

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“

|  |  |  |  |
| --- | --- | --- | --- |
| Виконала  студентка III курсу  групи КП-83  Дзьоба Уляна Олегівна  (*прізвище, ім’я, по батькові*)  варіант № 5 |  |  | Зарахована  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладачем  Шкурат Оксаною Сергіївною (*прізвище, ім’я, по батькові*) |

Київ 2021

**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**

**package** lab3;

**import** javafx.animation.\*;

**import** javafx.application.Application;

**import** javafx.scene.Group;

**import** javafx.scene.Scene;

**import** javafx.scene.paint.Color;

**import** javafx.scene.shape.\*;

**import** javafx.scene.transform.Rotate;

**import** javafx.scene.transform.Scale;

**import** javafx.scene.transform.Translate;

**import** javafx.stage.Stage;

**import** javafx.util.Duration;

**public** **class** Main **extends** Application {

**public** **static** **void** main (String args[]) {

*launch*(args);

}

@Override

**public** **void** start(Stage primaryStage) {

Group root = **new** Group();

Scene scene = **new** Scene (root, 470, 310);

Ellipse ellipse = **new** Ellipse(290,245,70, 23);

ellipse.setFill(Color.*rgb*(236, 236, 236));

Rotate rotate = **new** Rotate();

rotate.setPivotX(210); //Pivot X Top-Left corner

rotate.setPivotY(245); //Pivot Y

rotate.setAngle(-25); //Angle degrees

//Add the transform to the node

ellipse.getTransforms().add(rotate);

root.getChildren().add(ellipse);

Circle w = **new** Circle(232, 195, 50);

w.setFill(Color.***WHITE***);

root.getChildren().add(w);

//outline

//head

Arc arc11 = **new** Arc(220, 98, 39, 37, 180, 82);

arc11.setType(ArcType.***OPEN***);

arc11.setStroke(Color.*rgb*(171, 208, 211));

arc11.setStrokeLineCap(StrokeLineCap.***ROUND***);

arc11.setStrokeWidth(7);

arc11.setFill(Color.***TRANSPARENT***);

arc11.setSmooth(**true**);

root.getChildren().add(arc11);

Arc arc111 = **new** Arc(220, 98, 40, 38, 180, 81);

arc111.setType(ArcType.***OPEN***);

arc111.setStroke(Color.***BLACK***);

arc111.setStrokeWidth(2);

arc111.setFill(Color.***TRANSPARENT***);

arc111.setSmooth(**true**);

root.getChildren().add(arc111);

Arc arc12 = **new** Arc(220, 100, 36, 38, -40, 100);

arc12.setType(ArcType.***OPEN***);

arc12.setStroke(Color.*rgb*(171, 208, 211));

arc12.setStrokeWidth(4);

arc12.setFill(Color.***TRANSPARENT***);

arc12.setSmooth(**true**);

root.getChildren().add(arc12);

Arc arc1 = **new** Arc(220, 100, 40, 40, -39, 250);

arc1.setType(ArcType.***OPEN***);

arc1.setStroke(Color.***BLACK***);

arc1.setStrokeWidth(5);

arc1.setFill(Color.***TRANSPARENT***);

arc1.setSmooth(**true**);

root.getChildren().add(arc1);

Arc arc41 = **new** Arc(231, 194, 48, 46, -188, 240);

arc41.setType(ArcType.***OPEN***);

arc41.setStroke(Color.*rgb*(171, 208, 211));

arc41.setStrokeWidth(5);

arc41.setFill(Color.***TRANSPARENT***);

arc41.setSmooth(**true**);

root.getChildren().add(arc41);

//middle

Arc arc21 = **new** Arc(215, 153, 43, 41, 175, 50);

arc21.setType(ArcType.***OPEN***);

arc21.setStroke(Color.*rgb*(171, 208, 211));

arc21.setStrokeWidth(5);

arc21.setFill(Color.***TRANSPARENT***);

arc21.setSmooth(**true**);

root.getChildren().add(arc21);

Arc arc2 = **new** Arc(215, 154, 45, 45, 135, 88);

arc2.setType(ArcType.***OPEN***);

arc2.setStroke(Color.***BLACK***);

arc2.setStrokeWidth(5);

arc2.setFill(Color.***TRANSPARENT***);

arc2.setSmooth(**true**);

root.getChildren().add(arc2);

Arc arc31 = **new** Arc(213, 152, 45, 43, -40, 80);

arc31.setType(ArcType.***OPEN***);

arc31.setStroke(Color.*rgb*(171, 208, 211));

arc31.setStrokeLineCap(StrokeLineCap.***ROUND***);

arc31.setStrokeWidth(5);

arc31.setFill(Color.***TRANSPARENT***);

arc31.setSmooth(**true**);

root.getChildren().add(arc31);

