НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ І. СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

ЛАБОРАТОРНА РОБОТА №5

з дисципліни

"МАТЕМАТИЧНІ ТА АЛГОРИТМІЧНІ ОСНОВИ КОМП’ЮТЕРНОЇ ГРАФІКИ"

ТЕМА: «Імпорт тривімирних моделей у середовище програмування java3D, обробка та маніапуляція цих зображень»

Підготував:

студент групи КП-71

Пилипенко І.О.

Київ-2020

**Код**

import com.sun.j3d.loaders.Loader;  
import com.sun.j3d.loaders.Scene;  
import com.sun.j3d.loaders.lw3d.Lw3dLoader;  
import com.sun.j3d.utils.image.TextureLoader;  
import com.sun.j3d.utils.universe.SimpleUniverse;  
import javax.media.j3d.\*;  
import javax.swing.\*;  
import javax.vecmath.\*;  
import java.awt.event.KeyAdapter;  
import java.awt.event.KeyEvent;  
  
  
public class Main extends JFrame {  
 static Color3f *gray* = new Color3f(0.2f, 0.2f, 0.2f);  
 static Material *materialGates* = new Material(new Color3f( 0.33f, 0.26f, 0.23f ), new Color3f( 0.33f, 0.26f, 0.23f ),  
 new Color3f( 0.50f, 0.11f, 0.00f ),new Color3f( 0.95f, 0.73f, 0.00f ), 0.9f);  
  
 static Material[] *materials* = new Material[] {  
 new Material(new Color3f( 0.03f, 0.26f, 0.03f ), new Color3f( 0.2f, 0.26f, 0.03f ),  
 new Color3f( 0.1f, 0.11f, 0.00f ),new Color3f( 0.4f, 0.73f, 0.00f ), 0.9f),  
 new Material(*gray*, *gray*, *gray*, *gray*, 0.9f),  
 new Material(new Color3f( 0.10f, 0.08f, 0.10f ), new Color3f(0.10f, 0.08f, 0.10f ),  
 new Color3f( 0.25f, 0.03f, 0.00f ),new Color3f( 0.45f, 0.15f, 0.00f ), 0.9f),  
  
 *materialGates*, *materialGates*, *materialGates*, *materialGates*, *materialGates*,  
  
 };  
 static SimpleUniverse *universe*;  
 static Canvas3D *canvas*;  
 static View *myView*;  
 static BranchGroup *sceneRoot*;  
 static TransformGroup *mainTransformGroup*;  
 static Transform3D *mainTransform3D*;  
 static double *angle* = 0;  
 static int *rotate* = 0;  
  
 Main(){  
 *canvas* = new Canvas3D(SimpleUniverse.*getPreferredConfiguration*());  
 init();  
 add(*canvas*);  
 setDefaultCloseOperation(WindowConstants.*EXIT\_ON\_CLOSE*);  
 setSize(800, 600);  
 setLocationRelativeTo(null);  
 setVisible(true);  
 *canvas*.addKeyListener(new KeyAdapter() {  
 @Override  
 public void keyPressed(KeyEvent e) {  
 if(e.getKeyCode() == KeyEvent.*VK\_LEFT*) {  
 *rotate* = -1;  
 } else if(e.getKeyCode() == KeyEvent.*VK\_RIGHT*) {  
 *rotate* = +1;  
 }  
 }  
 @Override  
 public void keyReleased(KeyEvent e) {  
 if(e.getKeyCode() == KeyEvent.*VK\_LEFT* && *rotate* == -1) {  
 *rotate* = 0;  
 } else if(e.getKeyCode() == KeyEvent.*VK\_RIGHT* && *rotate* == +1) {  
 *rotate* = 0;  
 }  
 }  
 });  
 new Timer(16, (e) -> {  
 *angle* += 0.005 \* *rotate*;  
 *mainTransform3D*.rotZ(*angle*);  
 *mainTransformGroup*.setTransform(*mainTransform3D*);  
 }).start();  
 }  
  
 public Node createBackground() {  
 TextureLoader t = new TextureLoader("sky.png", *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);  
 return background;  
 }  
  
 public void init() {  
 Loader lw3dLoader = new Lw3dLoader(Loader.*LOAD\_ALL*);  
 Scene loaderScene = null;  
  
 try {  
 loaderScene = lw3dLoader.load("scene.lws");  
 } catch (Exception e) {  
 e.printStackTrace();  
 System.*exit*(1);  
 }  
  
