

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 №** **4**

з дисципліни “Математичні та алгоритмічні основи комп’ютерної графіки”

|  |  |  |  |
| --- | --- | --- | --- |
| Виконав  студент III курсу  групи КП-83  Клименко Ярослав  (*прізвище, ім’я, по батькові*)  варіант № 7 |  |  | Зарахована  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.  викладачем  Шкурат Оксаною Сергіївною (*прізвище, ім’я, по батькові*) |

Київ 2020

**Варіант завдання**

**Завдання**:

За допомогою засобів, що надає бібліотека Java3D, побудувати тривимірний об’єкт. Для цього скористатися основними примітивами, що буде доцільно використовувати згідно варіанту: сфера, конус, паралелепіпед, циліндр. Об'єкт має складатися з 5-15 примітивів. Задати матеріал кожного примітиву, в разі необхідності накласти текстуру. В сцені має бути мінімум одне джерело освітлення.

Виконати анімацію сцени таким чином, щоб можна було розглянути об'єкт з усіх сторін. За бажанням можна виконати інтерактивні взаємодію з об'єктом за допомогою миші та клавіатури.

**Варіант 7:**

Морозиво

**Лістинг коду програми**

import java.awt.Container;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import com.sun.j3d.utils.geometry.\*;

import com.sun.j3d.utils.image.TextureLoader;

import com.sun.j3d.utils.universe.SimpleUniverse;

import javax.media.j3d.\*;

import javax.swing.Timer;

import javax.vecmath.\*;

import java.applet.Applet;

import javax.media.j3d.Appearance;

import javax.media.j3d.Material;

import javax.vecmath.Color3f;

import com.sun.j3d.utils.geometry.Cone;

import com.sun.j3d.utils.geometry.Primitive;

import java.awt.Container;

import javax.media.j3d.Appearance;

import javax.media.j3d.Material;

import javax.media.j3d.Texture;

import javax.media.j3d.TextureAttributes;

import javax.vecmath.Color3f;

import javax.vecmath.Color4f;

import com.sun.j3d.utils.geometry.Sphere;

import com.sun.j3d.utils.geometry.Primitive;

import com.sun.j3d.utils.image.TextureLoader;

public class LabFour extends Applet implements ActionListener {

private TransformGroup treeTransformGroup = new TransformGroup();

private final Transform3D treeTransform3D = new Transform3D();

private final Timer timer;

private float angle = 0;

public static void main(String[] args) {

new LabFour();

}

public LabFour() {

timer = new Timer(50, this);

timer.start();

BranchGroup scene = createSceneGraph();

SimpleUniverse u = new SimpleUniverse();

u.getViewingPlatform().setNominalViewingTransform();

u.addBranchGraph(scene);

}

public BranchGroup createSceneGraph() {

BranchGroup objRoot = new BranchGroup();

treeTransformGroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_WRITE);

buildIceCream();

objRoot.addChild(treeTransformGroup);

BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);

Color3f light1Color = new Color3f(1.0f, 0.5f, 0.4f);

Vector3f light1Direction = new Vector3f(4.0f, -7.0f, -12.0f);

DirectionalLight light1 = new DirectionalLight(light1Color,

light1Direction);

light1.setInfluencingBounds(bounds);

objRoot.addChild(light1);

Color3f ambientColor = new Color3f(1.0f, 1.0f, 1.0f);

AmbientLight ambientLightNode = new AmbientLight(ambientColor);

ambientLightNode.setInfluencingBounds(bounds);

objRoot.addChild(ambientLightNode);

return objRoot;

}

private void buildIceCream() {

TransformGroup cone = new TransformGroup();

Transform3D transformTop = new Transform3D();

Appearance coneAp = new Appearance();

Color3f emissive = new Color3f(0.0f, 0.0f, 0.0f);

Color3f ambient = new Color3f(1, 1, 0.6f);

Color3f diffuse = new Color3f(0.0f, 0.0f, 0.0f);

