TASK----02

// My Id is 19201101.So I have to draw a boat.

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 javax.swing.JFrame;

public class Assignment02 implements GLEventListener{

private GLU glu;

@Override

public void display(GLAutoDrawable drawable) {

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

gl.glColor3d(1, 0, 0);

MidPointLDA(gl, -35, -40, 35, -40);

MidPointLDA(gl, -50, -20, 50, -20);

MidPointLDA(gl, 50, -20, 35, -40);

MidPointLDA(gl, -50, -20, -35, -40);

MidPointLDA(gl, 0, 20, 1, -20);

MidPointLDA(gl, 0, 10, -30, -20);

MidPointLDA(gl, 0, 20, 30, -20);

}

@Override

public void dispose(GLAutoDrawable arg0) {

}

@Override

public void init(GLAutoDrawable gld) {

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

glu = new GLU();

gl.glClearColor(0.0f, 0.0f, 0.0f, 0.0f);

gl.glViewport(-100, -50, 50, 100);

gl.glMatrixMode(GL2.GL\_PROJECTION);

gl.glLoadIdentity();

glu.gluOrtho2D(-100.0, 100.0, -100.0, 100.0);

}

@Override

public void reshape(GLAutoDrawable arg0, int arg1, int arg2, int arg3, int arg4) {

}

public void MidPointLDA(GL2 gl, int x1, int y1, int x2, int y2) {

gl.glPointSize(3.0f);

int dx = x2-x1;

int dy = y2-y1;

int zone = findZone(dx, dy);

int[] newCoordinates = convertToZone\_0(gl, x1, y1, x2, y2,zone);

int newX1 = newCoordinates[0];

int newY1 = newCoordinates[1];

int newX2 = newCoordinates[2];

int newY2 = newCoordinates[3];

dx = newX2-newX1;

dy = newY2-newY1;

int d = 2 \* dy - dx;

int incE = 2 \* dy;

int incNE = 2 \* (dy-dx);

int plotX = newX1;

int plotY = newY1;

while(plotX <= newX2) {

OriginalZone(gl, plotX, plotY, zone);

plotX++;

if(d > 0) {

d+= incNE;

plotY++;

}else {

d += incE;

}

}

}

public static int findZone(float dx, float dy) {

int zone = 0;

if (Math.abs(dx)>=Math.abs(dy)) {

if (dx>=0 && dy>0) {

zone = 0;

}

else if (dx<0 && dy>0) {

zone = 3;

}

else if (dx>0 && dy<0) {

zone = 7;

}

else if (dx<0 && dy<0) {

zone = 4;

}

} else {

if (dx>0 && dy>0) {

zone = 1;

}

else if (dx<0 && dy>0) {

zone = 2;

}

else if (dx>0 && dy<0) {

zone = 6;

}

else if (dx<0 && dy<0) {

zone = 5;

}

}

return zone;

}

public static int[] convertToZone\_0 (GL2 gl, int X1, int Y1, int X2, int Y2, int zone) {

int x1 = X1;

int y1 = Y1;

int x2 = X2;

int y2 = Y2;

if (zone == 0) {

x1 = X1;

y1 = Y1;

x2 = X2;

y2 = Y2;

}

else if (zone == 1) {

x1 = Y1;

y1 = X1;

x2 = Y2;

y2 = X2;

} else if (zone == 2) {

x1 = Y1;

y1 = -X1;

x2 = Y2;

y2 = -X2;

} else if (zone == 3) {

x1 = -X1;

y1 = Y1;

x2 = -X2;

y2 = Y2;

} else if (zone == 4) {

x1 = -X1;

y1 = -Y1;

x2 = -X2;

y2 = -Y2;

} else if (zone == 5) {

x1 = -Y1;

y1 = -X1;

x2 = -Y2;

y2 = -X2;

} else if (zone == 6) {

x1 = -Y1;

y1 = X1;

x2 = -Y2;

y2 = X2;

} else if (zone == 7) {

x1 = X1;

y1 = -Y1;

x2 = X2;

y2 = -Y2;

}

return new int[] {x1,y1,x2,y2};

}

public static void OriginalZone(GL2 gl, int X, int Y, int zone) {

int a = 0,b = 0;

if (zone ==0) {

a = X;

b = Y;

}

else if (zone == 1) {

a = Y;

b = X;

} else if (zone == 2) {

a = -Y;

b = X;

} else if (zone == 3) {

a = -X;

b = Y;

} else if (zone == 4) {

a = -X;

b = -Y;

} else if (zone == 5) {

a = -Y;

b = -X;

} else if (zone == 6) {

a = Y;

b = -X;

} else if (zone == 7) {

a = X;

b = -Y;

}

plot(gl, a, b);

}

public static void plot(GL2 gl, int x, int y) {

gl.glBegin(GL2.GL\_POINTS);

gl.glVertex2d(x, y);

gl.glEnd();

}

public static void main(String[] args) {

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

GLCapabilities capabilities = new GLCapabilities(profile);

final GLCanvas glcanvas = new GLCanvas(capabilities);

Assignment02 l = new Assignment02();

glcanvas.addGLEventListener(l);

glcanvas.setSize(600, 600);

final JFrame frame = new JFrame ("straight Line");

frame.getContentPane().add(glcanvas);

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

frame.setVisible(true);

}

}