

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ
ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»**

**Факультет прикладної математики
Кафедра програмного забезпечення комп'ютерних систем**

Лабораторна робота №3
з дисципліни “Математичні та алгоритмічні основи комп'ютерної графіки”

Виконала
студентка 3 курсу
групи КП-83:
Тиченко Анна
Варіант: 20

Київ 2021

Завдання:

За допомогою примітивів JavaFX максимально реально зобразити персонажа за варіантом та виконати його 2D анімацію. Для анімації скористатися стандартними засобами бібліотеки JavaFX.

Обов'язковою є реалізація таких видів анімації:

- 1) переміщення;
- 2) поворот;
- 3) масштабування

Код програми

FirTree.java

```
public class Bird extends Application {

    public static void main(String args[]) {
        launch(args);
    }

    @Override
    public void start(Stage primaryStage) {
        Group root = new Group();
        // Scene (root, x, y)
        Scene scene = new Scene(root, 800, 600);
        scene.setFill(Color.rgb(167, 215, 243));

        //body

        Ellipse ellipse = new Ellipse(400, 250, 90, 100);
        ellipse.setFill(Color.BLACK);
        ellipse.setStroke(Color.BLACK);
        ellipse.setStrokeType(StrokeType.OUTSIDE);
        root.getChildren().add(ellipse);

        //stomach

        Arc arc = new Arc();
        arc.setFill(Color.rgb(67, 67, 67));
        arc.setType(ArcType.OPEN);
        arc.setCenterX(400.0f);
        arc.setCenterY(340.0f);
        arc.setRadiusX(40.0f);
```

```

arc.setRadiusY(40.0f);
arc.setLength(180.0f);
root.getChildren().add(arc);

//stomach2

arc = new Arc();
arc.setFill(Color.rgb(67, 67, 67));
arc.setType(ArcType.OPEN);
arc.setCenterX(400.0f);
arc.setCenterY(340.0f);
arc.setRadiusX(36.0f);
arc.setRadiusY(9.0f);
arc.setStartAngle(180.0f);
arc.setLength(180.0f);
root.getChildren().add(arc);

//tail

Rectangle rect = new Rectangle(370, 100, 70, 20);
rect.setFill(Color.BLACK);
rect.setStroke(Color.BLACK);
rect.setStrokeType(StrokeType.OUTSIDE);
Rotate rotate = new Rotate();
rotate.setPivotX(rect.getX()); //Pivot X Top-Left corner
rotate.setPivotY(rect.getY()); //Pivot Y
rotate.setAngle(50); //Angle degrees
//Add the transform to the node
rect.getTransforms().add(rotate);
root.getChildren().add(rect);

//tail2

rect = new Rectangle(370, 100, 30, 20);
rect.setFill(Color.rgb(254, 180, 0));
rect.setStroke(Color.BLACK);
rect.setStrokeType(StrokeType.OUTSIDE);
rotate = new Rotate();
rotate.setPivotX(rect.getX()); //Pivot X Top-Left corner
rotate.setPivotY(rect.getY()); //Pivot Y
rotate.setAngle(50); //Angle degrees
//Add the transform to the node
rect.getTransforms().add(rotate);
root.getChildren().add(rect);

//blick

ellipse = new Ellipse(420, 200, 15, 10);
ellipse.setFill(Color.WHITE);
rotate = new Rotate();
rotate.setPivotX(455); //Pivot X Top-Left corner
rotate.setPivotY(180); //Pivot Y
rotate.setAngle(30); //Angle degrees

