**郑州轻工业大学**

**《计算机图形学》**

**课程****实验报告**

实验名称：圆和椭圆生成算法

姓 名： 原彬贺

学 号： 542001020223

地 点： 实验楼302

指导老师： 黄 伟、陈启强

专业班级： 计算机科学与技术20-02班

实验成绩：

1. 实验目的
2. 实现绘制**圆的中点算法和****Bresenham算法**，并和鼠标进行交互。
3. 实现绘制**椭圆的中点算法**，并和鼠标进行交互。

二、实验内容

简述中心点算法，并编程实现利用中心点算法完成圆的绘制。

简述Bresenham算法，并编程实现Bresenham算法绘制圆。

利用中心点算法完成椭圆的绘制。

三、实验步骤及结果

**1.中心点算法画园：**

中点画园法：在进行圆的扫描转换时，首先应注意，只要能生成8分圆，那么圆的其他部分可以通过一系列的简单反射变换得到。那么已知圆心在原点的圆上的一点的(x,y)，那么根据对称性可以得到七个8分圆的对应的点(y,x),(y,-x),(x,-y),(-x,-y),(-y,-x),(-y,x),(-x,y).因此只需要构造八分圆。

核心代码：

void midpointcircle(int x0, int y0, int r)//中点画圆法

{

int x = 0;

int y = r;

int d = 1 - r; // 为了简化算法，摆脱浮点数，所以对d进行取整计算所以从1.25-r变到了1-r

glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_POLYGON);

setpoint(x0, y0, x, y);

while (x < y)

{

if (d < 0)

{

d += 2 \* x + 3;

}

else

{

d += 2 \* (x - y) + 5;

y--;

}

x++;

setpoint(x0, y0, x, y);

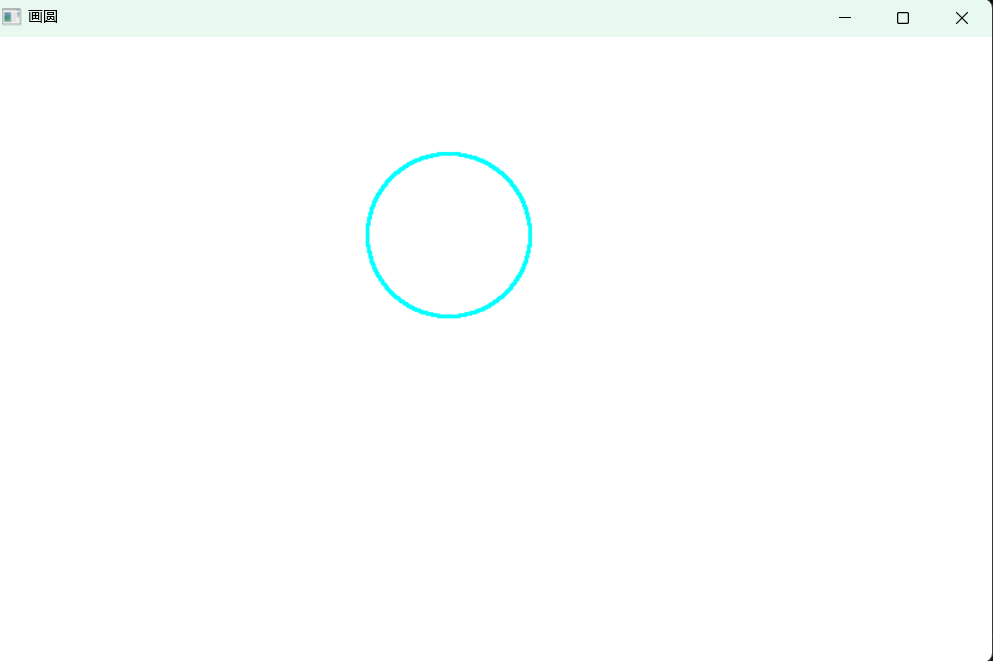
}

glEnd();

glFlush();

}

图片：



图片1 中心点算法画园

**2. Bresenham算法画园：**

Bresenham算法: 通过各行、各列像素中心构造一组虚拟网格线，按照直线起点到终点的顺序，计算直线与各垂直网格线的交点，然后根据误差项的符号确定该列象素中与此交点最近的象素。

核心代码：

void bresenhamcircle(int x0, int y0, int r)

{

int x, y;

y = r;

x = 0;

int d = 3 - 2 \* r;

while (x < y) {

Sleep(100);