Color3f specular = new Color3f(0.0f, 0.0f, 0.0f);

TextureLoader loader = new TextureLoader("D:\\projects\\maokg\\lab4\\lab4\\texture.jpg", "LUMINANCE", new Container());

Texture texture = loader.getTexture();

texture.setBoundaryModeS(Texture.WRAP);

texture.setBoundaryModeT(Texture.WRAP);

texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 1.0f, 0.0f));

TextureAttributes texAttr = new TextureAttributes();

texAttr.setTextureMode(TextureAttributes.MODULATE);

coneAp.setTexture(texture);

coneAp.setTextureAttributes(texAttr);

coneAp.setMaterial(new Material(ambient, emissive, diffuse, specular, 1.0f));

int primflags = Primitive.GENERATE\_NORMALS + Primitive.GENERATE\_TEXTURE\_COORDS;

Cone coneTop = new Cone(0.3f, 0.5f, primflags, coneAp);

transformTop.rotX(Math.PI + Math.PI / 20);

cone.setTransform(transformTop);

cone.addChild(coneTop);

treeTransformGroup.addChild(cone);

// BALLS

createBallOne();

createBallTwo();

createBallThree();

createBallFour();

}

private void createBallOne() {

Color3f emissive = new Color3f(0.0f, 0.0f, 0.0f);

Color3f ambient = new Color3f(0.99f, 0.6f, 0.8f);

Color3f diffuse = new Color3f(0.0f, 0.0f, 0.0f);

Color3f specular = new Color3f(0.0f, 0.0f, 0.0f);

Appearance ballOneAp = new Appearance();

int primflags = Primitive.GENERATE\_NORMALS + Primitive.GENERATE\_TEXTURE\_COORDS;

TextureLoader loaderBall = new TextureLoader("D:\\projects\\maokg\\lab4\\lab4\\iceCreamBall.jpg", "LUMINANCE", new Container());

Texture texture = loaderBall.getTexture();

texture.setBoundaryModeS(Texture.WRAP);

texture.setBoundaryModeT(Texture.WRAP);

texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 1.0f, 0.0f));

TextureAttributes texAttr = new TextureAttributes();

texAttr.setTextureMode(TextureAttributes.MODULATE);

ballOneAp.setTexture(texture);

ballOneAp.setTextureAttributes(texAttr);

ballOneAp.setMaterial(new Material(ambient, emissive, diffuse, specular, 1.0f));

TransformGroup tg = new TransformGroup();

Transform3D transform = new Transform3D();

Sphere ballOne = new Sphere(0.2f, primflags, ballOneAp);

Vector3f vector = new Vector3f(0.0f, 0.3f, -0.038f);

transform.setTranslation(vector);

tg.setTransform(transform);

tg.addChild(ballOne);

treeTransformGroup.addChild(tg);

}

private void createBallTwo() {

Color3f emissive = new Color3f(0.0f, 0.0f, 0.0f);

Color3f ambient = new Color3f(0.4f, 0.6f, 1f);

Color3f diffuse = new Color3f(0.0f, 0.0f, 0.0f);

Color3f specular = new Color3f(0.0f, 0.0f, 0.0f);

Appearance ballAp = new Appearance();

int primflags = Primitive.GENERATE\_NORMALS + Primitive.GENERATE\_TEXTURE\_COORDS;

TextureLoader loaderBall = new TextureLoader("D:\\projects\\maokg\\lab4\\lab4\\iceCreamBall.jpg", "LUMINANCE", new Container());

Texture texture = loaderBall.getTexture();

texture.setBoundaryModeS(Texture.WRAP);

texture.setBoundaryModeT(Texture.WRAP);

texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 1.0f, 0.0f));

TextureAttributes texAttr = new TextureAttributes();

texAttr.setTextureMode(TextureAttributes.MODULATE);

ballAp.setTexture(texture);

ballAp.setTextureAttributes(texAttr);

ballAp.setMaterial(new Material(ambient, emissive, diffuse, specular, 1.0f));

