

基本图形技术

实验目的

- 掌握OpenGL实验环境配置方法
- 掌握直线与圆弧绘制方法
- 掌握平面多边形区域填充绘制方法

实验环境及工具

- Centos操作系统
- OpenGL库、g++编译工具
- MobaXterm远程控制工具

实验简介

OpenGL是一组绘图命令的API集合，利用这些API能够方便地描述二维和三维几何物体，并绘制到显示缓冲区。OpenGL具有如下特点：跨平台特性、应用广泛、网格透明性、高性能高质量、出色的编程特性等。

基本图形技术实验包括以下内容：

基本实现：

直线与圆弧的绘制、平面多边形区域填充（二选一）

直线与圆弧绘制：使用DDA算法/Bresenham算法/OpenGL绘制直线图元函数，在指定位置绘制直线。使用Bresenham算法/中点画圆法，在指定位置绘制圆弧。

平面多边形区域填充：使用OpenGL图元绘制函数/区域填充算法，绘制带区域填充效果的基本图元，包括三角形、矩形、圆等。

使用上述绘制的直线与圆弧/平面多边形区域填充，构建自己喜欢的图案，显示到屏幕空间。

进阶实现：

添加鼠标交互功能：

添加菜单：点击鼠标右键，弹出菜单，选择绘制的图形类别（直线、三角形、矩形等）、填充颜色。

鼠标交互：读取鼠标单击的坐标，使用鼠标拖拽生成图形，完成鼠标交互绘制。

实验流程与结果展示

文档介绍OpenGL的安装方式，展示图形编辑实现效果。

1. Centos安装OpenGL

(1) 安装链接器：

```
yum install mesa-libGL-devel mesa-libGLU-devel freeglut-devel
```

(2) 安装相关依赖包：

```
yum install mesa* freeglut*
```

(3) 安装C++编译器:

```
yum -y install gcc-c++
```

(4) 安装X11 (仅在远程连接机器时需要):

```
vim /etc/ssh/sshd_config
```

使用上述命令, 打开ssh配置文件, 找到下述内容, 修改成如下配置:

```
#AllowAgentForwarding yes
AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
X11DisplayOffset 10
X11UseLocalhost no yes
#PermitTTY yes
#PrintMotd yes no
#PrintLastLog yes
#TCPKeepAlive yes
```

安装相关依赖包:

```
yum install -y xorg-x11-xauth          #安装x11组件包
```

安装完成后, 重新启动命令行, 连接机器。

(5) 验证OpenGL是否安装成功:

使用vim编辑器, 创建cpp文件, 测试OpenGL

执行命令: vi test.cpp

粘贴下述测试代码:

```
#include <GL/gl.h>
#include <GL/glut.h>
#include <stdlib.h>

void init(void)
{
    glClearColor(0.0, 0.0, 0.0, 0.0);
    glMatrixMode(GL_PROJECTION);
    glOrtho(-5,5,-5,5,5,15);
    glMatrixMode(GL_MODELVIEW);
    gluLookAt(0,0,10,0,0,0,0,1,0);

    return;
}

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glutWireTeapot(3);
    glFlush();

    return;
}
```

```
int main(int argc, char* argv[])
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_RGB|GLUT_SINGLE);
    glutInitWindowPosition(0,0);
    glutInitWindowSize(300, 300);
    glutCreateWindow("OpenGL 3D View");
    init();
    glutDisplayFunc(display);

    glutMainLoop();
    return 0;
}
```