CirclePoint(x0, y0, x, y);

if (d < 0) d += 4 \* x + 6;

else {

d += 4 \* (x - y) + 10;

y--;

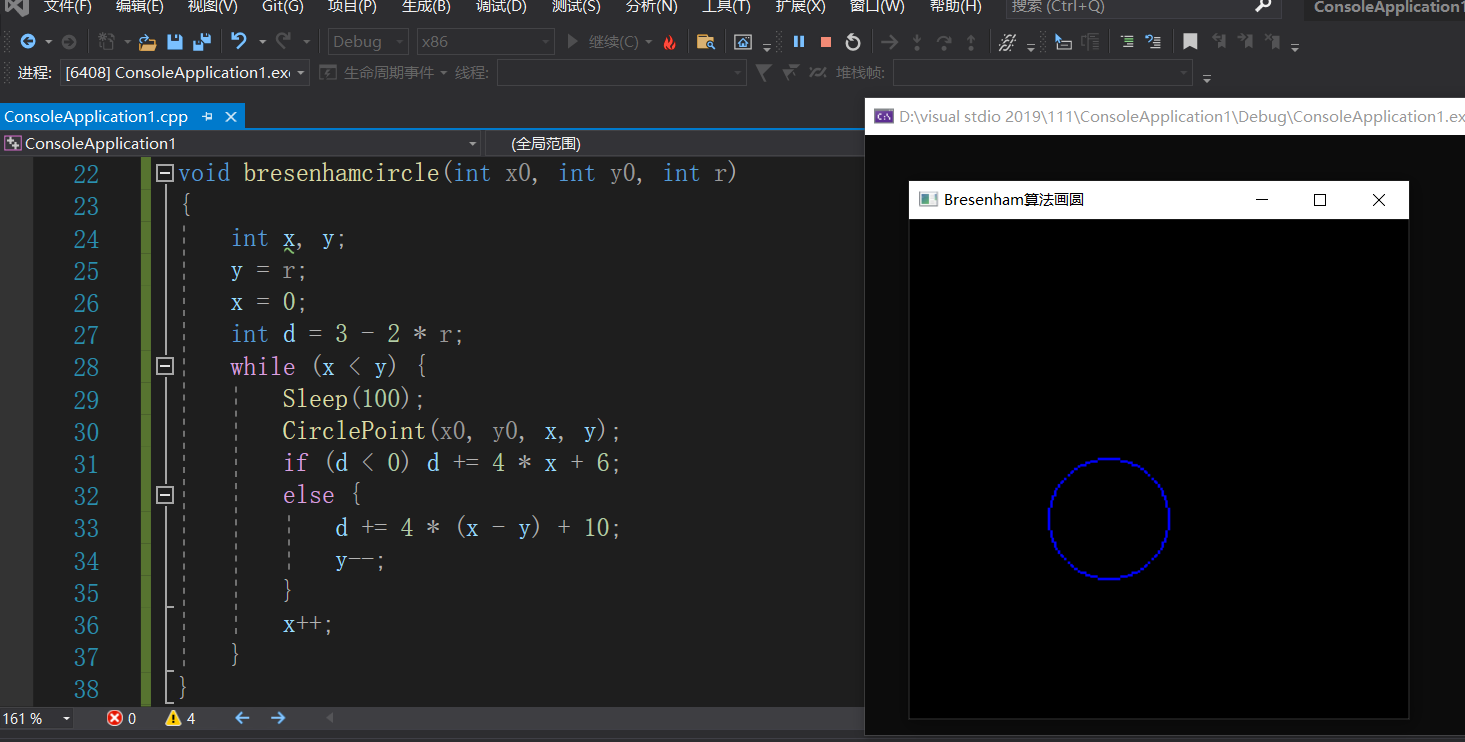
}

x++;

}

}

图片：



图片2 Bresenham算法画园

**3.中心点算法绘制椭圆**

核心代码：

/\*椭圆的中点算法\*/

void MidpointOval(int x0, int y0, int a, int b)

{

glClear(GL\_COLOR\_BUFFER\_BIT);//清除窗口显示内容

glBegin(GL\_POINTS);

int x = 0, y = b;

int a2 = a \* a, b2 = b \* b;

float d1 = b2 + a2 \* (0.25 - b);

float d2 = b2 \* (x + 0.5) \* (x + 0.5) + a2 \* (y - 1) \* (y - 1) - a2 \* b2;

OvalPoints(x, y, x0, y0);

