

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»  
Факультет прикладної математики  
Кафедра програмного забезпечення комп'ютерних систем

**ЗВІТ**  
з лабораторної роботи № 3  
Варіант 1

Виконав:  
студент 3-го курсу,  
групи КП-83,  
спеціальності 121 –  
Інженерія  
програмного забезпечення  
Бойчук Владислав  
Андрійович

Київ – 2021

**Тема:**

Структура файлів формату .bmp. Анімація примітивів за допомогою засобів бібліотеки JavaFX

**Мета:**

- 1) вивчення структури та особливостей використання файлів формату .bmp; 2) вивчення стандартних засобів JavaFX для візуалізації зображення; 3) вивчення засобів анімації примітивів в JavaFX.

**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.

Варіант 1



Приклади роботи програми:



18

main.java

```
public class helloworld extends Application {  
    @Override
```

```

public void start(Stage primaryStage) throws IOException {
    Group root = new Group();
    Scene scene = new Scene(root, 600, 400);

    Path backLeaf = new Path(new MoveTo(200, 50), new QuadCurveTo(220, 20, 260, 40),
        new QuadCurveTo(290, 60, 320, 30), new QuadCurveTo(315, 80, 280, 100));

    LinearGradient backLeafGradient = new LinearGradient(0, 0, 0, 1, true, CycleMethod.NO_CYCLE,
        new Stop(0, Color.rgb(32, 160, 7)), new Stop(1, Color.rgb(175, 222, 8)));

    backLeaf.setStroke(Color.BLACK);
    backLeaf.setStrokeWidth(2);
    backLeaf.setFill(backLeafGradient);

    Path backLeafHalfLine = new Path(new MoveTo(220, 55), new QuadCurveTo(270, 90, 315, 40));

    backLeafHalfLine.setStroke(Color.BLACK);

    Path backLeafLine1 = new Path(new MoveTo(220, 55), new QuadCurveTo(240, 53, 239, 45));

    backLeafLine1.setStroke(Color.BLACK);

    Path backLeafLine2 = new Path(new MoveTo(238, 62), new QuadCurveTo(258, 60, 257, 52));

    backLeafLine2.setStroke(Color.BLACK);

    Path backLeafLine3 = new Path(new MoveTo(248, 67), new QuadCurveTo(268, 65, 267, 57));

    backLeafLine3.setStroke(Color.BLACK);

    Path backLeafLine4 = new Path(new MoveTo(271, 67), new QuadCurveTo(291, 65, 290, 53));

    backLeafLine4.setStroke(Color.BLACK);

    Path frontLeaf = new Path(new MoveTo(200, 50), new QuadCurveTo(215, 35, 235, 60),
        new QuadCurveTo(275, 90, 245, 140), new QuadCurveTo(225, 120, 205, 115),
        new QuadCurveTo(170, 110, 180, 70), new QuadCurveTo(180, 50, 200, 50));

    LinearGradient frontLeafGradient = new LinearGradient(0, 0, 1, 1, true, CycleMethod.NO_CYCLE,
        new Stop(0, Color.rgb(32, 160, 7)), new Stop(0.5, Color.rgb(175, 222, 8)),
        new Stop(1, Color.rgb(32, 160, 7)));

    frontLeaf.setStroke(Color.BLACK);
    frontLeaf.setStrokeWidth(2);
    frontLeaf.setFill(frontLeafGradient);

    Path frontLeafLine = new Path(new MoveTo(200, 50), new QuadCurveTo(260, 90, 245, 140));

    frontLeafLine.setStroke(Color.BLACK);

    Path branch = new Path(new MoveTo(200, 50), new QuadCurveTo(197, 47, 192, 40),
        new QuadCurveTo(188, 40, 182, 40), new QuadCurveTo(190, 47, 192, 52));

    branch.setStroke(Color.BLACK);
    branch.setFill(Color.rgb(78, 53, 40));

    Path branchSide = new Path(new MoveTo(192, 40), new QuadCurveTo(188, 35, 182, 40));

    branchSide.setStroke(Color.BLACK);
    branchSide.setFill(Color.rgb(161, 131, 55));

    Path appleBody = new Path(new MoveTo(250, 75), new QuadCurveTo(285, 95, 300, 130),
        new QuadCurveTo(320, 220, 230, 260), new QuadCurveTo(215, 250, 200, 260),
        new QuadCurveTo(135, 240, 130, 180), new QuadCurveTo(110, 110, 180, 80)

