SPRAWOZDANIE

Zajęcia: Grafika komputerowa
Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium 1
22.02.2022
Temat:" Przekształcenia 2D w bibliotece Java2D "
Wariant ??

Bartosz Medoń
Informatyka I stopień,
stacjonarne,
4 semestr,
Gr.1a

1. Polecenie: Dodać opcję przekształceń do klasy Transforms2D oraz dodać wyświetlanie odpowiedniego wielokąta zamiast obrazka. Wyświetlić odpowiedni wariant obrazka.

2. Wykorzystane komendy:

a) Wielokąt (14kąt)

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.imageio.ImageIO;
import java.awt.image.BufferedImage;
import java.io.IOException;

public class Transforms2D extends JPanel {

	private class Display extends JPanel {
		protected void paintComponent(Graphics g) {
			super.paintComponent(g);
			Graphics2D g2 = (Graphics2D)g;
			g2.translate(300,300);
			int whichTransform = transformSelect.getSelectedIndex();
			double pi = Math.PI;

			switch(whichTransform){
			 case 0:
		    g2.rotate(0);
		    break;
		case 1:
		 g2.scale(0.5,0.5);
		    break;
		case 2:
		 g2.rotate(pi/4);
		 break;
		case 3:
		 g2.scale(-0.4,1);
		 g2.rotate(pi);
		 break;
		case 4:
		 g2.shear(0.5,0);
		 break;
		case 5:
		 g2.scale(1,0.4);
		 g2.translate(0,-620);
		    break;
		case 6:
		 g2.rotate(pi/2);
		 g2.shear(0.5,0);
		 break;
		case 7:
		 g2.scale(0.4,1);
		 g2.rotate(pi);
```

```
                break;
               case 8:
                g2.rotate(pi/6);
                g2.scale(1,0.4);
                g2.translate(25,350);
                break;
               case 9:
                g2.rotate(pi);
                g2.shear(0, 0.5);
                g2.translate(-159,50);
                break;
              default:
                  g2.rotate(0);

                      }

                      Polygon p = new Polygon();

                      for (int i=0;i<14;i++)
                      {

p.addPoint((int)(150*Math.cos(i*2*pi/14)),(int)(150*Math.sin(i*2*pi/14)));
                      }

                      g.drawPolygon(p);
                      g2.fill(p);
               }
        }

        private Display display;
        private BufferedImage pic;
        private JComboBox<String> transformSelect;

        public Transforms2D() throws IOException {
               display = new Display();
               display.setBackground(Color.YELLOW);
               display.setPreferredSize(new Dimension(600,600));
               transformSelect = new JComboBox<String>();
               transformSelect.addItem("None");
               for (int i = 1; i < 10; i++) {
                     transformSelect.addItem("No. " + i);
               }
               transformSelect.addActionListener( new ActionListener() {

                      public void actionPerformed(ActionEvent e) {
                            display.repaint();
                      }
               });
               setLayout(new BorderLayout(3,3));
               setBackground(Color.GRAY);
               setBorder(BorderFactory.createLineBorder(Color.GRAY,10));
               JPanel top = new JPanel();
               top.setLayout(new FlowLayout(FlowLayout.CENTER));
               top.setBorder(BorderFactory.createEmptyBorder(4, 4, 4, 4));
               top.add(new JLabel("Transform: "));
               top.add(transformSelect);
               add(display,BorderLayout.CENTER);
               add(top,BorderLayout.NORTH);
        }
```

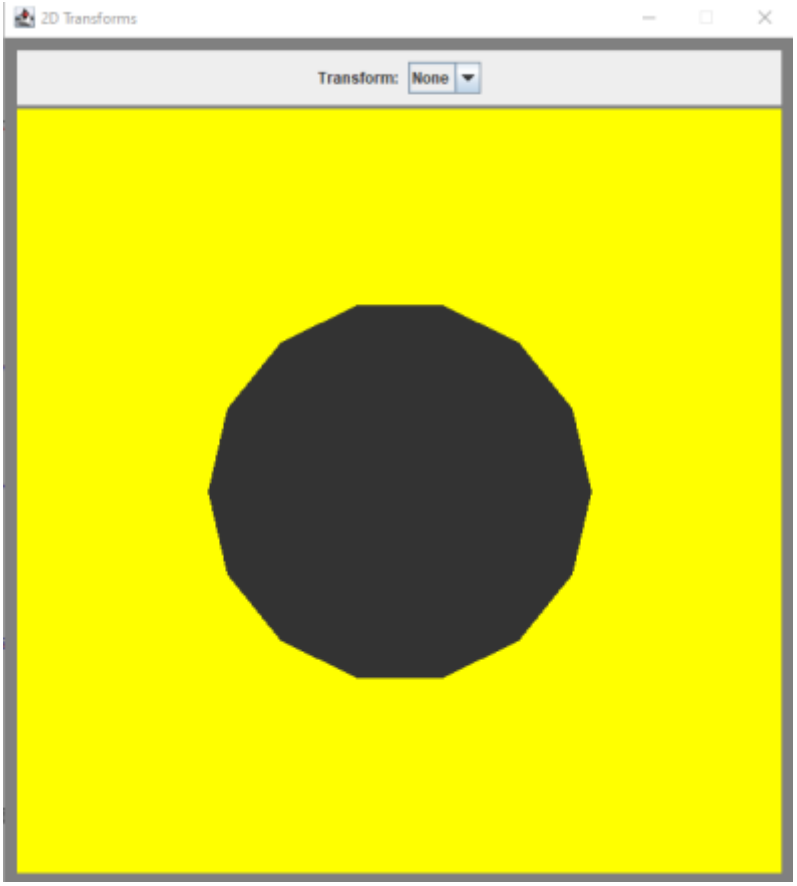
```
	public static void main(String[] args) throws IOException {
		JFrame window = new JFrame("2D Transforms");
		window.setContentPane(new Transforms2D());
		window.pack();
		window.setResizable(false);
		window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		Dimension screen = Toolkit.getDefaultToolkit().getScreenSize();
		window.setLocation( (screen.width - window.getWidth())/2,
(screen.height - window.getHeight())/2 );
		window.setVisible(true);
	}

