#include<stdlib.h>

#include<stdio.h>

#include <GL/glut.h>

#include <GL/gl.h>

void circlePoints(int xc, int yc, int x, int y)

{

glVertex2i(xc + x, yc + y);

glVertex2i(xc - x, yc + y);

glVertex2i(xc + x, yc - y);

glVertex2i(xc - x, yc - y);

glVertex2i(xc + y, yc + x);

glVertex2i(xc - y, yc + x);

glVertex2i(xc + y, yc - x);

glVertex2i(xc - y, yc - x);

}

void bresenhamCircle(int xc, int yc, int r) {

int x = 0, y = r;

int d = 3 - 2 \* r;

glBegin(GL\_POINTS);

while (y >= x) {

circlePoints(xc, yc, x, y);

x++;

if (d > 0)

{

y--;

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

}

else

{

d = d + 4 \* x + 6;

}

}

glEnd();

}

void drawCirclemidPoint(int xc, int yc, int r)

{

int x = 0;

int y = r;

int p = 1 - r;

glBegin(GL\_POINTS);

circlePoints(xc, yc, x, y);

while (x < y)

{

x++;

if (p < 0)

{

p += 2 \* x + 1;

}

else

{

y--;

p += 2 \* (x - y) + 1;

}

circlePoints(xc, yc, x, y);

}

glEnd();

}

void display()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

drawCirclemidPoint(50, 50, 30);

bresenhamCircle(50, 50, 30);

glFlush();

}

void init() {

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(100, 100);

glutInitWindowPosition(100, 100);

glutCreateWindow("Midpoint Circle Algorithm");

glClearColor(0.0, 0.0, 0.0, 0.0);

gluOrtho2D(0, 100, 0, 100);

}

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

{

glutInit(&argc, argv);

init();

glutDisplayFunc(display);

glutMainLoop();

return 0;

}