import com.jogamp.opengl.GL2;

import com.jogamp.opengl.GLAutoDrawable;

import com.jogamp.opengl.GLCapabilities;

import com.jogamp.opengl.GLEventListener;

import com.jogamp.opengl.GLProfile;

import com.jogamp.opengl.awt.GLCanvas;

import com.jogamp.opengl.glu.GLU;

import java.lang.Math;

import javax.swing.JFrame;

//Md. Shamiul Islam

//ID: 17301108

//section: 13

public class CSE423Lab3Task1 implements GLEventListener {

private GLU glu;

public void init(GLAutoDrawable gld) {

GL2 gl = gld.getGL().getGL2();

glu = new GLU();

gl.glClearColor(1.0f, 1.0f, 1.0f, 1.0f);

gl.glViewport(-250, -150, 250, 150);

gl.glMatrixMode(GL2.GL\_PROJECTION);

gl.glLoadIdentity();

glu.gluOrtho2D(-250.0, 250.0, -150.0, 150.0);

}

/\*\*

\* Take care of drawing here.

\* @param drawable

\*/

public void display(GLAutoDrawable drawable) {

GL2 gl = drawable.getGL().getGL2();

gl.glClear(GL2.GL\_COLOR\_BUFFER\_BIT);

/\*

\* put your code here

\*/

DrawMPC(gl,100,0,0);

DrawMPC(gl,50,0,50);

DrawMPC(gl,50,50,0);

DrawMPC(gl,50,0,-50);

DrawMPC(gl,50,-50,0);

}

@Override

public void reshape(GLAutoDrawable drawable, int x, int y, int width, int height) {

//do nothing

}

public void displayChanged(GLAutoDrawable drawable, boolean modeChanged, boolean deviceChanged) {

//do nothing

}

private void DrawMPC(GL2 gl, double r, double cX, double cY) {

//write your own code

gl.glPointSize(3.1f);

gl.glColor3d(0, 0, 0);

double x,y,d;

x=r;

y=0;

d=5-(4\*r);

gl.glBegin(GL2.GL\_POINTS);

gl.glVertex2d(x+cX, y+cY);

gl.glVertex2d(convertXinit(x,y,1)+cX,convertYinit(x,y,1)+cY);

gl.glVertex2d(convertXinit(x,y,2)+cX,convertYinit(x,y,2)+cY);

gl.glVertex2d(convertXinit(x,y,3)+cX,convertYinit(x,y,3)+cY);

gl.glVertex2d(convertXinit(x,y,4)+cX,convertYinit(x,y,4)+cY);

gl.glVertex2d(convertXinit(x,y,5)+cX,convertYinit(x,y,5)+cY);

gl.glVertex2d(convertXinit(x,y,6)+cX,convertYinit(x,y,6)+cY);

gl.glVertex2d(convertXinit(x,y,7)+cX,convertYinit(x,y,7)+cY);

while(x>y) {

if(d<0) {

d=d+(4\*((2\*y)+3));

y++;

}

else {

d=d+(4\*((2\*y)-(2\*x)+5));

x--;

y++;

}

gl.glVertex2d(x+cX, y+cY);

gl.glVertex2d(convertXinit(x,y,1)+cX,convertYinit(x,y,1)+cY);

gl.glVertex2d(convertXinit(x,y,2)+cX,convertYinit(x,y,2)+cY);

gl.glVertex2d(convertXinit(x,y,3)+cX,convertYinit(x,y,3)+cY);

gl.glVertex2d(convertXinit(x,y,4)+cX,convertYinit(x,y,4)+cY);

gl.glVertex2d(convertXinit(x,y,5)+cX,convertYinit(x,y,5)+cY);

gl.glVertex2d(convertXinit(x,y,6)+cX,convertYinit(x,y,6)+cY);

gl.glVertex2d(convertXinit(x,y,7)+cX,convertYinit(x,y,7)+cY);

}

gl.glEnd();

}

public void dispose(GLAutoDrawable arg0) {

//do nothing

}

double convertX0(double x, double y, double zone) {

double newX=0;

if(zone==0) {newX=x;}

else if(zone==1) {newX=y;}

else if(zone==2) {newX=y;}

else if(zone==3) {newX=-x;}

else if(zone==4) {newX=-x;}

else if(zone==5) {newX=-y;}

else if(zone==6) {newX=-y;}

else if(zone==7) {newX=x;}

return newX;

}

double convertY0(double x, double y, double zone) {

double newY=0;

if(zone==0) {newY=y;}

else if(zone==1) {newY=x;}

else if(zone==2) {newY=-x;}

else if(zone==3) {newY=y;}

else if(zone==4) {newY=-y;}

else if(zone==5) {newY=-x;}

else if(zone==6) {newY=x;}

else if(zone==7) {newY=-y;}

return newY;

}

double convertXinit(double x, double y, double zone) {

double newX=0;

if(zone==0) {newX=x;}

else if(zone==1) {newX=y;}

else if(zone==2) {newX=-y;}

else if(zone==3) {newX=-x;}

else if(zone==4) {newX=-x;}

else if(zone==5) {newX=-y;}

else if(zone==6) {newX=y;}

else if(zone==7) {newX=x;}

return newX;

}

double convertYinit(double x, double y, double zone) {

double newY=0;

if(zone==0) {newY=y;}

else if(zone==1) {newY=x;}

else if(zone==2) {newY=x;}

else if(zone==3) {newY=y;}

else if(zone==4) {newY=-y;}

else if(zone==5) {newY=-x;}

else if(zone==6) {newY=-x;}

else if(zone==7) {newY=-y;}

return newY;

}

double convertX(double x, double y, double zone){

double convertedX=0;

if(zone==0) {

convertedX=x;

}

else if(zone==1) {

convertedX=y;

}

else if(zone==2) {

convertedX=y;

}

else if(zone==3) {

convertedX=-x;

}

else if(zone==4) {

convertedX=-x;

}

else if(zone==5) {

convertedX=-y;

}

else if(zone==6) {

convertedX=-y;

}

else if(zone==7) {

convertedX=x;

}

return convertedX;

}

double convertY(double x, double y, double zone){

double convertedY=0;

if(zone==0) {

convertedY=y;

}

else if(zone==1) {

convertedY=x;

}

else if(zone==2) {

convertedY=-x;

}

else if(zone==3) {

convertedY=y;

}

else if(zone==4) {

convertedY=-y;

}

else if(zone==5) {

convertedY=-x;

}

else if(zone==6) {

convertedY=x;

}

else if(zone==7) {

convertedY=-y;

}

return convertedY;

}

public static void main(String[] args) {

//getting the capabilities object of GL2 profile

final GLProfile profile = GLProfile.get(GLProfile.GL2);

GLCapabilities capabilities = new GLCapabilities(profile);

// The canvas

final GLCanvas glcanvas = new GLCanvas(capabilities);

CSE423Lab3Task1 drawing = new CSE423Lab3Task1();

glcanvas.addGLEventListener(drawing);

glcanvas.setSize(550, 550);

//creating frame

final JFrame frame = new JFrame ("MidPoint Circle");

//adding canvas to frame

frame.getContentPane().add(glcanvas);

frame.setSize(frame.getContentPane().getPreferredSize());

frame.setVisible(true);

}

}