СПбНИУ ИТМО

Отчет

По лабораторной работе №4

«Программирование интернет приложений»

Выполнил: Ратников В.И

Группа 3105

Преподаватель: Гаврилов А.В.

**Исходный код программы**

**файл Lab4.java**

import java.awt.BorderLayout;

import java.awt.Canvas;

import java.awt.Color;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.Panel;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.MouseEvent;

import java.awt.event.MouseListener;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JComboBox;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JRadioButton;

import javax.swing.JSlider;

import javax.swing.SwingUtilities;

import javax.swing.event.ChangeEvent;

import javax.swing.event.ChangeListener;

import var333.Contour;

import var333.Mark;

public class Lab4 implements Runnable

{

//глобальные переменные: радиус,контур,точка из прошлой лабы

public static int radius = 1,

curr\_x,

curr\_y;

public static Contour contour;

public static Mark mark;

//элементы управления

public static JFrame form;

public static Panel p\_elements,p\_canvas;

public static Canvas paint\_area;

public static JComboBox x\_coord;

public static JRadioButton y\_1,y\_2,y\_3,y\_4;

public static JSlider radius\_slider;

public static JLabel label\_coords;

//слушатели событий

ActionListener x\_changed;

ActionListener y\_changed;

ChangeListener radius\_changed;

MouseListener mouse\_click;

//точка входа

public static void main(String[] args)

{

SwingUtilities.invokeLater(new Lab4());

}

public void paint\_all()

{

paint\_area.getGraphics().clearRect(0, 0, paint\_area.getWidth(), paint\_area.getHeight());

painter.Painter.draw\_var333\_figure(paint\_area, 25, radius);

painter.Painter.draw\_grid\_on\_panel(paint\_area,25);

}

//инициализация глобальных переменных

public void init\_globals()

{

mark = new Mark(curr\_x, curr\_y);

contour = new Contour(radius);

form.setTitle("radius = " + String.valueOf(radius));

}

//инициализация элементов управления

public void init\_elements()

{

form = new JFrame("Lab4,var 333");

form.getContentPane().setLayout(new BorderLayout());

//панель с элементами управления

p\_elements = new Panel(new GridLayout(7,1));

p\_elements.setForeground(Color.WHITE);

p\_elements.add(x\_coord = new JComboBox());

x\_coord.addItem("X=1");

x\_coord.addItem("X=2");

x\_coord.addItem("X=3");

x\_coord.addItem("X=4");

p\_elements.add(y\_1 = new JRadioButton("Y=1"));

p\_elements.add(y\_2 = new JRadioButton("Y=2"));

p\_elements.add(y\_3 = new JRadioButton("Y=3"));

p\_elements.add(y\_4 = new JRadioButton("Y=4"));

p\_elements.add(radius\_slider = new JSlider(1, 10, 1));

p\_elements.add(label\_coords = new JLabel("here will be coordinates"));

label\_coords.setFont(new Font(Font.MONOSPACED, Font.BOLD, 20));

//панель с областью рисования

p\_canvas = new Panel(new GridLayout());

paint\_area = new Canvas();

paint\_area.setBackground(new Color(0,250,0));

paint\_area.setSize(400, 300);

p\_canvas.add(paint\_area);

form.add(p\_elements,BorderLayout.WEST);

form.add(p\_canvas,BorderLayout.CENTER);

form.pack();

form.setVisible(true);

}

//инициализация событий

public void init\_events()

{

//обработчик события смены радиуса

radius\_changed = new ChangeListener() {

@Override

public void stateChanged(ChangeEvent e)

{

//поменять радиус

radius = radius\_slider.getValue();

radius\_slider.setToolTipText( String.valueOf(radius\_slider.getValue()) );

//пересчитать контур

init\_globals();

paint\_all();

}

};

radius\_slider.addChangeListener(radius\_changed);

//обработчик события смены координаты X

x\_changed = new ActionListener()

{

@Override

public void actionPerformed(ActionEvent e)

{

curr\_x = x\_coord.getSelectedIndex()+1;

init\_globals();

label\_coords.setText(mark.toString());

try

{

painter.Painter.draw\_animate\_cursor(paint\_area, 25,(int)Mark.Translate\_reverse(mark, paint\_area, 25).getX(), (int)Mark.Translate\_reverse(mark, paint\_area, 25).getY(), radius);