执行下述命令，编译测试文件，输出为test文件：

```
g++ test.cpp -lglut -lGLU -lGL -o test
```

执行下述命令，输出渲染出的茶壶结果：

```
./test
```

运行结果如下：



(6) 文件传输工具:使用Xftp工具传输文件，传输协议使用SFTP，ip地址、端口号与MobaXterm中填写相同。

2. 基本图形绘制

直线的绘制包括DDA算法，Bresenham算法等常用算法，OpenGL也可以直接绘制直线图元，使用GL_LINES函数。

圆弧的绘制包括Bresenham算法、中点画圆法等算法。

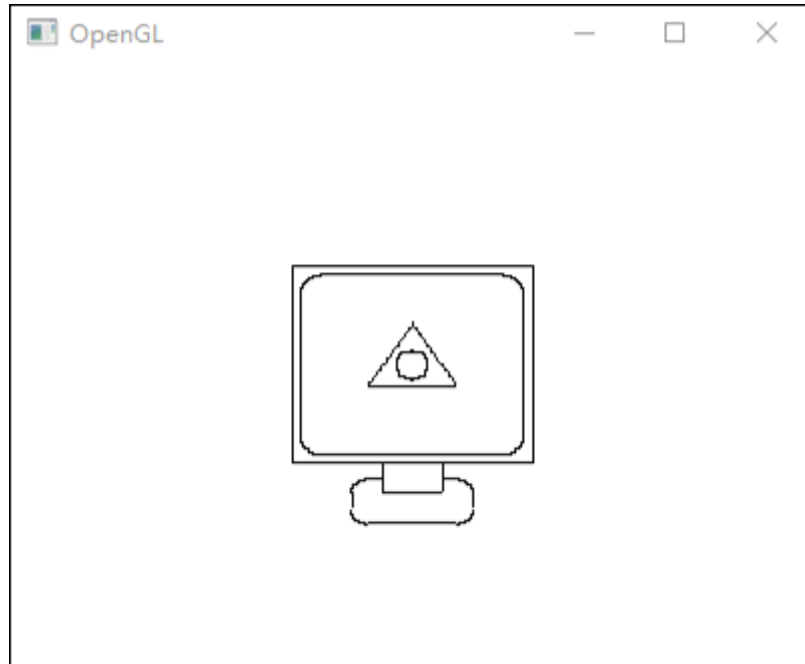
使用直线、圆弧，绘制任意图形，编写至main.cpp文件，使用下述命令编译可执行文件：

(不同的程序，单独放置于不同文件夹，使用命令 `mkdir xxx` 可以创建名为xxx的文件夹，使用命令 `cd xxx` 进入创建的文件夹，xxx为创建的文件夹名称，使用命令 `cd ../` 返回上级目录，**一个文件夹下，只能有一个main文件**)

```
g++ main.cpp -lglut -lGLU -lGL -o main
```

