



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: «Структура файлів формату .bmp. Анімація примітивів
за допомогою засобів бібліотеки JavaFX»**

Виконав
студент III курсу
каф. ПЗКС ФПМ
групи КП-93

Інюшев Артем
Владиславович

Перевірила

старший
викладач

Шкурат О. С.

Київ 2022

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.

Студентам пропонується скористатися розглянутими класами для читання, обробки

та збереження зображень формату .bmp з метою використання рисунку для створення

траєкторії руху або меж, в яких дозволений рух об'єктів. В даному випадку

рекомендується використовувати кольори великої контрастності для різних призначень

(наприклад, чорний колір відповідатиме за траєкторію руху, а інші кольори – заборонятимуть рух).

Варіант 9



Программный код:

```
package com.example.lab3;

import javafx.animation.*;
import javafx.application.Application;
import javafx.scene.*;
import javafx.scene.paint.Color;
import javafx.scene.shape.*;
import javafx.stage.Stage;
import javafx.util.Duration;

public class HelloApplication extends Application {
    public static void main(String args[]) {
        launch(args);
    }

    public void start(Stage primaryStage) {
        Group root = new Group();
        Scene scene = new Scene(root, 1200, 600);

        //head
        Circle head = new Circle(400, 150, 80);
        head.setFill(Color.YELLOW);
        root.getChildren().add(head);

        //eye
        Ellipse eye1 = new Ellipse(420, 130, 12, 8);
        eye1.setRotate(45);
        eye1.setFill(Color.BLACK);
        root.getChildren().add(eye1);

        Circle eye2 = new Circle(419, 127, 3);
        eye2.setFill(Color.WHITE);
        root.getChildren().add(eye2);

        //body
        Ellipse body = new Ellipse(400, 310, 150, 100);
        body.setFill(Color.YELLOW);
        root.getChildren().add(body);

        Ellipse body1 = new Ellipse(300, 280, 90, 65);
        body1.setRotate(70);
        body1.setFill(Color.rgb(255, 252, 0));
        root.getChildren().add(body1);

        Ellipse body2 = new Ellipse(300, 350, 70, 40);
        body2.setRotate(30);
        body2.setFill(Color.YELLOW);
        root.getChildren().add(body2);

        //beak
        Polygon beak = new Polygon();
        beak.getPoints().addAll(new Double[]{
            460.0, 190.0,
            475.0, 150.0,
            520.0, 151.0
        });
        beak.setFill(Color.ORANGE);
        root.getChildren().add(beak);

        Polygon beak1 = new Polygon();
        beak1.getPoints().addAll(new Double[]{
            460.0, 190.0,
            500.0, 190.0,
            500.0, 195.0,
            460.0, 195.0,
        });
        beak1.setFill(Color.ORANGE);
        root.getChildren().add(beak1);

        Polygon beak2 = new Polygon();
        beak2.getPoints().addAll(new Double[]{
            460.0, 190.0,
```

```

        495.0, 190.0,
        495.0, 165.0
    });
    beak2.setFill(Color.RED);
    root.getChildren().add(beak2);

    // 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.setResizable(false);
    primaryStage.setTitle("LAB 3");
    primaryStage.setScene(scene);
    primaryStage.show();
}
}

```

Результати