/\*方法一\*/

while (a2 \* y > b2 \* x) { //注意！！！

glBegin(GL\_POINTS);

OvalPoints(x, y, x0, y0);

x++;

//++x;

if (d1 < 0) {

d1 += b2 \* (2 \* x + 3);

}

else {

y--;

//--y;

d1 += b2 \* (2 \* x + 3) + a2 \* (2 - 2 \* y);

}

}

x--; y++;

while (y > 0) {

y--;

if (d2 >= 0) { //取正下方的点

d2 += a2 \* (3 - 2 \* y);

}

else {

x++;

d2 += b2 \* (2 \* x + 2) + a2 \* (3 - 2 \* y);

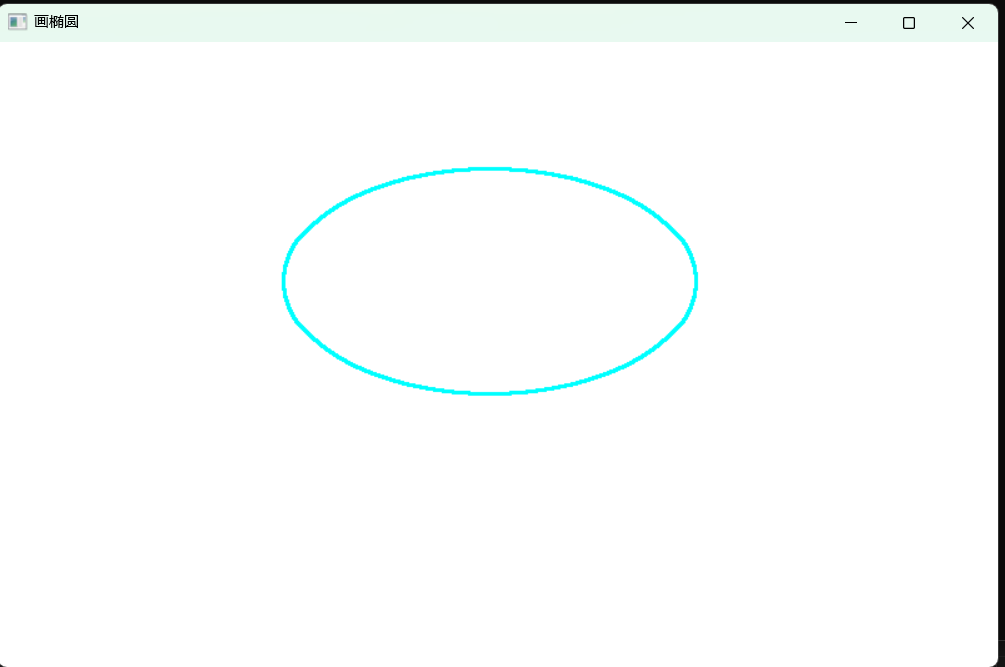
}

glBegin(GL\_POINTS);

OvalPoints(x, y, x0, y0);

