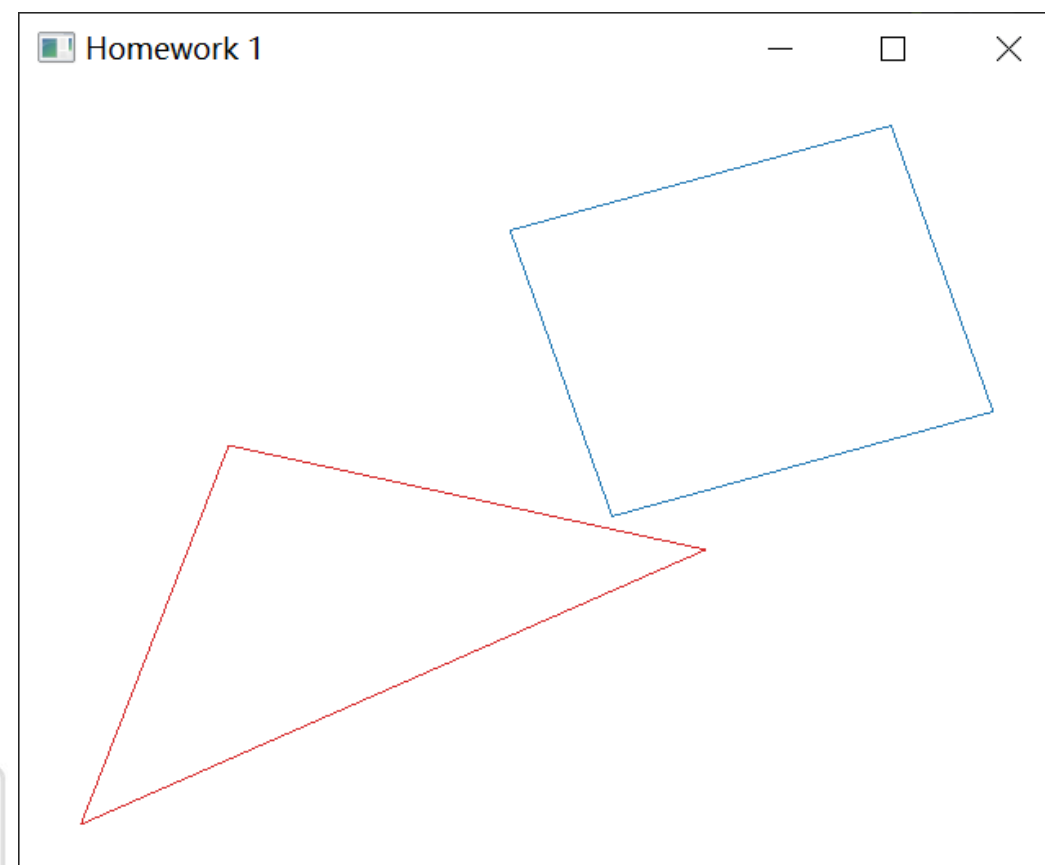


复现图中场景

- 要求：不使用GL_LINES或GL_LINE_STRIP, glTranslate, glRotate, glMultMatrix
- 实现平移旋转矩阵的构造
- 使用矩阵乘法对点进行变换
 - 可选做：使用四元数实现旋转
- 实现二维画线算法
 - 将变换后的点投影至二维
 - 于屏幕空间中连接两个二维顶点
 - 可选做：抗锯齿
- 需保证在窗口大小改变时，绘制结果与模板一致



• 程序模板使用Qt编写

– Qt官网下载

- https://download.qt.io/official_releases/qt/5.13/5.13.0/

– Qt镜像列表（exe for windows）

- <https://download.qt.io/archive/qt/5.13/5.13.0/qt-opensource-windows-x86-5.13.0.exe.mirrorlist>

– 模板包含四个文件

- CGTemplate.pro: Qt项目文件
 - 可用Qt Creator打开，或通过qmake生成visual studio/xcode项目/Makefile
 - Windows下生成vs项目文件：在命令行中使用qmake -tp vc
- main.cpp: 主函数，不需要修改
- myglwidget.cpp: 现有场景见scene_0(), 作业需实现其中scene_1()函数
- myglwidget.h: 对应cpp的头文件