```

```

);

LinearGradient appleBodyGradient = new LinearGradient(0, 0, 1, 0, true, CycleMethod.NO_CYCLE,
    new Stop(0, Color.rgb(250, 51, 14)), new Stop(0.5, Color.rgb(254, 253, 86)),
    new Stop(1, Color.rgb(154, 220, 0)));

appleBody.setStroke(Color.BLACK);
appleBody.setStrokeWidth(3);
appleBody.setFill(appleBodyGradient);

Path leftEye = new Path(new MoveTo(200, 180), new QuadCurveTo(205, 190, 195, 200),
    new QuadCurveTo(185, 185, 200, 180)

);

leftEye.setStroke(Color.BLACK);
leftEye.setFill(Color.WHITE);

Path rightEye = new Path(new MoveTo(235, 180), new QuadCurveTo(240, 190, 230, 200),
    new QuadCurveTo(220, 185, 235, 180)

);

rightEye.setStroke(Color.BLACK);
rightEye.setFill(Color.WHITE);

Path leftDot = new Path(new MoveTo(198, 189), new QuadCurveTo(204, 192, 195, 195)

);

leftDot.setStroke(Color.BLACK);
leftDot.setFill(Color.BLACK);

Path rightDot = new Path(new MoveTo(230, 189), new QuadCurveTo(237, 192, 231, 195)

);

rightDot.setStroke(Color.BLACK);
rightDot.setFill(Color.BLACK);

Path leftBrew = new Path(new MoveTo(192, 155), new QuadCurveTo(192, 155, 200, 150)

);

leftBrew.setStroke(Color.BLACK);

Path rightBrew = new Path(new MoveTo(232, 150), new QuadCurveTo(232, 150, 240, 155)

);

rightBrew.setStroke(Color.BLACK);

Path mouth = new Path(new MoveTo(202, 230), new QuadCurveTo(222, 240, 228, 230)

);

mouth.setStroke(Color.BLACK);

Path nose = new Path(new MoveTo(212, 210), new QuadCurveTo(224, 220, 212, 225)

);

nose.setStroke(Color.BLACK);

```

```

Path podborodok = new Path(new MoveTo(212, 243), new QuadCurveTo(215, 245, 215, 250)

);

podborodok.setStroke(Color.BLACK);

Path leftArm = new Path(new MoveTo(130, 180), new QuadCurveTo(150, 190, 138, 210),
    new QuadCurveTo(118, 240, 160, 240)

);

leftArm.setStroke(Color.BLACK);
leftArm.setStrokeWidth(2);

Path leftHand = new Path(new MoveTo(160, 240), new QuadCurveTo(160, 240, 165, 225),
    new QuadCurveTo(165, 215, 170, 225), new QuadCurveTo(174, 215, 175, 228),
    new QuadCurveTo(180, 222, 180, 232), new QuadCurveTo(180, 232, 175, 240),
    new QuadCurveTo(190, 244, 177, 248), new QuadCurveTo(169, 253, 160, 240)

);

leftHand.setStroke(Color.BLACK);
leftHand.setStrokeWidth(2);
leftHand.setFill(Color.rgb(110, 195, 117));

Path rightArm = new Path(new MoveTo(300, 180), new QuadCurveTo(320, 190, 308, 210),
    new QuadCurveTo(288, 240, 330, 240)

);

rightArm.setStroke(Color.BLACK);
rightArm.setStrokeWidth(2);

Path rightHand = new Path(new MoveTo(330, 240), new QuadCurveTo(330, 240, 335, 225),
    new QuadCurveTo(335, 215, 340, 225), new QuadCurveTo(344, 215, 345, 228),
    new QuadCurveTo(350, 222, 350, 232), new QuadCurveTo(350, 232, 345, 240),
    new QuadCurveTo(360, 244, 347, 248), new QuadCurveTo(339, 253, 330, 240)

);

rightHand.setStroke(Color.BLACK);
rightHand.setStrokeWidth(2);
rightHand.setFill(Color.rgb(110, 195, 117));

Path touch1 = new Path(new MoveTo(275, 115), new QuadCurveTo(300, 155, 275, 205)

);

touch1.setStroke(Color.BLACK);
touch1.setStrokeWidth(2);

Path touch2 = new Path(new MoveTo(280, 115), new QuadCurveTo(295, 130, 295, 170)

);

touch2.setStroke(Color.BLACK);
touch2.setStrokeWidth(2);

Path touch3 = new Path(new MoveTo(160, 115), new QuadCurveTo(130, 155, 160, 205)

);

touch3.setStroke(Color.BLACK);