}

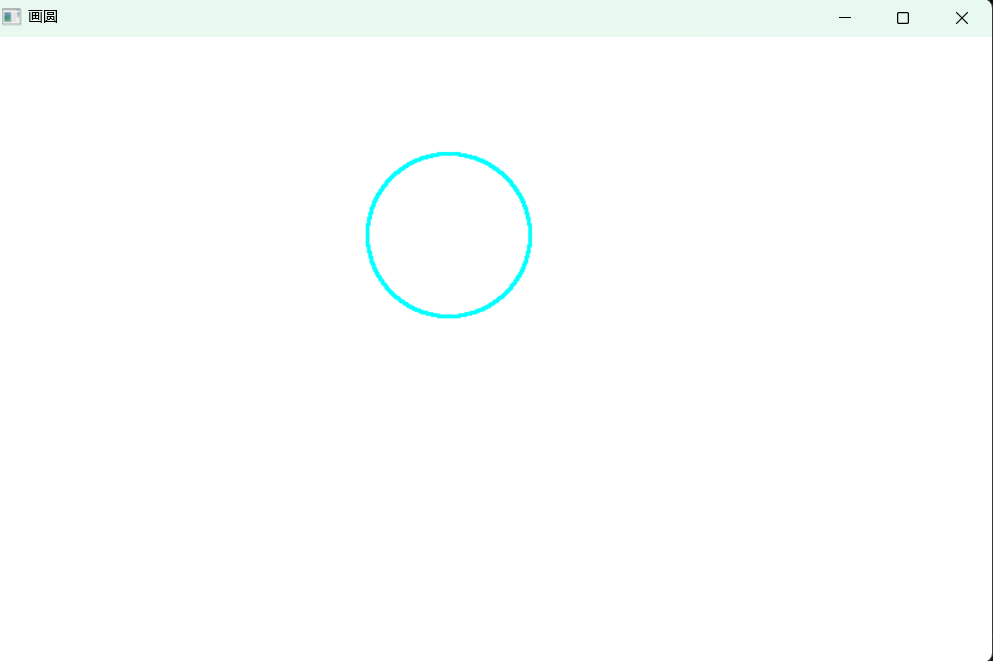
**图片：**

****

图片3 绘制椭圆

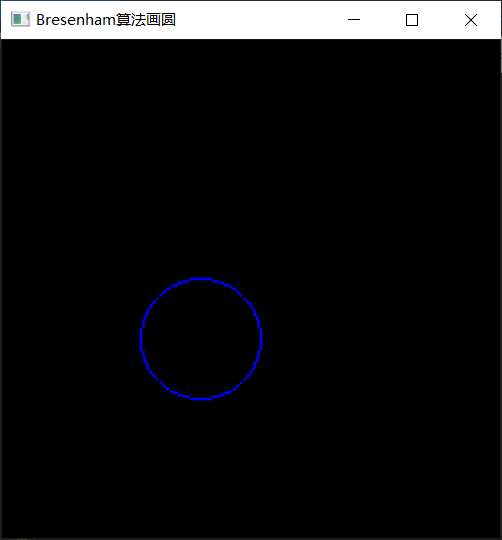
四、实验结果

**1.中心点算法：**



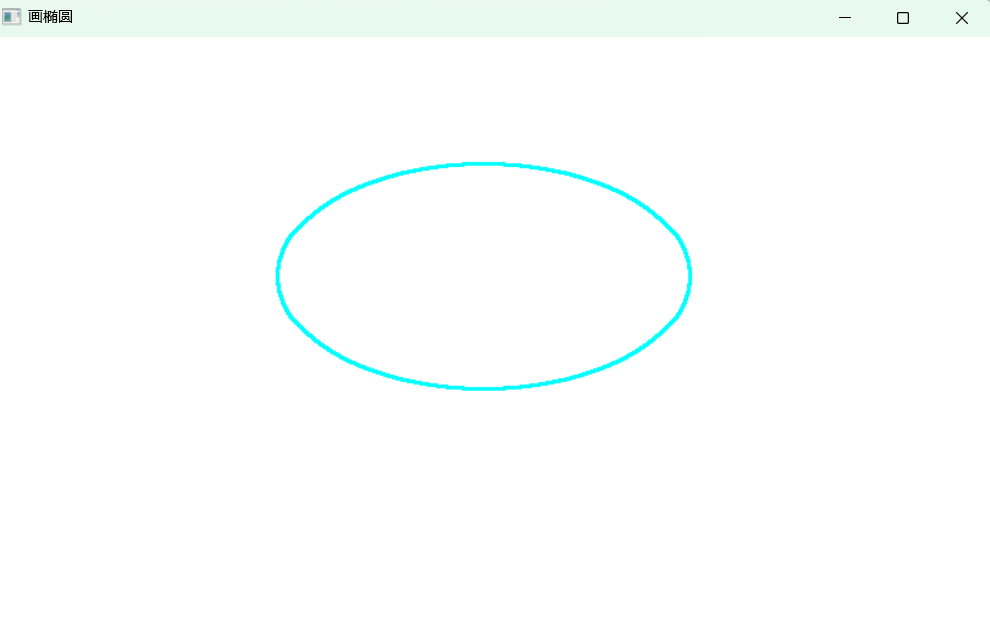
图片4 中心点算法绘制园

**2.Bresenham算法:**



图片5 Bresenham算法绘制园

**3.中心点算法绘制椭圆：**

****

图片6 中心点算法绘制椭圆

五、实验总结

通过本次实验我学会了中点画圆法和Bresenham算法画圆法，同时也使用这两种方法进行椭圆的绘制。在实验过程中，我通过自己实现这两种画图的方法，使我对这两种方法有了更加深刻的体会，同时也了解到这两个算法背后的原理，更能够将这两种方法运用到生活中。同时，也让我明白了之前自己掌握方法的不足，这次实验给我帮助很大。

|  |
| --- |
| **评语**（不要删除，指导老师填写评语，放置在实验报告后面单独成页） |
|  |