TransformGroup tg = new TransformGroup();

Transform3D transform = new Transform3D();

Sphere ballOne = new Sphere(0.2f, primflags, ballAp);

Vector3f vector = new Vector3f(0.08f, 0.3f, 0.11f);

transform.setTranslation(vector);

tg.setTransform(transform);

tg.addChild(ballOne);

treeTransformGroup.addChild(tg);

}

private void createBallThree() {

Color3f emissive = new Color3f(0.0f, 0.0f, 0.0f);

Color3f ambient = new Color3f(1f, 0.6f, 0.2f);

Color3f diffuse = new Color3f(0.0f, 0.0f, 0.0f);

Color3f specular = new Color3f(0.0f, 0.0f, 0.0f);

Appearance ballAp = new Appearance();

int primflags = Primitive.GENERATE\_NORMALS + Primitive.GENERATE\_TEXTURE\_COORDS;

TextureLoader loaderBall = new TextureLoader("D:\\projects\\maokg\\lab4\\lab4\\iceCreamBall.jpg", "LUMINANCE", new Container());

Texture texture = loaderBall.getTexture();

texture.setBoundaryModeS(Texture.WRAP);

texture.setBoundaryModeT(Texture.WRAP);

texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 1.0f, 0.0f));

TextureAttributes texAttr = new TextureAttributes();

texAttr.setTextureMode(TextureAttributes.MODULATE);

ballAp.setTexture(texture);

ballAp.setTextureAttributes(texAttr);

ballAp.setMaterial(new Material(ambient, emissive, diffuse, specular, 1.0f));

TransformGroup tg = new TransformGroup();

Transform3D transform = new Transform3D();

Sphere ballOne = new Sphere(0.2f, primflags, ballAp);

Vector3f vector = new Vector3f(-0.07f, 0.3f, 0.1f);

transform.setTranslation(vector);

tg.setTransform(transform);

tg.addChild(ballOne);

treeTransformGroup.addChild(tg);

}

private void createBallFour() {

Color3f emissive = new Color3f(0.0f, 0.0f, 0.0f);

Color3f ambient = new Color3f(0.8f, 1f, 0.6f);

Color3f diffuse = new Color3f(0.0f, 0.0f, 0.0f);

Color3f specular = new Color3f(0.0f, 0.0f, 0.0f);

Appearance ballAp = new Appearance();

int primflags = Primitive.GENERATE\_NORMALS + Primitive.GENERATE\_TEXTURE\_COORDS;

TextureLoader loaderBall = new TextureLoader("D:\\projects\\maokg\\lab4\\lab4\\iceCreamBall.jpg", "LUMINANCE", new Container());

Texture texture = loaderBall.getTexture();

texture.setBoundaryModeS(Texture.WRAP);

texture.setBoundaryModeT(Texture.WRAP);

texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 1.0f, 0.0f));

TextureAttributes texAttr = new TextureAttributes();

texAttr.setTextureMode(TextureAttributes.MODULATE);

ballAp.setTexture(texture);

ballAp.setTextureAttributes(texAttr);

ballAp.setMaterial(new Material(ambient, emissive, diffuse, specular, 1.0f));

TransformGroup tg = new TransformGroup();

Transform3D transform = new Transform3D();

Sphere ballOne = new Sphere(0.2f, primflags, ballAp);

Vector3f vector = new Vector3f(0.04f, 0.4f, 0.04f);

transform.setTranslation(vector);

tg.setTransform(transform);

tg.addChild(ballOne);

treeTransformGroup.addChild(tg);

}

@Override

public void actionPerformed(ActionEvent e) {

treeTransform3D.rotY(angle);

// treeTransform3D.rotX( Math.PI / 2);

Vector3f vectorTop = new Vector3f(-.0f, -0.2f, -.0f);

treeTransform3D.setTranslation(vectorTop);

treeTransformGroup.setTransform(treeTransform3D);

angle += 0.05;

}

}

**Результат**

****