



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”

Виконала
студентка III курсу
групи КП-81

Мозгова Катерина Олегівна
(*прізвище, ім’я, по батькові*)

варіант № 12

Зарахована
“ ____ ” “ ____ ” 20__ р.
викладачем

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

Завдання за варіантом

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 взаємодії з об'єктом за допомогою миші та клавіатури.

Варіант: мухомор.

Код програми

Main.java

```
package sample;

import com.sun.j3d.utils.applet.MainFrame;
import com.sun.j3d.utils.geometry.*;

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

import javax.media.j3d.*;
import javax.swing.*;
import java.applet.Applet;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.media.j3d.AmbientLight;

import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.DirectionalLight;

import javax.vecmath.Color3f;
import javax.vecmath.Color4f;
import javax.vecmath.Point3d;
import javax.vecmath.Vector3f;
import com.sun.j3d.utils.geometry.Sphere;

public class Main extends Applet implements ActionListener {
    private final TransformGroup tg = new TransformGroup();
    private final Transform3D t3D = new Transform3D();
    private final Timer timer = new Timer(50, this);
    private float angle = 0;
    private double scale = 0;
    private boolean rotateY = true;
    private boolean isDecreasing = false;

    public static void main(String[] args) {
        var obj = new Main();
        MainFrame mf = new MainFrame(obj, 600, 600);
        mf.run();
    }

    private Main() {
        setLayout(new BorderLayout());
        GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
        Canvas3D c = new Canvas3D(config);
        add("Center", c);
        SimpleUniverse universe = new SimpleUniverse(c);
```

```

timer.start();

universe.getViewingPlatform().setNominalViewingTransform();
universe.addBranchGraph(createSceneGraph());
}

private BranchGroup createSceneGraph() {
    BranchGroup root = new BranchGroup();

    tg.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
    root.addChild(tg);
    buildFungus();

    //light
    BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0),100);

    Color sunLightColor = new Color(242, 255, 0);
    DirectionalLight lightDirect = new DirectionalLight(new Color3f(sunLightColor), new Vector3f(0, 0,
0));
    lightDirect.setInfluencingBounds(bounds);
    root.addChild(lightDirect);

    AmbientLight ambientLightNode = new AmbientLight(new Color3f(new Color(100, 255, 255)));
    ambientLightNode.setInfluencingBounds(bounds);
    root.addChild(ambientLightNode);
    //

    return root;
}

private void buildFungus() {
    //leg
    TransformGroup fungusLegGroup = new TransformGroup();
    Transform3D transformLeg = new Transform3D();
    Cylinder fungusLeg = Fungus.getLeg(0.1f, 0.4f);
    Vector3f vectorLeg = new Vector3f(.0f, 0.1f, .0f);
    transformLeg.setTranslation(vectorLeg);
    fungusLegGroup.setTransform(transformLeg);
    fungusLegGroup.addChild(fungusLeg);
    tg.addChild(fungusLegGroup);

    //middle leg
    TransformGroup fungusMiddleLegGroup = new TransformGroup();
    Transform3D transformMiddleLeg = new Transform3D();
    Cylinder fungusMiddleLeg = Fungus.getMiddleLeg(0.2f, 0.1f);
    Vector3f vectorMiddleLeg = new Vector3f(.0f, 0.3f, .0f);
    transformMiddleLeg.setTranslation(vectorMiddleLeg);
    fungusMiddleLegGroup.setTransform(transformMiddleLeg);
    fungusMiddleLegGroup.addChild(fungusMiddleLeg);
    tg.addChild(fungusMiddleLegGroup);

    //head

