**实验名称：**基于中点和Bresenham分别光栅化绘制椭圆

1. **具体内容**

基于中点和Bresenham分别光栅化绘制椭圆

1. **设计思路**

**通过椭圆方程分别用中点和Bresenham推导出递推式，然后利用OpenGL写出代码**

1. **主要函数及其简要说明**

//用bresenham算法绘制椭圆 ab分别为长半轴和短半轴长 默认中心在原点

void BresenhamEllipse(int a, int b)

{

int sqa = a \* a;

int sqb = b \* b;

//x、y初始坐标

int x = 0;

int y = b;

int d = 2 \* sqb - 2 \* b \* sqa + sqa;//初始的判断值

draw(x, y);//绘制

int P\_x = (sqa / sqrt(static\_cast<unsigned \_int64>(sqa) + sqb));

while (x <= P\_x)

{

if (d < 0)

{

d += 2 \* sqb \* (2 \* x + 3);

x++;

}

else

{

d += 2 \* sqb \* (2 \* x + 3) - 4 \* sqa \* (y - 1);

y--;

x++;

}

draw(x, y);

}

d = sqb \* (x \* x + x) + sqa \* (y \* y - y) - sqa \* sqb;

while (y >= 0)

{

if (d < 0)

{

d -= 2 \* sqa \* y + sqa - 2 \* sqb \* x - 2 \* sqb;

y--;

x++;

}

else

{

d -= 2 \* sqa \* y + sqa;

y--;

}

draw(x, y);

}

}

//用中点算法绘制椭圆 ab分别为长半轴和短半轴长 默认中心在原点

void MidpointEllipse(int a, int b)

{

int sqa = a \* a;

int sqb = b \* b;

//x、y初始坐标

int x = 0;

int y = b;

int d = 0.25 \* sqa + sqb;//初始的判断值

draw(x, y);//绘制

int P\_x = (sqa / sqrt(static\_cast<unsigned \_int64>(sqa) + sqb));

while (x <= P\_x)

{

if (d < 0)

{

d += (2 \* x + 3) \* sqb;

x++;

}

else

{

d += (2 \* x + 3) \* sqb + (1 - 2 \* y) \* sqa;

y--;

x++;

}

draw(x, y);

}

x = a;

y = 0;

d = 0.25 \* sqb + sqa;

while (x > P\_x)

{

if (d < 0)

{

d += (2 \* y + 3) \* sqa;

y++;

}

else

{

d += (2 \* y + 3) \* sqa + (1 - 2 \* x) \* sqb;

x--;

y++;

}

draw(x, y);

}

}

1. **遇到的主要问题及解决思路：**

**递推式不会推，借鉴他人推导方式，继续推**

1. **程序运行结果**