• Qt .pro文件

– 参考<https://doc.qt.io/qt-5/qmake-variable-reference.html>

```
QT += core gui opengl  
greaterThan(QT_MAJOR_VERSION, 4): QT += widgets
```

```
CONFIG += console qt c++11
```

```
DEFINES += QT_DEPRECATED_WARNINGS
```

```
LIBS += \  
    Glu32.lib \  
    OpenGL32.lib
```

```
SOURCES += \  
    main.cpp \  
    myglwidget.cpp
```

```
HEADERS += \  
    myglwidget.h
```

main.cpp

```
#include "myglwidget.h"
#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MyGLWidget w;
    w.setWindowTitle("Homework 1");
    w.show();
    return a.exec();
}
```

myglwidget.h

```
#ifdef MAC_OS
#include <QtOpenGL/QtOpenGL>
#else
#include <GL/glew.h>
#endif
class MyGLWidget : public QOpenGLWidget{
    Q_OBJECT

public:
    MyGLWidget(QWidget *parent = nullptr);
    ~MyGLWidget();

protected:
    void initializeGL();
    void paintGL();
    void resizeGL(int width, int height);
    void keyPressEvent(QKeyEvent *e);

private:
    int scene_id;
    void scene_0();
    void scene_1();
};
```

• scene_0()函数：当前场景的OpenGL实现

```
glClear(GL_COLOR_BUFFER_BIT);  
glMatrixMode(GL_PROJECTION);  
glLoadIdentity();  
glOrtho(0.0f, 100.0f, 0.0f, 100.0f, -1000.0f, 1000.0f);
```

```
glMatrixMode(GL_MODELVIEW);  
glLoadIdentity();  
glTranslatef(50.0f, 50.0f, 0.0f);
```

//draw a triangle

```
glPushMatrix();  
glColor3f(0.839f, 0.153f, 0.157f);  
glTranslatef(-20.0f, -10.0f, 0.0f);  
glRotatef(45.0f, 1.0f, 0.0f, 1.0f);  
glTranslatef(-50.0f, -30.0f, 0.0f);  
glBegin(GL_LINE_LOOP);  
glVertex2f(10.0f, 10.0f);  
glVertex2f(50.0f, 50.0f);  
glVertex2f(80.0f, 10.0f);  
glEnd();  
glPopMatrix();
```

//draw a quad

```
glPushMatrix();  
glColor3f(0.122f, 0.467f, 0.706f);  
glTranslatef(20.0f, 20.0f, 0.0f);  
glRotatef(30.0f, 1.0f, 1.0f, 1.0f);  
glBegin(GL_LINE_LOOP);  
glVertex2f(-20.0f, -20.0f);  
glVertex2f(20.0f, -20.0f);  
glVertex2f(20.0f, 20.0f);  
glVertex2f(-20.0f, 20.0f);  
glEnd();  
glPopMatrix();
```

- **scene_1()函数：需要实现的函数**
 - 自行添加其他函数保证代码模块化

```
void MyGLWidget::scene_1()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(0.0f, width(), 0.0f, height(), -1000.0f, 1000.0f);

    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();

    glBegin(GL_POINTS);
    //your implementation
    //glVertex2i()
    glEnd();
}
```

评分细则

- 完成度及正确性(50%) + 编程规范(20%) + 书面报告(30%)
- 允许讨论代码，但作业代码和报告的编写和提交应独立完成
- 严禁任何形式的抄袭，否则将无法通过本课程

提交作业

- 邮箱: sysucg2019@163.com
- 时间: 2019.11.3 23:59

缓交作业

- 迟交作业不得分，但同学们可在作业截止前向TA:
liuj285@mail2.sysu.edu.cn提出使用3天的slip days(包括周末节假日，且不适用于期末现场报告)，并在slip days结束前补交作业