} catch (InterruptedException ex)

{

Logger.getLogger(Lab4.class.getName()).log(Level.SEVERE, null, ex);

}

}

};

x\_coord.addActionListener(x\_changed);

//обработчик события смены координаты Y

y\_changed = new ActionListener()

{

@Override

public void actionPerformed(ActionEvent e)

{

//some indian code

if(y\_1.isSelected())

{

y\_2.setSelected(false);

y\_3.setSelected(false);

y\_4.setSelected(false);

curr\_y = 1;

}

if(y\_2.isSelected())

{

y\_1.setSelected(false);

y\_3.setSelected(false);

y\_4.setSelected(false);

curr\_y = 2;

}

if(y\_3.isSelected())

{

y\_1.setSelected(false);

y\_2.setSelected(false);

y\_4.setSelected(false);

curr\_y = 3;

}

if(y\_4.isSelected())

{

y\_1.setSelected(false);

y\_2.setSelected(false);

y\_3.setSelected(false);

curr\_y = 4;

}

init\_globals();

try

{

painter.Painter.draw\_animate\_cursor(paint\_area, 25,(int)Mark.Translate\_reverse(mark, paint\_area, 25).getX(), (int)Mark.Translate\_reverse(mark, paint\_area, 25).getY(), radius);

} catch (InterruptedException ex)

{

Logger.getLogger(Lab4.class.getName()).log(Level.SEVERE, null, ex);

}

label\_coords.setText(mark.toString());

}

};

y\_1.addActionListener(y\_changed);

y\_2.addActionListener(y\_changed);

y\_3.addActionListener(y\_changed);

y\_4.addActionListener(y\_changed);

mouse\_click = new MouseListener() {

@Override

public void mouseClicked(MouseEvent e) {

label\_coords.setText("Clicked at " + e.getX() + ":" + e.getY());

try

{

painter.Painter.draw\_animate\_cursor(paint\_area, 25,e.getX(), e.getY(), radius);

label\_coords.setText(Mark.Translate(paint\_area, e.getX(), e.getY(), 25).toString());

} catch (InterruptedException ex)

{

Logger.getLogger(Lab4.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

public void mousePressed(MouseEvent e) {

//label\_coords.setText("Pressed");

//throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

@Override

public void mouseReleased(MouseEvent e) {

//label\_coords.setText("Released");

//throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

@Override

public void mouseEntered(MouseEvent e) {

//label\_coords.setText("Entered at ");

// throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

@Override

public void mouseExited(MouseEvent e) {

//label\_coords.setText("Exited");

// throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

};

paint\_area.addMouseListener(mouse\_click);

}

@Override

public void run()

{

init\_elements();

init\_events();

init\_globals();

}

}

**Файл Painter.java**

import java.awt.Canvas;

import java.awt.Color;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.Panel;

import java.awt.Polygon;

import java.awt.Shape;

import java.awt.geom.Ellipse2D;

import java.util.Random;

import javax.swing.DebugGraphics;

import var333.Mark;

public class Painter implements Runnable

{

public static void draw\_grid\_on\_panel(Canvas \_canvas,int ppd)

{

int h = \_canvas.getHeight();

int w = \_canvas.getWidth();

Graphics g = \_canvas.getGraphics();

//g.clearRect(0, 0, w, h);

//horisontal\_line

g.drawLine(0, h/2,w,h/2);

//vertical line

g.drawLine(w/2,0,w/2,h);

for(int i=0;i<w/ppd/2;i++)

{

//horosontal gaps

g.drawLine(w/2 + i\*ppd, h/2-4,w/2 + i\*ppd, h/2+4);

g.drawLine(w/2 - i\*ppd, h/2-4,w/2 - i\*ppd, h/2+4);

//vertical gaps

g.drawLine(w/2-4, h/2 - i\*ppd ,w/2+4, h/2 - i\*ppd );

g.drawLine(w/2-4, h/2 + i\*ppd ,w/2+4, h/2 + i\*ppd );

}

}

public static void draw\_var333\_figure(Canvas \_canvas,int ppd,int radius)

