实验十一 建模实验

时间：2022年5月18日

地点：信息学院机房2202

1、实验内容

使用opengl，书写教材P541页代码，进行颜色编码建模显示。

2、实验目的

调用函数完成颜色编码建模实验。

3、实验代码

#include <GL/glut.h>

#include <stdlib.h>

#include <windows.h>

GLsizei winWidth = 500,winHeight = 500;

GLfloat xComplexMin = -2.00,xComplexMax = 0.50;

GLfloat yComplexMin = -1.25,yComplexMax = 1.25;

GLfloat complexWidth = xComplexMax - xComplexMin;

GLfloat complexHeight = yComplexMax - yComplexMin;

class complexNum

{

public:

GLfloat x,y;

};

struct color

{

GLfloat r,g,b;

};

void init (void)

{

glClearColor(1.0,1.0,1.0,0.0);

}

void plotPoint(complexNum z)

{

glBegin (GL\_POINTS);

glVertex2f(z.x,z.y);

glEnd();

}

complexNum complexSquare (complexNum z)

{

complexNum zSquare;

zSquare.x = z.x\*z.x-z.y\*z.y;

zSquare.y = 2\*z.x\*z.y;

return zSquare;

}

GLint mandelSqTransf(complexNum z0,GLint maxIter)

{

complexNum z = z0;

GLint count = 0;

while((z.x\*z.x+z.y\*z.y<=4.0)&&(count<maxIter))

{

z = complexSquare(z);

z.x += z0.x;

z.y+=z0.y;

count++;

}

return count;

}

void mandelbrot(GLint nx,GLint ny,GLint maxIter)

{

complexNum z,zIncr;

color ptColor;

GLint iterCount;

zIncr.x = complexWidth/GLfloat(nx);

zIncr.y = complexHeight/GLfloat(ny);

for(z.x = xComplexMin; z.x<xComplexMax; z.x += zIncr.x)

{

for(z.y = yComplexMin; z.y<yComplexMax; z.y += zIncr.y)

{

iterCount = mandelSqTransf(z,maxIter);

if(iterCount>= maxIter)

ptColor.r = ptColor.g = ptColor.b = 0.0;

else if(iterCount>(maxIter/8))

{

ptColor.r = 1.0;

ptColor.g = 0.5;

ptColor.b = 0.0;

}

else if(iterCount>(maxIter/10))

{

ptColor.r = 1.0;

ptColor.g = ptColor.b = 0.0;

}

else if(iterCount>(maxIter/20))

{

ptColor.b = 0.5;

ptColor.r = ptColor.g = 0.0;

}

else if(iterCount>(maxIter/40))

{

ptColor.r = ptColor.g = 1.0;

ptColor.b = 0.0;

}

else if(iterCount>(maxIter/100))

{

ptColor.r = ptColor.b = 0.0;

ptColor.g=0.3;

}

else

{

ptColor.r = 0.0;

ptColor.g = ptColor.b = 1.0;

}

glColor3f(ptColor.r,ptColor.g,ptColor.b);

plotPoint(z);

}

}

}

void displayFcn(void)

{

GLint nx = 1000,ny = 1000,maxIter = 1000;

glClear(GL\_COLOR\_BUFFER\_BIT);

mandelbrot (nx,ny,maxIter);

glFlush();

}

void winReshapeFcn(GLint newWidth,GLint newHeight)

{

glViewport(0,0,newHeight,newHeight);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(xComplexMin,xComplexMax,yComplexMin,yComplexMax);

glClear(GL\_COLOR\_BUFFER\_BIT);

}

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

{

glutInit (&argc, argv); // 初始 GLUT.

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB); //设定显示模式

glutInitWindowPosition (50, 50); // 设定窗口位置

glutInitWindowSize (winWidth,winHeight); // 设定窗口大小

glutCreateWindow ("Mandelbrot Set");

// 用前面指定参数创建窗口，并定义窗口名称

init( ); // 进行一些初始化工作

glutDisplayFunc(displayFcn); // 指定绘制的回调函数

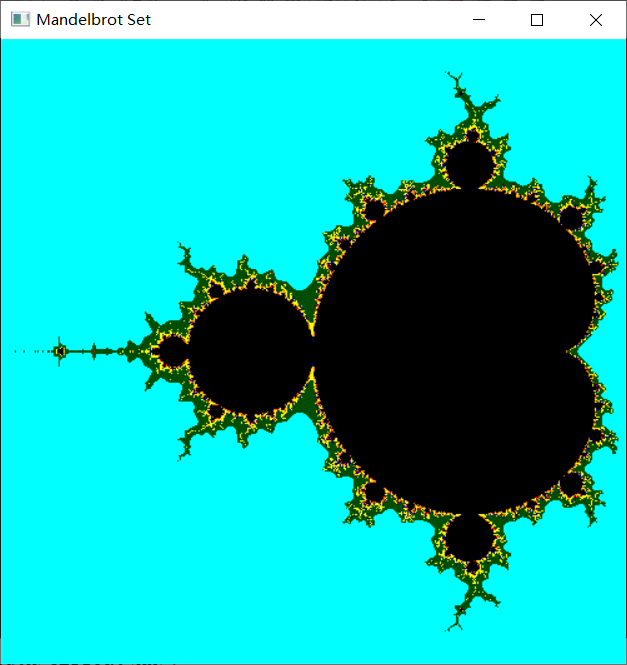
glutReshapeFunc(winReshapeFcn);

glutMainLoop ( );

return 0; // 进入无穷循环，等待事件处理

}

4、实验结果



5、实验总结

1.glClearColor：red、green、blue、alpha分别是红、绿、蓝、不透明度，值域均为[0,1]。即设置颜色，为后面的glClear做准备，默认值为（0,0,0,0）。切记：此函数仅仅设定颜色，并不执行清除工作。

2.gluOrtho2D：参数分别代表（左下角x坐标，右上角x坐标，左下角y坐标，右上角y坐标）—坐标相对于窗口左下角——原点）。