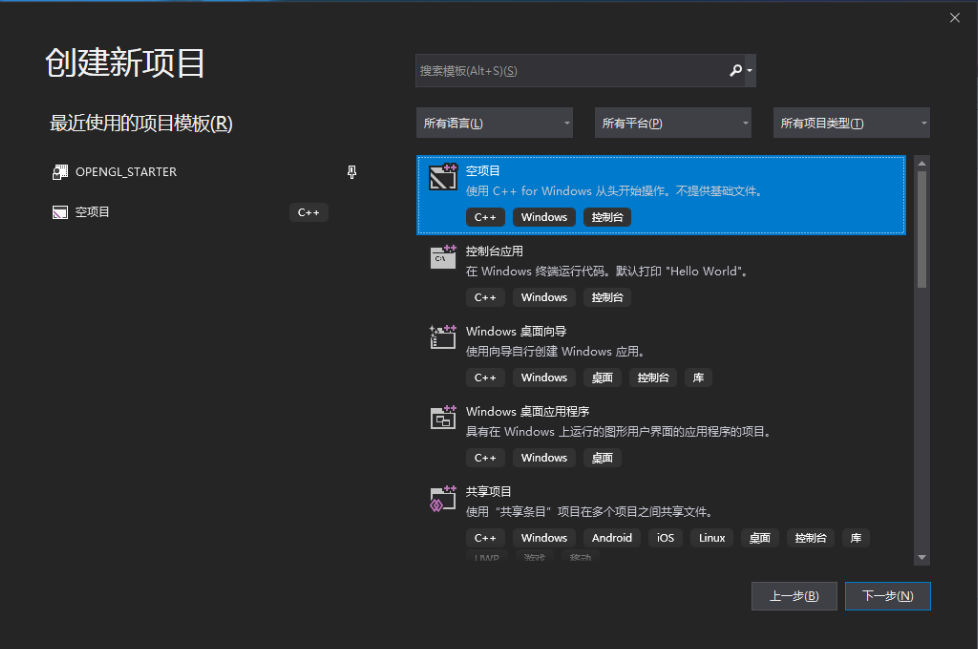
**OpenGL的配置**（Win7+Visual Studio 2019+freeglut）**[请务必熟悉此流程，今后每次实验都会用到]**

第0步: 环境配置

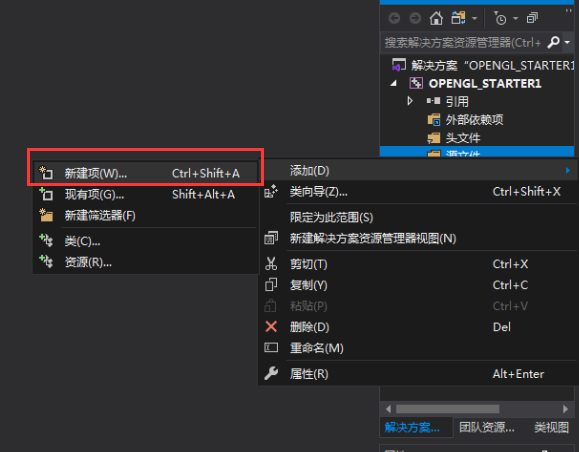
按照exp\_0配置Visual studio 2019 及 freeglut

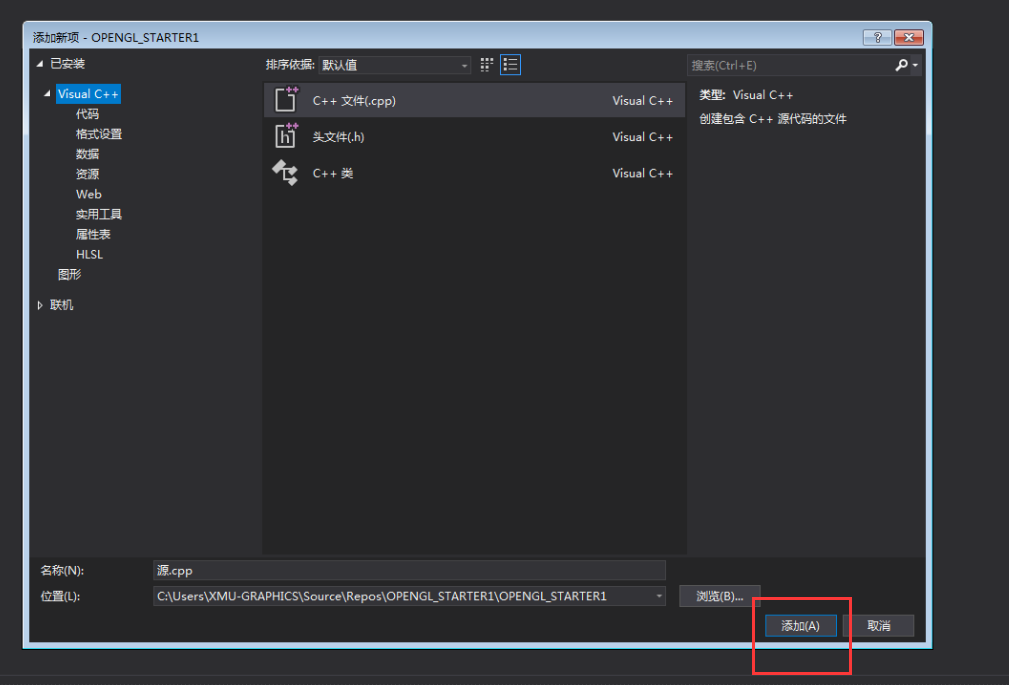
第1步 创建一个空的工程

1.1 新建一个空的工程，指定目录位置(Location)和项目名称(Name)，点击确认(OK)