{

int h = \_canvas.getHeight();

int w = \_canvas.getWidth();

Graphics g = \_canvas.getGraphics();

//нарисуем окружность

g.setColor(new Color(0,180,0));

g.fillOval(w/2 - radius\*ppd, h/2 - radius\*ppd, radius\*ppd\*2 , radius\*ppd\*2);

//оставим только 4 четверть окружности

g.setColor(new Color(0,250,0));

g.fillRect(0, 0, w/2, h/2);

//нарисуем треугольник

g.setColor(new Color(0,180,0));

Polygon p = new Polygon();

p.addPoint(w/2, h/2);

p.addPoint(w/2+radius\*ppd\*2, h/2);

p.addPoint(w/2 , h/2 - radius\*ppd\*2 );

p.addPoint(w/2, h/2);

g.fillPolygon(p);

g.fillRect(w/2 - radius\*ppd , h/2, radius\*ppd, radius\*ppd\*2);

}

public static void draw\_animate\_cursor(Canvas \_canvas,int ppd,int X,int Y,int radius) throws InterruptedException

{

int h = \_canvas.getHeight();

int w = \_canvas.getWidth();

Graphics g = \_canvas.getGraphics();

g.setXORMode(g.getColor());

g.setColor(Color.red);

float r;

for(r=radius/5.f;r>radius/10.f;r-=radius/20.f)

{

g.fillOval((int)(X-Math.round(r\*2)), (int)(Y-Math.round(r\*2)), (int)(ppd\*r), (int)(ppd\*r) );

Thread.sleep(50);

g.fillOval((int)(X-Math.round(r\*2)), (int)(Y-Math.round(r\*2)), (int)(ppd\*r), (int)(ppd\*r) );

}

if(lab4.Lab4.contour.is\_hit(Mark.Translate(\_canvas, X, Y, ppd)))

g.setColor(Color.BLUE);

g.fillOval((int)(X-Math.round(r\*4)), (int)(Y-Math.round(r\*2)), (int)(ppd\*r), (int)(ppd\*r) );

}

@Override

public void run()

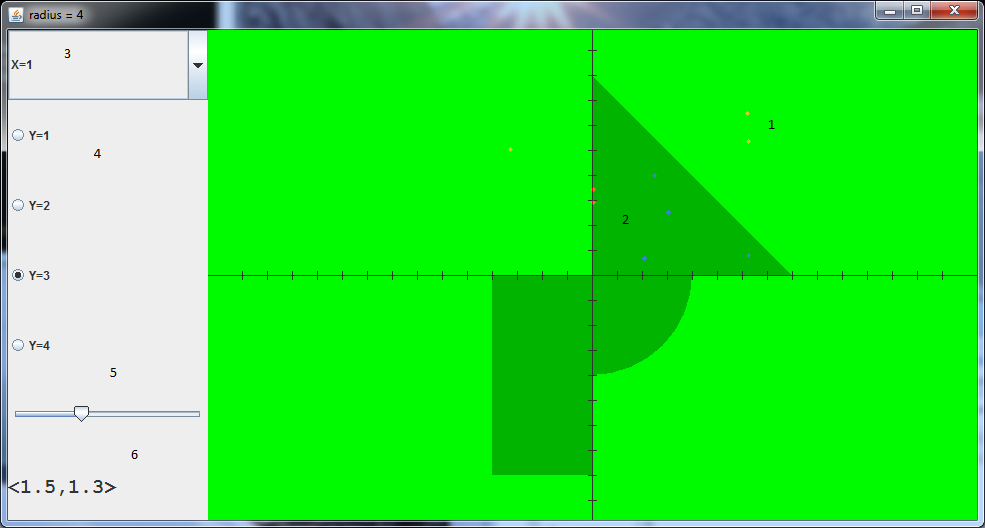
{

//throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.

}

}

**Интерфейс программы**



описание :

1. красные точки. не попали в область.

2. синие точки. попали в область

3. элемент JComboBox служит для смены координаты X

4. группа элементов JRadioButton служит для смены координаты Y

5. элемент JSlider служит для смены радиуса

6. элемент JLabel отобраает текущую точку.