

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 № 6

Виконав:
студент 3-го курсу,
групи КП-82,
спеціальності 121 –
Інженерія
програмного
забезпечення
Клапатюк Олександр
Петрович

Київ – 2020

Тема: Анімація тривимірних об'єктів

Мета: Навчитися анімувати складні об'єкти тривимірної сцени.

PokemonTrainer.java

```
package lab6;
import javax.vecmath.*;

import com.sun.j3d.utils.image.TextureLoader;
import com.sun.j3d.utils.universe.*;
import javax.media.j3d.*;
import com.sun.j3d.utils.behaviors.vp.*;
import javax.swing.JFrame;
import com.sun.j3d.loaders.*;
import com.sun.j3d.loaders.objectfile.*;
import java.util.Hashtable;
import java.util.Enumeration;

public class PokemonTrainer extends JFrame{
    public Canvas3D myCanvas3D;

    public PokemonTrainer(){
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        myCanvas3D = new Canvas3D(SimpleUniverse.getPreferredConfiguration());
        SimpleUniverse simpUniv = new SimpleUniverse(myCanvas3D);

        simpUniv.getViewingPlatform().setNominalViewingTransform();

        createSceneGraph(simpUniv);
        addLight(simpUniv);

        // навігація с допомогою миши
        OrbitBehavior ob = new OrbitBehavior(myCanvas3D);
        ob.setSchedulingBounds(new BoundingSphere(new Point3d(0.0,0.0,0.0),Double.MAX_VALUE));
        simpUniv.getViewingPlatform().setViewPlatformBehavior(ob);

        setTitle("Pokemon Trainer");
        setSize(700,700);
        getContentPane().add("Center", myCanvas3D);
        setVisible(true);
    }

    public void createSceneGraph(SimpleUniverse su){
        ObjectFile f = new ObjectFile(ObjectFile.RESIZE);
        BoundingSphere bs = new BoundingSphere(new Point3d(0.0,0.0,0.0),Double.MAX_VALUE);
        String name;
        BranchGroup trainerBranchGroup = new BranchGroup();
        TextureLoader t = new TextureLoader("source_folder//bf.jpg", myCanvas3D);
        Background trainerBackground = new Background(t.getImage());

        Scene trainerScene = null;
        try{
            trainerScene = f.load("source_folder//pokemon_trainer.obj");
        }
        catch (Exception e){
            System.out.println("File loading failed:" + e);
        }
        Hashtable roachNamedObjects = trainerScene.getNamedObjects();
        Enumeration enumerator = roachNamedObjects.keys();
        while (enumerator.hasMoreElements()){
            name = (String) enumerator.nextElement();
            System.out.println("Name: " + name);
        }

        // start animation
        Transform3D startTransformation = new Transform3D();
        startTransformation.setScale(2.0/6);
        Transform3D combinedStartTransformation = new Transform3D();
        combinedStartTransformation.mul(startTransformation);

        TransformGroup scratStartTransformGroup = new TransformGroup(combinedStartTransformation);

        int movesCount = 100; // moves count
    }
}
```

```

int movesDuration = 500; // moves for 0,3 seconds
int startTime = 0; // launch animation after timeStart seconds

// голова

Appearance headApp = new Appearance(); setToMyDefaultAppearance(headApp, new
Color3f(0.2f, 0.2f, 0.5f));

Alpha headRotAlpha = new Alpha(movesCount, Alpha.INCREASING_ENABLE,
startTime, 0, movesDuration,0,0,0,0,0);

Shape3D head = (Shape3D) roachNamedObjects.get("polygon1");
head.setAppearance(headApp); TransformGroup headTG = new TransformGroup();
headTG.addChild(head.cloneTree());

Transform3D headRotAxis = new Transform3D(); headRotAxis.set(new
Vector3d(0.0, 0.0, 0.0));

RotationInterpolator headRot = new RotationInterpolator(headRotAlpha, headTG,
headRotAxis, 0.0f, (float) Math.PI*2); headRot.setSchedulingBounds(bs);
headTG.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
headTG.addChild(headRot);

//мяч
Appearance ballApp = new Appearance();
setToMyDefaultAppearance(ballApp, new Color3f(0.9f, 0.0f, 0.0f));

Alpha ballAlpha = new Alpha(movesCount, Alpha.INCREASING_ENABLE, startTime, 0, movesDuration,0,0,0,0,0);

Shape3D ball = (Shape3D) roachNamedObjects.get("ball2");
ball.setAppearance(ballApp);
TransformGroup ballTG = new TransformGroup();
ballTG.addChild(ball.cloneTree());

Transform3D ballRotAxis = new Transform3D();

RotationInterpolator ballrot = new RotationInterpolator(ballAlpha, ballTG, ballRotAxis, 0.0f, (float)
-Math.PI); // Math.PI*2
ballrot.setSchedulingBounds(bs);
ballTG.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
ballTG.addChild(ballrot);

// тіло
Appearance bodyApp = new Appearance();
setToMyDefaultAppearance(bodyApp, new Color3f(0.2f, 0.3f, 0.2f));

TransformGroup sceneGroup = new TransformGroup();
sceneGroup.addChild(headTG);
sceneGroup.addChild(ballTG);

TransformGroup tgBody = new TransformGroup();
Shape3D nShape = (Shape3D) roachNamedObjects.get("polygon0");
nShape.setAppearance(bodyApp);
tgBody.addChild(nShape.cloneTree());
sceneGroup.addChild(tgBody.cloneTree());

//обличчя шлем
Appearance faceApp = new Appearance();
setToMyDefaultAppearance(faceApp, new Color3f(0.3f, 0.3f, 0.2f));

TransformGroup tgFace1 = new TransformGroup();
Shape3D face1Shape = (Shape3D) roachNamedObjects.get("polygon3");
face1Shape.setAppearance(faceApp);
tgFace1.addChild(face1Shape.cloneTree());
sceneGroup.addChild(tgFace1.cloneTree());

TransformGroup tgFace2 = new TransformGroup();
Shape3D face2Shape = (Shape3D) roachNamedObjects.get("polygon4");
face2Shape.setAppearance(faceApp);
tgFace2.addChild(face2Shape.cloneTree());
sceneGroup.addChild(tgFace2.cloneTree());

TransformGroup tgFace = new TransformGroup();
Shape3D faceShape = (Shape3D) roachNamedObjects.get("polygon2");
faceShape.setAppearance(faceApp);
tgFace.addChild(faceShape.cloneTree());
sceneGroup.addChild(tgFace.cloneTree());

//сумка
Appearance bagApp = new Appearance();
setToMyDefaultAppearance(bagApp, new Color3f(0.0f, 0.6f, 0.0f));