```

```

touch3.setStrokeWidth(2);

Path touch4 = new Path(new MoveTo(140, 135), new QuadCurveTo(130, 150, 140, 190)

);

touch4.setStroke(Color.BLACK);
touch4.setStrokeWidth(2);

////////////////////////////////////

root.getChildren().add(backLeaf);
root.getChildren().add(backLeafHalfLine);
root.getChildren().add(backLeafLine1);
root.getChildren().add(backLeafLine2);
root.getChildren().add(backLeafLine3);
root.getChildren().add(backLeafLine4);
root.getChildren().add(branch);
root.getChildren().add(branchSide);
root.getChildren().add(appleBody);
root.getChildren().add(frontLeaf);
root.getChildren().add(frontLeafLine);
root.getChildren().add(leftEye);
root.getChildren().add(rightEye);
root.getChildren().add(leftDot);
root.getChildren().add(rightDot);
root.getChildren().add(leftBrew);
root.getChildren().add(rightBrew);
root.getChildren().add(mouth);
root.getChildren().add(nose);
root.getChildren().add(podborodok);
root.getChildren().add(leftArm);
root.getChildren().add(leftHand);
root.getChildren().add(rightArm);
root.getChildren().add(rightHand);
root.getChildren().add(touch1);
root.getChildren().add(touch2);
root.getChildren().add(touch3);
root.getChildren().add(touch4);

//Animation

int cycleCount = 2;
int time = 2000;

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

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

```

```

        //переміщення по x
        TranslateTransition translateTransition = new
        TranslateTransition(Duration.millis(time), root);
        translateTransition.setFromX(350);
        translateTransition.setToX(0);
        translateTransition.setCycleCount(cycleCount+1);
        translateTransition.setAutoReverse(true);

        //масштабування у напрямку зменшення
        ScaleTransition scaleTransition2 = new ScaleTransition(Duration.millis(time),root);
        scaleTransition2.setToX(0.1); scaleTransition2.setToY(0.1);
        scaleTransition2.setCycleCount(cycleCount);
        scaleTransition2.setAutoReverse(true);

        //паралельна рисовка всіх анімацій
        ParallelTransition parallelTransition = new ParallelTransition();

        parallelTransition.getChildren().addAll(rotateTransition,scaleTransition,translateTransition,scaleTransition2)
    ;

    parallelTransition.setCycleCount(Timeline.INDEFINITE);
    parallelTransition.play();
    primaryStage.setResizable(false);
    primaryStage.setTitle("Lab3");
    primaryStage.setScene(scene);
    primaryStage.show();
}

```

```

private String returnPixelColor(int color) {
    String col = "BLACK";
    switch (color) {
        case 0:
            return "BLACK";    //BLACK;
        case 1:
            return "LIGHTCORAL"; //LIGHTCORAL;
        case 2:
            return "GREEN";    //GREEN
        case 3:
            return "BROWN";    //BROWN
        case 4:
            return "BLUE";    //BLUE;
        case 5:
            return "MAGENTA"; //MAGENTA;
        case 6:
            return "CYAN";    //CYAN;
        case 7:
            return "LIGHTGRAY"; //LIGHTGRAY;
        case 8:
            return "DARKGRAY"; //DARKGRAY;
        case 9:
            return "RED";    //RED;
        case 10:
            return "LIGHTGREEN"; //LIGHTGREEN
        case 11:
            return "YELLOW";    //YELLOW;
        case 12:
            return "LIGHTBLUE"; //LIGHTBLUE;
        case 13:
            return "LIGHTPINK"; //LIGHTMAGENTA
        case 14:
            return "LIGHTCYAN"; //LIGHTCYAN;
        case 15:
            return "WHITE";    //WHITE;
    }
}

```



```
        }  
        return col;  
    }  
}
```

### **Висновок:**

- 1) вивчив структуру та особливості використання файлів формату .bmp;
- 2) вивчив стандартні засоби JavaFX для візуалізації зображення;
- 3) вивчив засоби анімації примітивів в JavaFX.