#include <Windows.h>

#include <GL/glut.h>

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

int h,k;

float a;

void drawDot (GLint x, GLint y, GLfloat r, GLfloat g, GLfloat b)

{ glColor3f(r,g,b);

glPointSize(1.0);

glBegin (GL\_POINTS);

glVertex2i (x,y);

glEnd();

}

void drawAxis(){

glColor3f(1.0,1.0,1.0);

glBegin(GL\_LINES);

{

glVertex3f(-100,0,0);

glVertex3f(100,0,0);

}glEnd();

glBegin(GL\_LINES);

{

glVertex3f(0,-100,0);

glVertex3f(0,100,0);

}glEnd();

}

void symmetricPixels (int x, int y, int xc, int yc, float r, float g, float b)

{ drawDot(xc+x,yc-y,r,g,b);

drawDot(xc-x,yc-y,r,g,b);

drawDot(xc+x,yc+y,r,g,b);

drawDot(xc-x,yc+y,r,g,b);

drawDot(xc+y,yc-x,r,g,b);

drawDot(xc-y,yc-x,r,g,b);

drawDot(xc+y,yc+x,r,g,b);

drawDot(xc-y,yc+x,r,g,b);

}

void Circle (int a, int xc, int yc, float r, float g, float bl)

{ int x,y,p;

x=0;

y=a;

//drawDot(xc+x,yc-y,1,1,1);

p=3-(2\*a);

for(x=0;x<=y;x++)

{

if (p<0)

{

y=y;

p=(p+(4\*x)+6);

}

else

{

y=y-1;

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

}

symmetricPixels(x,y,xc,yc,r,g,bl);

//delay(50);

}

glFlush();

}

void display(void)

{

double I,J;

int i,j;

glClear (GL\_COLOR\_BUFFER\_BIT);

glColor3f (1.0, 0.0, 0.0);

drawAxis();

glBegin(GL\_POINTS);

Circle (a,h,k,1,1,1);

Sleep(100);

Circle (a,-h,k,1,1,1);

Sleep(100);

Circle (a,h,-k,1,1,1);

Sleep(100);

glEnd();

glFlush();

}

void init(void)

{

glClearColor (0.0, 0.0, 0.0, 0.0);

glOrtho(-100.0, 100.0, -100.0, 100.0, -1.0, 1.0);

}

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

{

printf("Enter the center of Circle:\n");

scanf("%d %d",&h,&k);

printf("Enter the parameters a:\n");

scanf("%f",&a);

glutInit(&argc, argv);

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize (500, 500);

glutInitWindowPosition (100, 100);

glutCreateWindow ("Circle : Polynomial Method ");

init ();

glutDisplayFunc(display);

glutMainLoop();

return 0;

}