```

```
ellipse.getTransforms().add(rotate);
root.getChildren().add(ellipse);

//left circle around eye

ellipse = new Ellipse(365, 220, 20, 15);
ellipse.setFill(Color.rgb(67, 67, 67));
root.getChildren().add(ellipse);

//right circle around eye

ellipse = new Ellipse(465, 220, 20, 15);
ellipse.setFill(Color.rgb(67, 67, 67));
root.getChildren().add(ellipse);

//left eye

arc = new Arc();
arc.setFill(Color.WHITE);
arc.setType(ArcType.OPEN);
arc.setCenterX(370.0f);
arc.setCenterY(220.0f);
arc.setRadiusX(15.0f);
arc.setRadiusY(20.0f);
arc.setLength(180.0f);
root.getChildren().add(arc);

//right eye

arc = new Arc();
arc.setFill(Color.WHITE);
arc.setType(ArcType.OPEN);
arc.setCenterX(460.0f);
arc.setCenterY(220.0f);
arc.setRadiusX(15.0f);
arc.setRadiusY(20.0f);
arc.setLength(180.0f);
root.getChildren().add(arc);

//left pupil

Circle circle = new Circle(375, 210, 5);
circle.setFill(Color.BLACK);
root.getChildren().add(circle);

// //right pupil

circle = new Circle(465, 210, 5);
circle.setFill(Color.BLACK);
root.getChildren().add(circle);

//eyebrow left
```

```

rect = new Rectangle(345, 175, 50, 10);
rect.setFill(Color.rgb(202, 54, 0));
rotate = new Rotate();
rotate.setPivotX(rect.getX()); //Pivot X Top-Left corner
rotate.setPivotY(rect.getY()); //Pivot Y
rotate.setAngle(30); //Angle degrees
//Add the transform to the node
rect.getTransforms().add(rotate);
root.getChildren().add(rect);

//eyebrow right

rect = new Rectangle(490, 185, 50, 10);
rect.setFill(Color.rgb(202, 54, 0));
rotate = new Rotate();
rotate.setPivotX(rect.getX()); //Pivot X Top-Left corner
rotate.setPivotY(rect.getY()); //Pivot Y
rotate.setAngle(150); //Angle degrees
//Add the transform to the node
rect.getTransforms().add(rotate);
root.getChildren().add(rect);

//beak down

ellipse = new Ellipse(500, 300, 20, 16);
ellipse.setFill(Color.rgb(255, 179, 0));
rotate = new Rotate();
rotate.setPivotX(468); //Pivot X Top-Left corner
rotate.setPivotY(268); //Pivot Y
rotate.setAngle(140); //Angle degrees
ellipse.getTransforms().add(rotate);
root.getChildren().add(ellipse);

//beak up

Polygon polygon = new Polygon(
    400.0, 240.0,
    420.0, 230.0,
    470.0, 260.0
);
polygon.setFill(Color.rgb(255, 179, 0));
polygon.setStroke(Color.BLACK);
polygon.setStrokeType(StrokeType.OUTSIDE);
root.getChildren().add(polygon);

//teeth

polygon = new Polygon(
    400.0, 240.0,
    435.0, 251.0,
    405.0, 255.0
);
polygon.setFill(Color.WHITE);

```

```
    polygon.setStroke(Color.BLACK);
    polygon.setStrokeType(StrokeType.OUTSIDE);
    root.getChildren().add(polygon);

    //beak down

    polygon = new Polygon(
        405.0, 255.0,
        430.0, 249.0,
        458.0, 258.0,
        436.0, 274.0,
        406.0, 274.0
    );
    polygon.setFill(Color.rgb(255, 179, 0));
    root.getChildren().add(polygon);

    // Animation
    int cycleCount = 2;
    int time = 2000;

    ScaleTransition scaleTransition = new ScaleTransition(
        Duration.millis(time),
        root
    );
    scaleTransition.setToX(2);
    scaleTransition.setToY(2);
    scaleTransition.setAutoReverse(true);

    RotateTransition rotateTransition = new RotateTransition(
        Duration.millis(time),
        root
    );
    rotateTransition.setByAngle(360f);
    rotateTransition.setCycleCount(cycleCount);
    rotateTransition.setAutoReverse(true);

    TranslateTransition translateTransition = new TranslateTransition(
        Duration.millis(time),
        root
    );
    translateTransition.setFromX(150);
    translateTransition.setToX(50);
    translateTransition.setCycleCount(cycleCount + 1);
    translateTransition.setAutoReverse(true);

    TranslateTransition translateTransition2 = new TranslateTransition(
        Duration.millis(time),
        root
    );
    translateTransition2.setFromX(50);
    translateTransition2.setToX(150);
    translateTransition2.setCycleCount(cycleCount + 1);
    translateTransition2.setAutoReverse(true);
```

```
ScaleTransition scaleTransition2 = new ScaleTransition(
    Duration.millis(time),
    root
);
scaleTransition2.setToX(0.1);
scaleTransition2.setToY(0.1);
scaleTransition2.setCycleCount(cycleCount);
scaleTransition2.setAutoReverse(true);

ParallelTransition parallelTransition = new ParallelTransition();
parallelTransition
    .getChildren()
    .addAll(
        rotateTransition,
        scaleTransition,
        scaleTransition2,
        translateTransition
    );
parallelTransition.setCycleCount(Timeline.INDEFINITE);
parallelTransition.play();

//End of animation

primaryStage.setTitle("Bird");
primaryStage.setScene(scene);
primaryStage.show();
}
}
```

Результати роботи програми