```

```
TransformGroup fungusHeadGroup = new TransformGroup();
Transform3D transformHead = new Transform3D();
Cone fungusHead = Fungus.getHead(0.4f, 0.2f);
Vector3f vectorHead = new Vector3f(0f, 0.4f, .0f);
transformHead.setTranslation(vectorHead);
fungusHeadGroup.setTransform(transformHead);
fungusHeadGroup.addChild(fungusHead);
tg.addChild(fungusHeadGroup);
```

```
//circle1
TransformGroup fungusCircleGroup1 = new TransformGroup();
Transform3D transformCircle1 = new Transform3D();
Sphere fungusCircle1 = Fungus.getCircle(0.05f);
Vector3f vectorCircle1 = new Vector3f(0.1f, 0.45f, .0f);
transformCircle1.setTranslation(vectorCircle1);
fungusCircleGroup1.setTransform(transformCircle1);
fungusCircleGroup1.addChild(fungusCircle1);
tg.addChild(fungusCircleGroup1);
```

```
//circle2
TransformGroup fungusCircleGroup2 = new TransformGroup();
Transform3D transformCircle2 = new Transform3D();
Sphere fungusCircle2 = Fungus.getCircle(0.05f);
Vector3f vectorCircle2 = new Vector3f(-0.1f, 0.45f, .0f);
transformCircle2.setTranslation(vectorCircle2);
fungusCircleGroup2.setTransform(transformCircle2);
fungusCircleGroup2.addChild(fungusCircle2);
tg.addChild(fungusCircleGroup2);
```

```
//circle3
TransformGroup fungusCircleGroup3 = new TransformGroup();
Transform3D transformCircle3 = new Transform3D();
Sphere fungusCircle3 = Fungus.getCircle(0.05f);
Vector3f vectorCircle3 = new Vector3f(0.3f, 0.35f, .0f);
transformCircle3.setTranslation(vectorCircle3);
fungusCircleGroup3.setTransform(transformCircle3);
fungusCircleGroup3.addChild(fungusCircle3);
tg.addChild(fungusCircleGroup3);
```

```
//circle4
TransformGroup fungusCircleGroup4 = new TransformGroup();
Transform3D transformCircle4 = new Transform3D();
Sphere fungusCircle4 = Fungus.getCircle(0.05f);
Vector3f vectorCircle4 = new Vector3f(-0.3f, 0.35f, .0f);
transformCircle4.setTranslation(vectorCircle4);
fungusCircleGroup4.setTransform(transformCircle4);
fungusCircleGroup4.addChild(fungusCircle4);
tg.addChild(fungusCircleGroup4);
```

```
}
```

```
@Override
public void actionPerformed(ActionEvent e) {
    t3D.rotY(angle);
```

```
    angle += 0.05;

    if (angle >= 25) {
        rotateY = !rotateY;
        angle = 0;
    }

    tg.setTransform(t3D);
}
```

Результат

```
"C:\Program Files\Java\jdk-11.0.10\bin\java.exe" "-javaagent:C:\Program
Files\JetBrains\IntelliJ IDEA Community Edition
2021.1.1\lib\idea_rt.jar=51729:C:\Program Files\JetBrains\IntelliJ IDEA
Community Edition 2021.1.1\bin" -Dfile.encoding=UTF-8 -p
"C:\КПИ\MAOKG\maokg\l4\out\production\l4;C:\КПИ\MAOKG\maokg\l4\lib
\javafx-fxml-11.jar;C:\КПИ\MAOKG\maokg\l4\lib\javafx-fxml-11-win.jar;C:\К
ПИ\MAOKG\maokg\l4\lib\javafx-controls-11.jar;C:\КПИ\MAOKG\maokg\l4\li
b\javafx-controls-11-win.jar;C:\КПИ\MAOKG\maokg\l4\lib\javafx-graphics-11.
jar;C:\КПИ\MAOKG\maokg\l4\lib\javafx-graphics-11-win.jar;C:\КПИ\MAOK
G\maokg\l4\lib\javafx-base-11.jar;C:\КПИ\MAOKG\maokg\l4\lib\javafx-base-1
1-win.jar;C:\Program Files\Java\Java3D\1.5.1\lib\ext\j3dcore.jar;C:\Program
Files\Java\Java3D\1.5.1\lib\ext\vecmath.jar;C:\Program
Files\Java\Java3D\1.5.1\lib\ext\j3dutils.jar" -m l4/sample.Main

Exception in thread "J3D-Renderer-1" java.lang.IllegalAccessError: class
javax.media.j3d.Win32NativeConfigTemplate3D (in module j3dcore) cannot
access class sun.awt.Win32GraphicsConfig (in module java.desktop) because
module java.desktop does not export sun.awt to module j3dcore

    at
j3dcore/javax.media.j3d.Win32NativeConfigTemplate3D.getBestConfiguration(
Win32NativeConfigTemplate3D.java:57)

    at
j3dcore/javax.media.j3d.NativePipeline.getBestConfiguration(NativePipeline.jav
a:3296)

    at j3dcore/javax.media.j3d.Renderer.doWork(Renderer.java:495)

    at j3dcore/javax.media.j3d.J3dThread.run(J3dThread.java:256)
```

Рис. 2. Результат роботи програми