   }
```

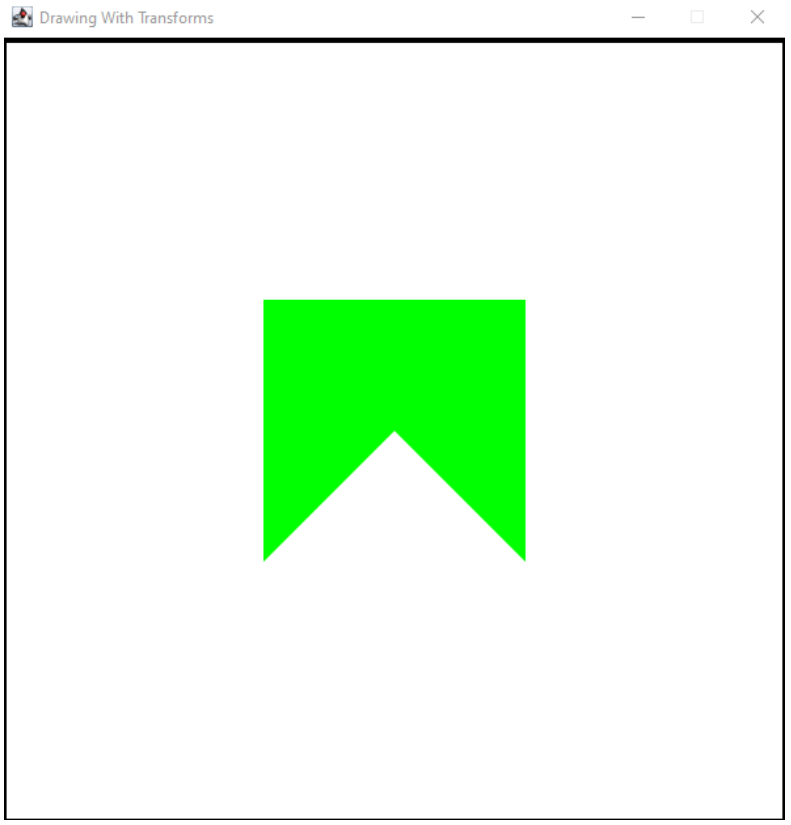
b) Grafika (wariant 2)

```
package lab1a;

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.awt.geom.AffineTransform;


public class TransformedShapes extends JPanel {


	private Graphics2D g2;
	private void resetTransform() {
		g2.setTransform(new AffineTransform());
	}

	private void square() {
		g2.fillRect(-50,-50,100,100);
	}


	private void triangle() {
		g2.fillPolygon(new int[] {-50,50,0}, new int[] {50,50,-50}, 3);

	}

	//-------------------------------------------------------------------------
----------


	protected void paintComponent(Graphics g) {
		super.paintComponent(g);
		g2 = (Graphics2D)g.create();
		g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
RenderingHints.VALUE_ANTIALIAS_ON);

		g2.translate(300,300);
		g2.scale(2, 2);
```

```
                g2.setColor(Color.green);
                square();
                g2.translate(0, 25);
                g2.scale(1,0.5);
                g2.setColor(Color.white);
                triangle();
        }

        public TransformedShapes() {
                setPreferredSize(new Dimension(600,600) );
                setBackground(Color.WHITE);
                setBorder(BorderFactory.createLineBorder(Color.BLACK,4));
        }

        public static void main(String[] args)  {
                JFrame window = new JFrame("Drawing With Transforms");
                window.setContentPane(new TransformedShapes());
                window.pack();
                window.setResizable(false);
                window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                Dimension screen = Toolkit.getDefaultToolkit().getScreenSize();
                window.setLocation( (screen.width - window.getWidth())/2,
(screen.height - window.getHeight())/2 );
                window.setVisible(true);
        }

}
```

4. Wynik działania:





5. **Wnioski:**Na podstawie otrzymanego wyniku można stwierdzić, że biblioteka java 2D umożliwia tworzenie obrazów 2D.