Arc arc32 = **new** Arc(216, 152, 45, 43, -40, 35);

arc32.setType(ArcType.***OPEN***);

arc32.setStroke(Color.***BLACK***);

arc32.setStrokeWidth(2);

arc32.setFill(Color.***TRANSPARENT***);

arc32.setSmooth(**true**);

root.getChildren().add(arc32);

Arc arc3 = **new** Arc(217, 152, 45, 43, -3, 42);

arc3.setType(ArcType.***OPEN***);

arc3.setStroke(Color.***BLACK***);

arc3.setStrokeWidth(5);

arc3.setFill(Color.***TRANSPARENT***);

arc3.setSmooth(**true**);

root.getChildren().add(arc3);

//buttons

Circle btn1 = **new** Circle(242, 140, 5);

Circle btn2 = **new** Circle(247, 156, 5);

Circle btn3 = **new** Circle(240, 171, 5);

root.getChildren().add(btn1);

root.getChildren().add(btn2);

root.getChildren().add(btn3);

//bottom

Arc arc4 = **new** Arc(232, 195, 50, 50, -188, 240);

arc4.setType(ArcType.***OPEN***);

arc4.setStroke(Color.***BLACK***);

arc4.setStrokeWidth(5);

arc4.setFill(Color.***TRANSPARENT***);

arc4.setSmooth(**true**);

root.getChildren().add(arc4);

//face

Circle eye1 = **new** Circle(225, 90, 9);

Circle eye2 = **new** Circle(225, 92, 5);

Circle eye3 = **new** Circle(226, 89, 3);

Circle eye4 = **new** Circle(238, 91, 6);

Circle eye5 = **new** Circle(238, 93, 3);

Circle eye6 = **new** Circle(239, 92, 2);

eye2.setFill(Color.***WHITE***);

eye5.setFill(Color.***WHITE***);

root.getChildren().add(eye1);

root.getChildren().add(eye2);

root.getChildren().add(eye3);

root.getChildren().add(eye4);

root.getChildren().add(eye5);

root.getChildren().add(eye6);

Polygon triangle = **new** Polygon(232, 104, 233, 101, 235, 99, 267, 99, 271, 102, 267, 104, 235, 109, 233, 107);

triangle.setStroke(Color.***BLACK***);

triangle.setStrokeWidth(2);

triangle.setFill(Color.*rgb*(219, 124, 44));

root.getChildren().add(triangle);

Arc t = **new** Arc(263, 102, 9, 5, -90, 180);

t.setType(ArcType.***OPEN***);

t.setStroke(Color.***BLACK***);

t.setStrokeWidth(3);

t.setFill(Color.***TRANSPARENT***);

t.setSmooth(**true**);

Rotate rotate1 = **new** Rotate();

rotate1.setPivotX(263); //Pivot X

rotate1.setPivotY(102); //Pivot Y

rotate1.setAngle(-7); //Angle degrees

//Add the transform to the node

t.getTransforms().add(rotate1);

root.getChildren().add(t);

//mouth

Arc m1 = **new** Arc(230, 90, 25, 30, 209, 90);

m1.setType(ArcType.***OPEN***);

m1.setStroke(Color.***BLACK***);

m1.setStrokeWidth(2);

m1.setFill(Color.***TRANSPARENT***);

m1.setSmooth(**true**);

root.getChildren().add(m1);

Arc m2 = **new** Arc(220, 109, 12, 14, 160, 180);

m2.setType(ArcType.***OPEN***);

m2.setSmooth(**true**);

root.getChildren().add(m2);

Arc m11 = **new** Arc(231, 89, 25, 30, 210, 90);

m11.setFill(Color.***WHITE***);

m11.setSmooth(**true**);

root.getChildren().add(m11);

Arc m3 = **new** Arc(206, 100, 4, 5, 220, 110);

m3.setType(ArcType.***OPEN***);

m3.setStroke(Color.***BLACK***);

m3.setStrokeWidth(1);

m3.setFill(Color.***TRANSPARENT***);

m3.setSmooth(**true**);

root.getChildren().add(m3);

**double**[] p = {170, 145, 135, 127,

110, 135, 107, 133, 106, 130, 108, 126,

123, 121, 108, 117, 106, 114, 107, 107,

126, 115,

116, 101, 115, 99, 118, 96, 121, 95, 122, 95,

133, 114, 136, 102, 143, 102, 145, 105,

143, 108,

142, 120, 175, 135,

186, 141, 187, 142, 189, 145, 185, 150, 180, 152};

//sticks

Polygon st1 = **new** Polygon(p);

st1.setStroke(Color.***BLACK***);

st1.setStrokeLineJoin(StrokeLineJoin.***ROUND***);

st1.setStrokeWidth(4);

st1.setFill(Color.*rgb*(114, 78, 43));

root.getChildren().add(st1);

