



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

Виконав(ла)
студент(ка) III курсу
групи КП-83
Симонюк Володимир Павлович
варіант №18

Зарахована
“ ____ ” “ ____ ” 20__ р.
Шкурат Оксаною Сергіївною

Варіант завдання

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:



Лістинг коду програми

Main.java

```
public class Main extends Application {
    Color cBody = Color.rgb(15,85,190);
    Color cWheel = Color.rgb(255,0,0);
    Color cDark = Color.rgb(75,66,50);
    Color cMouth = Color.rgb(12,37,132);
    Color cWindow = Color.rgb(43,134,208);
    Color cLight = Color.rgb(255,255,255);

    int vw = 800;
    int vh = 800;

    int cx = vw / 2;
    int cy = vh / 2;

    int bodyHeight = 100;
    int bodyWidth = 200;

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

    @Override
    public void start(Stage primaryStage) {
        Group root = new Group();
        Scene scene = new Scene (root, vw, vh);

        DrawSmallWheel(root);
        DrawBody(root);
        DrawBigWheel(root);
        DrawHead(root);
        DrawChimney(root);
    }
}
```

```

        PlayAnimations(root);

        primaryStage.setScene(scene);
        primaryStage.show();
    }

    public void PlayAnimations(Group root) {
        int time = 3000;

        TranslateTransition translateFromRight = new
TranslateTransition(Duration.millis(time), root);
        translateFromRight.setFromX(vw);
        translateFromRight.setToX(0);

        RotateTransition rotate = new RotateTransition(Duration.millis(time), root);
        rotate.setByAngle(360f);
        rotate.setCycleCount(Timeline.INDEFINITE);

        ScaleTransition scaleDown = new ScaleTransition(Duration.millis(time), root);
        scaleDown.setToX(0);
        scaleDown.setToY(0);

        TranslateTransition translateToTopRight = new
TranslateTransition(Duration.millis(time), root);
        translateToTopRight.setToX(vw);
        translateToTopRight.setToY(-vh);

        SequentialTransition ST = new SequentialTransition(
            translateFromRight,
            new ParallelTransition(
                rotate,
                scaleDown,
                translateToTopRight
            )
        );

        ST.play();
    }

    public void DrawHead(Group root) {
        int headCX = cx - bodyWidth / 2;
        int headCY = cy;

        Circle trainHead = new Circle(headCX, headCY, bodyHeight / 2, cBody);
        Ellipse leftEye = new Ellipse(headCX - 20, headCY - 10, 10, 20);
        Ellipse leftEyePupil = new Ellipse(headCX - 25, headCY - 5, 5, 10);

        Ellipse rightEye = new Ellipse(headCX + 20, headCY - 10, 10, 20);
        Ellipse rightEyePupil = new Ellipse(headCX + 15, headCY - 5, 5, 10);

        Polygon mouth = new Polygon(
            headCX - 30, headCY + 20,
            headCX - 10, headCY + 30,
            headCX + 10, headCY + 30,
            headCX + 30, headCY + 20,
            headCX + 15, headCY + 35,
            headCX - 15, headCY + 35
        );

        leftEye.setFill(cLight);
        rightEye.setFill(cLight);
        leftEyePupil.setFill(cDark);
        rightEyePupil.setFill(cDark);
        mouth.setFill(cMouth);

        root.getChildren().add(trainHead);
    }

```

```

        root.getChildren().add(leftEye);
        root.getChildren().add(leftEyePupil);
        root.getChildren().add(rightEye);
        root.getChildren().add(rightEyePupil);
        root.getChildren().add(mouth);
    };

    public void DrawChimney(Group root) {
        int lcx = cx - bodyWidth / 2;
        int lcy = cy - bodyHeight / 2;

        Polygon chimney = new Polygon(
            lcx, lcy,
            lcx, lcy - 10,
            lcx - 15, lcy - 50,
            lcx - 25, lcy - 60,
            lcx - 30, lcy - 75,
            lcx + 80, lcy - 75,
            lcx + 75, lcy - 60,
            lcx + 65, lcy - 50,
            lcx + 50, lcy - 10,
            lcx + 50, lcy
        );

        Polygon chimneyLine1 = new Polygon(
            lcx - 15, lcy - 50,
            lcx - 25, lcy - 60,
            lcx + 75, lcy - 60,
            lcx + 65, lcy - 50
        );

        Polygon chimneyLine2 = new Polygon(
            lcx - 30, lcy - 75,
            lcx - 15, lcy - 80,
            lcx + 65, lcy - 80,
            lcx + 80, lcy - 75
        );

        chimney.setFill(cDark);
        chimneyLine1.setFill(cWheel);
        chimneyLine2.setFill(cMouth);

        root.getChildren().add(chimney);
        root.getChildren().add(chimneyLine1);
        root.getChildren().add(chimneyLine2);
    };

    public void DrawBigWheel(Group root) {
        Circle bigWheel = new Circle(cx + bodyWidth / 4, cy + bodyHeight / 6,
bodyHeight * 0.65, cWheel);
        Circle bigWheelAxis = new Circle(cx + bodyWidth / 4, cy + bodyHeight / 6, 5,
cDark);

        root.getChildren().add(bigWheel);
        root.getChildren().add(bigWheelAxis);
    };

    public void DrawSmallWheel(Group root) {
        Circle smallWheel = new Circle(cx - bodyWidth / 2, cy + bodyHeight / 2, 30,
cWheel);
        Circle smallWheelAxis = new Circle(cx - bodyWidth / 2, cy + bodyHeight / 2,
5, cDark);

        root.getChildren().add(smallWheel);
        root.getChildren().add(smallWheelAxis);
    };

```

```

    public void DrawBody(Group root) {
        Rectangle main = new Rectangle(cx - bodyWidth / 2, cy - bodyHeight / 2,
bodyWidth, bodyHeight);
        Rectangle cabin = new Rectangle(cx , cy - bodyHeight, bodyWidth / 2,
bodyHeight / 2);
        Rectangle cabinRoof = new Rectangle(cx - 20, cy - bodyHeight - 10, bodyWidth
/ 2 + 20, 10);
        main.setFill(cBody);
        cabin.setFill(cBody);
        cabinRoof.setFill(cBody);

        Rectangle window = new Rectangle(cx + 10, cy - bodyHeight, bodyWidth / 2 -
20, bodyHeight / 2);
        window.setFill(cWindow);

        Circle trainHeadRightBorder = new Circle(cx - bodyWidth / 2 + 10, cy,
bodyHeight / 2, cDark);

        root.getChildren().add(main);
        root.getChildren().add(cabin);
        root.getChildren().add(cabinRoof);
        root.getChildren().add(window);
        root.getChildren().add(trainHeadRightBorder);
    };
}

```

Результат

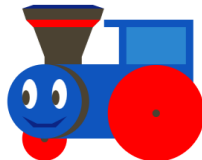


Рис. 1-2. Результаты работы программы