 *sceneRoot* = new BranchGroup();  
  
 *universe* = new SimpleUniverse(*canvas*);  
 *myView* = *universe*.getViewer().getView();  
 *myView*.setBackClipDistance(50000f);  
  
  
 if (loaderScene.getSceneGroup() != null) {  
 TransformGroup viewGroups[] = loaderScene.getViewGroups();  
 Transform3D t = new Transform3D();  
 viewGroups[0].getTransform(t);  
 Matrix4d m = new Matrix4d();  
 t.get(m);  
 m.invert();  
 t.set(m);  
 TransformGroup tg = new TransformGroup(t);  
 *mainTransform3D* = new Transform3D();  
 tg.addChild(loaderScene.getSceneGroup());  
 *mainTransformGroup* = new TransformGroup(*mainTransform3D*);  
 *mainTransformGroup*.addChild(createBackground());  
 *mainTransformGroup*.addChild(tg);  
 *mainTransformGroup*.setCapability(TransformGroup.*ALLOW\_TRANSFORM\_WRITE*);  
 *proceedGroup*(*mainTransformGroup*);  
 *sceneRoot*.addChild(*mainTransformGroup*);  
 }  
 *universe*.addBranchGraph(*sceneRoot*);  
 }  
  
 static int *materialCounter* = 0;  
  
 private static void proceedGroup(Group tg) {  
 for(int i = tg.numChildren() - 1; i >= 0; --i) {  
 Node node = tg.getChild(i);  
 if(node instanceof Group) {  
 *proceedGroup*((Group) node);  
 } else if(node instanceof Shape3D) {  
 TextureLoader tl = new TextureLoader("sky.png", "LUMINANCE", *canvas*);  
 Texture texture = tl.getTexture();  
 texture.setBoundaryModeS(Texture.*WRAP*);  
 texture.setBoundaryModeT(Texture.*WRAP*);  
 texture.setBoundaryColor(new Color4f(0.0f, 1.0f, 0.0f, 0.0f));  
 TextureAttributes texAttr = new TextureAttributes();  
 texAttr.setTextureMode(TextureAttributes.*MODULATE*);  
 Appearance ap = new Appearance();  
 ap.setTexture(texture);  
 ap.setTextureAttributes(texAttr);  
 ap.setMaterial(*materials*[*materialCounter*]);  
 *materialCounter*++;  
 if(*materialCounter* >= *materials*.length) {  
 *materialCounter* = 0;  
 }  
 ((Shape3D) node).setAppearance(ap);  
 }  
 }  
 }  
  
 public static void main(String[] args) {  
 try {  
 new Main();  
 } catch (Exception e) {  
 e.printStackTrace();  
 System.*exit*(2);  
 }  
 }  
}

**Малюнки та анімації**

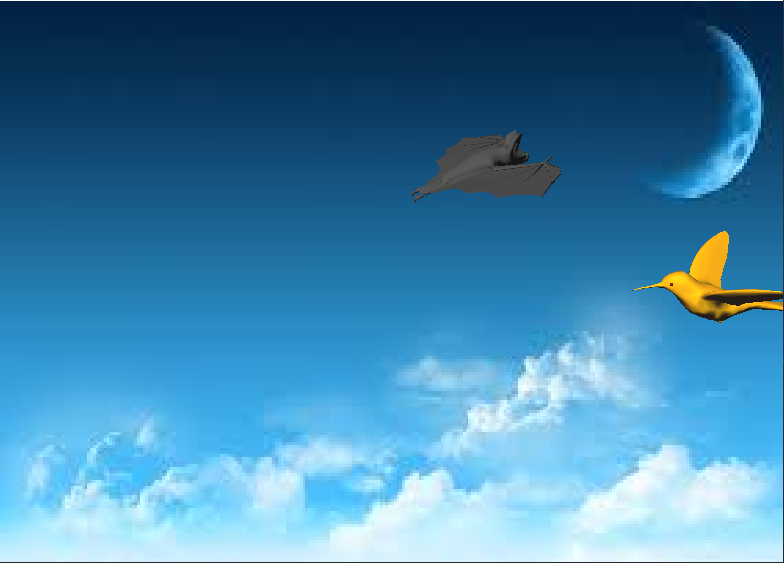
****

Рис1. Малюнок