**double** s[]={170, 145, 135, 127,

110, 135, 107, 133, 106, 130, 108, 126,

123, 121, 108, 117, 106, 114, 107, 107,

126, 115,

116, 101, 115, 99, 118, 96, 121, 95, 122, 95,

133, 114, 136, 102, 143, 102, 145, 105,

143, 108,

142, 120, 175, 135};

Polyline line = **new** Polyline(s);

line.setStroke(Color.***BLACK***);

line.setStrokeLineJoin(StrokeLineJoin.***ROUND***);

line.setStrokeWidth(5);

root.getChildren().add(line);

Polygon st2 = **new** Polygon(s);

st2.setStroke(Color.***BLACK***);

st2.setStrokeLineJoin(StrokeLineJoin.***ROUND***);

st2.setStrokeWidth(5);

st2.setFill(Color.*rgb*(114, 78, 43));

st2.setScaleX(-1);

Scale scale = **new** Scale(0.85, 0.85);

st2.getTransforms().add(scale);

Translate translate = **new** Translate(-148, 25);

st2.getTransforms().add(translate);

root.getChildren().add(st2);

// Animation

**int** time = 1500;

Path path = **new** Path();

MoveTo moveTo = **new** MoveTo(230, 150);

ArcTo arcTo = **new** ArcTo();

arcTo.setX(150);

arcTo.setY(180);

arcTo.setRadiusX(100);

arcTo.setRadiusY(100);

arcTo.setSweepFlag(**false**);

arcTo.setLargeArcFlag(**false**);

Path path2 = **new** Path();

MoveTo moveTo2 = **new** MoveTo(230, 150);

ArcTo arcTo2 = **new** ArcTo();

arcTo2.setX(310);

arcTo2.setY(180);

arcTo2.setRadiusX(100);

arcTo2.setRadiusY(100);

arcTo2.setSweepFlag(**true**);

arcTo2.setLargeArcFlag(**false**);

path.getElements().add(moveTo);

path.getElements().add(arcTo);

path2.getElements().add(moveTo2);

path2.getElements().add(arcTo2);

PathTransition pathTransition = **new** PathTransition();

pathTransition.setDuration(Duration.*millis*(time));

pathTransition.setPath(path);

pathTransition.setNode(root);

pathTransition.setCycleCount(2);

pathTransition.setAutoReverse(**true**);

PathTransition pathTransition2 = **new** PathTransition();

pathTransition2.setDuration(Duration.*millis*(time));

pathTransition2.setPath(path2);

pathTransition2.setNode(root);

pathTransition2.setCycleCount(2);

pathTransition2.setAutoReverse(**true**);

ScaleTransition scaleTransition = **new** ScaleTransition(Duration.*millis*(time), root);

scaleTransition.setToX(0.5);

scaleTransition.setToY(0.5);

scaleTransition.setAutoReverse(**true**);

RotateTransition rotateTransition = **new** RotateTransition(Duration.*millis*(time), root);

rotateTransition.setByAngle(-50);

rotateTransition.setCycleCount(2);

rotateTransition.setAutoReverse(**true**);

RotateTransition rotateTransition2 = **new** RotateTransition(Duration.*millis*(time), root);

rotateTransition2.setByAngle(50);

rotateTransition2.setCycleCount(2);

rotateTransition2.setAutoReverse(**true**);

TranslateTransition translateTransition = **new** TranslateTransition(Duration.*millis*(time), root);

translateTransition.setFromY(0);

translateTransition.setToY(100);

translateTransition.setAutoReverse(**true**);

ParallelTransition parallelTransition = **new** ParallelTransition();

parallelTransition.getChildren().addAll(

rotateTransition,

pathTransition

);

ParallelTransition parallelTransition2 = **new** ParallelTransition();

parallelTransition2.getChildren().addAll(

rotateTransition2,

pathTransition2

);

ParallelTransition parallelTransition3 = **new** ParallelTransition();

parallelTransition3.getChildren().addAll(

scaleTransition,

translateTransition

);

SequentialTransition sTransition = **new** SequentialTransition();

sTransition.getChildren().addAll(

parallelTransition,

parallelTransition2,

parallelTransition3

);

sTransition.setCycleCount(Timeline.***INDEFINITE***);

sTransition.play();

primaryStage.setTitle("Snowman");

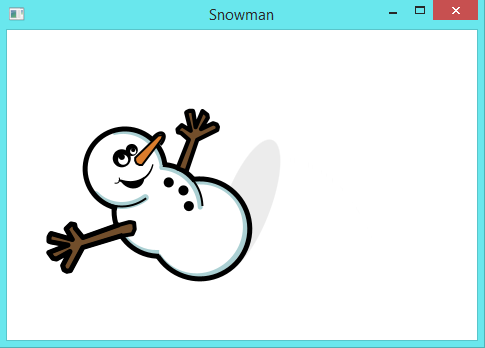
primaryStage.setScene(scene);

primaryStage.show();

}

}

**Результат**

****