执行下述命令，输出图形结果：（结果仅为示例，可绘制任意图案）

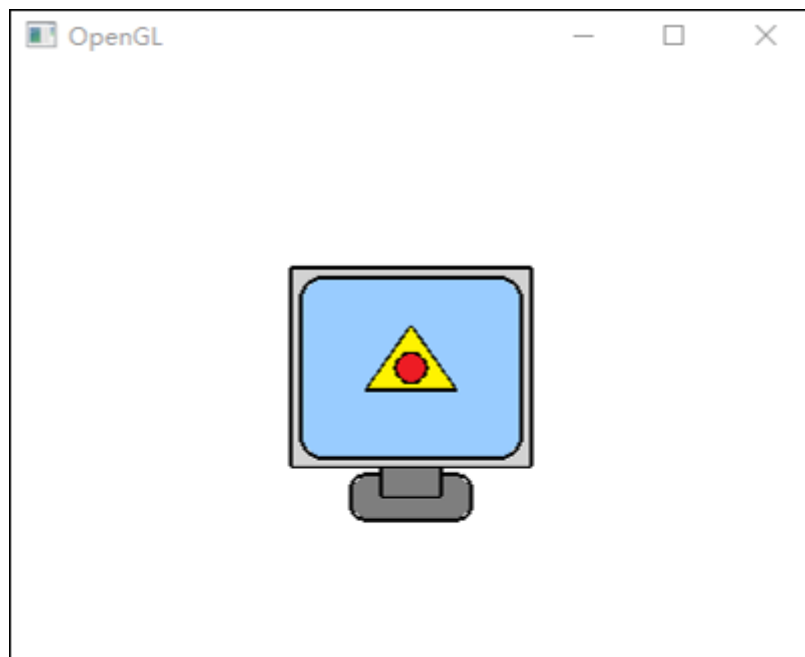
```
./main
```



3. 平面多边形区域填充

使用OpenGL图元绘制函数/区域填充算法，绘制带区域填充效果的基本图元，绘制任意图形。

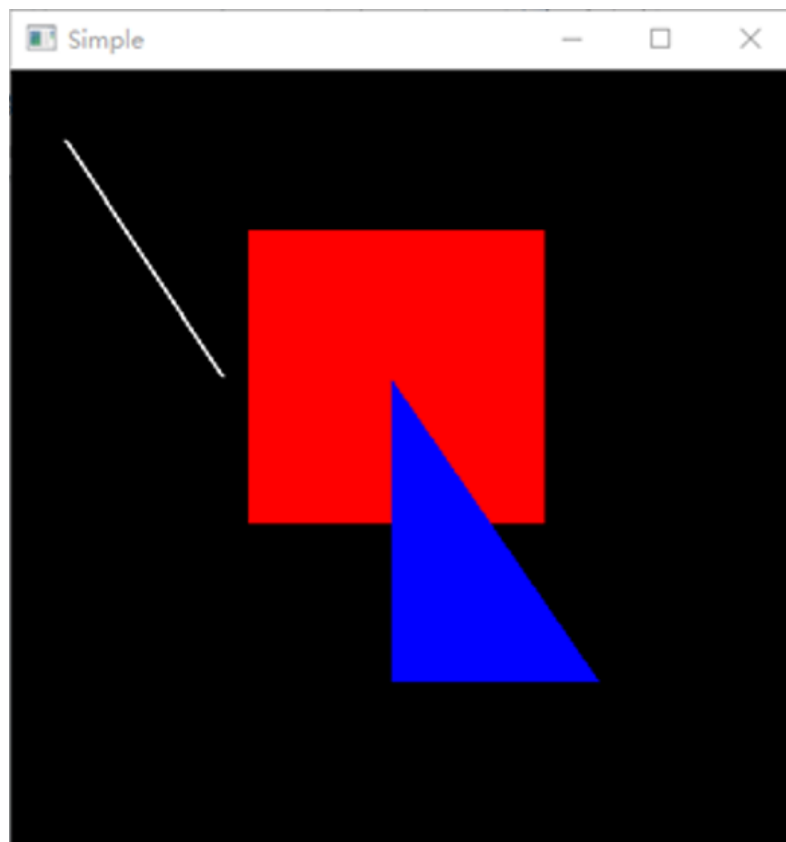
编写新的main.cpp文件，使用与上一步相同的编译方法，得到下述结果：（结果仅为示例）



4. 交互控制绘制

添加鼠标交互功能，使用鼠标右键打开菜单，选择绘制图案及绘制颜色。使用鼠标拖拽，绘制线段、三角形、矩形等图形。

编写新的main.cpp文件，使用与上一步相同的编译方法，得到下述结果：（图中结果为鼠标拖拽绘制）



提交说明：

请提交相关代码文件、执行脚本及说明文档，打包上传至OneLab系统提交，同时相关程序放置于服务器对应目录。

1.代码文件：

本次实验为第一次实验，请将代码实验代码整理于服务器/root/Exam1目录。如完成扩展实验（即实验代码和运行程序不止一份），可继续创建子目录。

2.执行脚本

.sh文件，包含程序编译及运行命令，可直接运行得到实验结果。执行脚本与实验代码位于相同文件夹。

3.说明文档：

说明文档需包含带**代码明细、环境配置、程序编译及运行命令、实验效果**等内容。

代码明细：对目录中代码进行简要说明。

环境配置：如果安装实验文档外其他依赖包，请在此说明。

程序编译及运行命令：说明程序的编译命令，运行命令。

实验效果：说明程序实现的效果，使用方式。