```

```

TransformGroup tgBag1 = new TransformGroup();
Shape3D bagShape1 = (Shape3D) roachNamedObjects.get("polygon6");
bagShape1.setAppearance(bagApp);
tgBag1.addChild(bagShape1.cloneTree());
sceneGroup.addChild(tgBag1.cloneTree());

TransformGroup tgBag = new TransformGroup();
Shape3D bagShape = (Shape3D) roachNamedObjects.get("polygon7");
bagShape.setAppearance(bagApp);
tgBag.addChild(bagShape.cloneTree());
sceneGroup.addChild(tgBag.cloneTree());

TransformGroup whiteTransXformGroup = translate(
    scratStartTransformGroup,
    new Vector3f(0.0f, 0.0f, -0.5f));

TransformGroup whiteRotXformGroup = rotate(whiteTransXformGroup, new Alpha(10, 5000));
trainerBranchGroup.addChild(whiteRotXformGroup);
scratStartTransformGroup.addChild(sceneGroup);

BoundingSphere bounds = new BoundingSphere(new Point3d(120.0, 250.0, 100.0), Double.MAX_VALUE);
trainerBackground.setApplicationBounds(bounds);
trainerBranchGroup.addChild(trainerBackground);

trainerBranchGroup.compile();
su.addBranchGraph(trainerBranchGroup);
}

// додаємо світло
public void addLight(SimpleUniverse su){
    BranchGroup bgLight = new BranchGroup();
    BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);
    Color3f lightColour1 = new Color3f(1.0f, 1.0f, 1.0f);
    Vector3f lightDir1 = new Vector3f(-1.0f, 0.0f, -0.5f);
    DirectionalLight light1 = new DirectionalLight(lightColour1, lightDir1);
    light1.setInfluencingBounds(bounds);
    bgLight.addChild(light1);
    su.addBranchGraph(bgLight);
}

private TransformGroup translate(Node node, Vector3f vector){

    Transform3D transform3D = new Transform3D();
    transform3D.setTranslation(vector);
    TransformGroup transformGroup =
        new TransformGroup();
    transformGroup.setTransform(transform3D);

    transformGroup.addChild(node);
    return transformGroup;
}

private TransformGroup rotate(Node node, Alpha alpha){
    TransformGroup xformGroup = new TransformGroup();
    xformGroup.setCapability(
        TransformGroup.ALLOW_TRANSFORM_WRITE);

    RotationInterpolator interpolator =
        new RotationInterpolator(alpha, xformGroup);

    interpolator.setSchedulingBounds(new BoundingSphere(
        new Point3d(0.0, 0.0, 0.0), 1.0));

    xformGroup.addChild(interpolator);
    xformGroup.addChild(node);

    return xformGroup;
}

public static void setToMyDefaultAppearance(Appearance app, Color3f col) {
    app.setMaterial(new Material(col, col, col, col, 150.0f));
}

public static void main(String[] args) {
    PokemonTrainer start = new PokemonTrainer();
}
}

```

Приклади роботи:



