



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 № 5**

з дисципліни “Імпорт тривімирних моделей у середовище програмування  
java3D, обробка та маніпуляція цих зображень”

Виконав  
студент III курсу  
групи КП-82

Жиров Даниїл  
*(прізвище, ім'я, по батькові)*

варіант № 6

Зарахована  
“ \_\_\_\_ ” “ \_\_\_\_ ” 20\_\_ р.  
викладачем

Шкурат Оксаною Сергіївною  
*(прізвище, ім'я, по батькові)*

## **Варіант завдання**

### **Завдання:**

Імпортувати моделі тривимірних об'єктів форматів, що визначені варіантом. Створити реалістичну анімацію об'єкту. Додати до сцени фон, інші об'єкти для надання сцені реалістичного вигляду. Для цього використати текстури, матеріали, імпортувати додаткові об'єкти з відкритих бібліотек, за бажанням створити прості об'єкти у графічному редакторі.

Студенти, які мають непарний номер варіанту у списку групи імпортують моделі формату .obj, парний варіант – .lwo.

**Варіант:** Модель lwo.

## Лістинг коду програми

```
package org.example;

import com.sun.j3d.loaders.Scene;
import com.sun.j3d.loaders.lw3d.Lw3dLoader;
import com.sun.j3d.loaders.objectfile.ObjectFile;
import com.sun.j3d.utils.image.TextureLoader;
import com.sun.j3d.utils.universe.*;

import javax.media.j3d.*;
import javax.swing.*;
import javax.vecmath.*;
import java.awt.*;
import java.io.FileReader;
import java.io.IOException;
import java.util.Map;

public class App extends JFrame {
    static SimpleUniverse universe;
    static Scene staircase1, staircase2, ball;
    static BranchGroup root;
    static Canvas3D canvas;

    static TransformGroup wholeBall;
    static Transform3D transform3D;

    public App() throws IOException {
        configureWindow();
        configureCanvas();
        configureUniverse();
        addStairsToUniverse();
        addBallToUniverse();
        addAppearanceToObjects();
        addColorBackground();
        addLightToUniverse();
        ChangeViewAngle();
        root.compile();
        universe.addBranchGraph(root);
    }

    private void configureWindow() {
        setTitle("Animation");
        setSize(760,680);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    private void configureCanvas(){
        canvas = new Canvas3D(SimpleUniverse.getPreferredConfiguration());
        canvas.setDoubleBufferEnable(true);
        getContentPane().add(canvas, BorderLayout.CENTER);
    }

    private void configureUniverse(){
        root = new BranchGroup();
        universe = new SimpleUniverse(canvas);
        universe.getViewingPlatform().setNominalViewingTransform();
    }

    private void addStairsToUniverse() throws IOException{
        staircase1 = getSceneFromFile("/Users/daniilzhyrov/Documents/учеба/компьютерная графика/
lab5/src/main/resources/staircase.obj");
        staircase2 = getSceneFromFile("/Users/daniilzhyrov/Documents/учеба/компьютерная графика/
lab5/src/main/resources/staircase.obj");
        BranchGroup group = new BranchGroup();
        TransformGroup tg = new TransformGroup();
        Transform3D transform3D = new Transform3D();
        transform3D.setTranslation(new Vector3f(-1.2f, .4f, 0));
        tg.setTransform(transform3D);
        tg.addChild(staircase1.getSceneGroup());
        group.addChild(tg);
        group.addChild(staircase2.getSceneGroup());
    }
}
```

```

        root = group;
    }

    public void addBallToUniverse() throws IOException {
        ball = getSceneFromFile("/Users/daniilzhyrov/Documents/учеба/компьютерная графика/lab5/src/main/resources/ball.obj");
        wholeBall = new TransformGroup();
        wholeBall.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
        transform3D = new Transform3D();
        transform3D.setTranslation(new Vector3f(-2.3f, .7f, 0));
        transform3D.setScale(0.1);
        wholeBall.setTransform(transform3D);
        wholeBall.addChild(ball.getSceneGroup());
        root.addChild(wholeBall);
    }

    private void printModelElementsList(Map<String, Shape3D> nameMap){
        for (String name : nameMap.keySet()) {
            System.out.printf("Name: %s\n", name);
        }
    }

    Texture getTexture(String path) {
        TextureLoader textureLoader = new TextureLoader(path, "RGB", canvas);
        Texture texture = textureLoader.getTexture();
        texture.setBoundaryModeS(Texture.WRAP);
        texture.setBoundaryModeT(Texture.WRAP);
        texture.setBoundaryColor( new Color4f( 0.0f, 1.0f, 0.0f, 0.0f ) );
        return texture;
    }

    Material getMaterial() {
        Material material = new Material();
        material.setAmbientColor (new Color3f(.15f, .15f, .15f));
        material.setDiffuseColor (new Color3f(1f, 1f, 1f));
        material.setSpecularColor(new Color3f(0f, 0f, 0f));
        material.setEmissiveColor(new Color3f(0f, 0f, 0f));
        material.setShininess(0.3f);
        // material.setLightingEnable(true);
        return material;
    }

    private void addAppearance(Shape3D object, String texturePath){
        Appearance objectAppearance = new Appearance();
        objectAppearance.setTexture(getTexture(texturePath));
        TextureAttributes texAttr = new TextureAttributes();
        texAttr.setTextureMode(TextureAttributes.MODULATE);
        texAttr.setPerspectiveCorrectionMode(TextureAttributes.NICEST);
        objectAppearance.setTextureAttributes(texAttr);
        objectAppearance.setMaterial(getMaterial());
        object.setAppearance(objectAppearance);
    }

    private void addAppearanceToObjects() {
        addAppearance((Shape3D) staircase1.getNamedObjects().get("stairsteps_layer1"), "/Users/daniilzhyrov/Documents/учеба/компьютерная графика/lab5/src/main/resources/texture.png");
        addAppearance((Shape3D) staircase2.getNamedObjects().get("stairsteps_layer1"), "/Users/daniilzhyrov/Documents/учеба/компьютерная графика/lab5/src/main/resources/texture.png");
        addAppearance((Shape3D) ball.getNamedObjects().get("mesh_soccer_ball"), "/Users/daniilzhyrov/Documents/учеба/компьютерная графика/lab5/src/main/resources/ball.png");
    }

    private void addColorBackground(){
        Background background = new Background(new Color3f(Color.PINK));
        BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);
        background.setApplicationBounds(bounds);
        root.addChild(background);
    }

    private void addImageBackground(){
        TextureLoader t = new TextureLoader(getClass().getClassLoader().getResource("ball"), canvas);
    }

```

```

        Background background = new Background(t.getImage());
        background.setImageScaleMode(Background.SCALE_FIT_ALL);
        BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);
        background.setApplicationBounds(bounds);
        root.addChild(background);
    }

    private void ChangeViewAngle(){
        ViewingPlatform vp = universe.getViewingPlatform();
        TransformGroup vpGroup = vp.getMultiTransformGroup().getTransformGroup(0);
        Transform3D vpTranslation = new Transform3D();
        // Vector3f translationVector = new Vector3f(0.0F, -1.2F, 6F);
        Vector3f translationVector = new Vector3f(-.6F, .8F, 3.8F);
        // Vector3f translationVector = new Vector3f(-1F, .5F, 3.8F);
        vpTranslation.setTranslation(translationVector);
        vpGroup.setTransform(vpTranslation);
    }

    private void addLightToUniverse() {
        Color3f directionalLightColor = new Color3f(Color.WHITE);
        Color3f ambientLightColor = new Color3f(Color.WHITE);
        Vector3f lightDirection = new Vector3f(-3F, -3F, -3F);

        AmbientLight ambientLight = new AmbientLight(ambientLightColor);
        DirectionalLight directionalLight = new DirectionalLight(directionalLightColor,
lightDirection);

        Bounds influenceRegion = new BoundingSphere();
        Transform3D transform3D = new Transform3D();
        transform3D.setScale(8);
        influenceRegion.transform(transform3D);

        ambientLight.setInfluencingBounds(influenceRegion);
        directionalLight.setInfluencingBounds(influenceRegion);
        root.addChild(ambientLight);
        root.addChild(directionalLight);
    }

    public static Scene getSceneFromFile(String location) throws IOException {
        ObjectFile file = new ObjectFile(ObjectFile.RESIZE);
        file.setFlags (ObjectFile.RESIZE | ObjectFile.TRIANGULATE | ObjectFile.STRIPIFY);
        return file.load(new FileReader(location));
    }

    //Not always works
    public static Scene getSceneFromLwoFile(String location) throws IOException {
        Lw3dLoader loader = new Lw3dLoader();
        return loader.load(location);
    }

    public static void main(String[] args){
        try {
            App window = new App();
            new BallAnimation(wholeBall, transform3D, window);
            window.setVisible(true);
        }
        catch (IOException ex) {
            System.out.println(ex.getMessage());
        }
    }
}

```

## Скріншот роботи програми