第2步 新建或添加C/C++文件

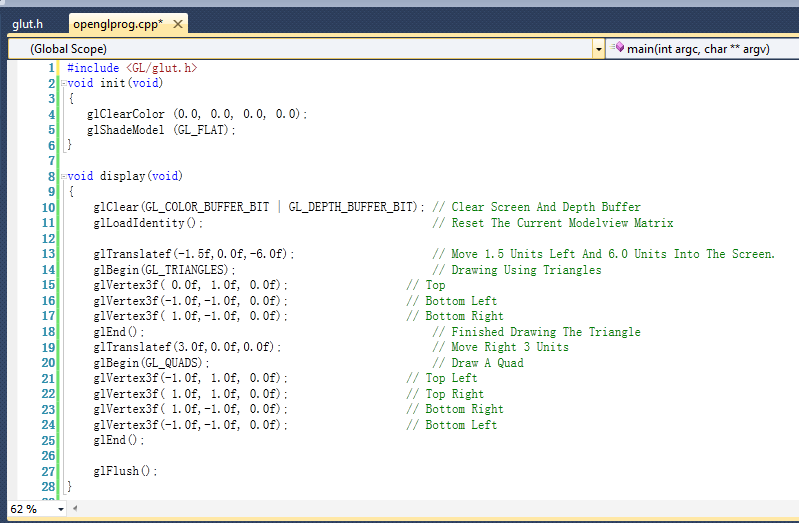




第4步 编写OpenGL代码，其中需要包括头文件

#include <gl/glut.h>

编写你的代码



第5步 生成(Bulid)并执行程序

**Task1. 第一个OpenGL程序：**

**在上述任务的第4步中，填入如下代码，编译执行：**

#include<GL/glut.h>

voidinit(void)

{

glClearColor(0.0,0.0,0.0,0.0);

glShadeModel(GL\_FLAT); //使用单一颜色

}

voiddisplay(void)

{

glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT);// Clear Screen And Depth Buffer

glLoadIdentity();// Reset The Current Modelview Matrix

glTranslatef(-1.5f,0.0f,-6.0f);// Move 1.5 Units Left And 6.0 Units Into The Screen.

glBegin(GL\_TRIANGLES);// Drawing Using Triangles

glVertex3f(0.0f,1.0f,0.0f);// Top

glVertex3f(-1.0f,-1.0f,0.0f);// Bottom Left

glVertex3f(1.0f,-1.0f,0.0f);// Bottom Right

glEnd();// Finished Drawing The Triangle

glTranslatef(3.0f,0.0f,0.0f);// Move Right 3 Units

glBegin(GL\_QUADS);// Draw A Quad

glVertex3f(-1.0f,1.0f,0.0f);// Top Left

glVertex3f(1.0f,1.0f,0.0f);// Top Right

glVertex3f(1.0f,-1.0f,0.0f);// Bottom Right

glVertex3f(-1.0f,-1.0f,0.0f);// Bottom Left

glEnd();

glFlush();

}

voidreshape(intw,inth)

{

glViewport(0,0,(GLsizei)w,(GLsizei)h);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

glFrustum(-1.0,1.0,-1.0,1.0,1.5,20.0);

glMatrixMode(GL\_MODELVIEW);

}

voidkeyboard(unsignedcharkey,intx,inty)

{

switch(key){

case27:

exit(0);

break;

}

}

intmain(intargc,char\*\*argv)

{

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowSize(500,500);

glutInitWindowPosition(100,100); //窗口处于屏幕左上角的位置

glutCreateWindow(argv[0]);

init();

glutDisplayFunc(display);

glutReshapeFunc(reshape);

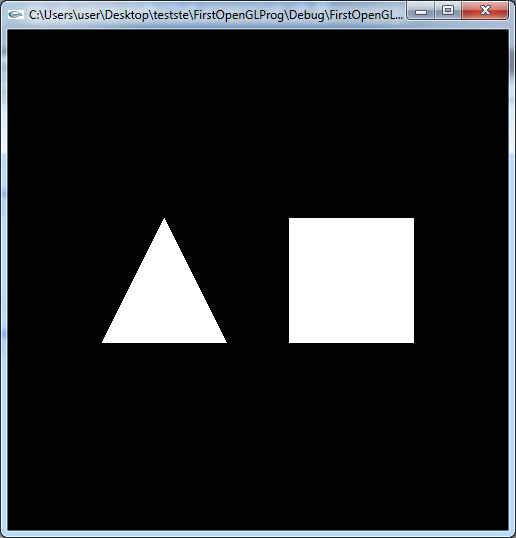
glutKeyboardFunc(keyboard);

glutMainLoop();

return0;

}

运行结果为：



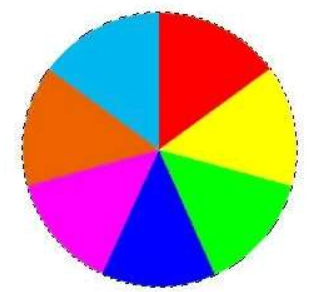
学习红宝书并思考：

（1）回调函数是什么意思？GLUT中有哪些回调函数？

（2）用鼠标改变窗口大小会发生什么？哪个函数在影响整个过程？

（3）试着画画其他图形？

**Task2. 绘制一个实心的圆。要求圆分成若干个扇形，每个扇形颜色不同。例如下图：**



**Task3. 绘制一个奥运五环，要求形状及颜色保持和下图一致，且要求颜色重叠部分完全和下图相同（即需要虑颜色的遮挡）。**

