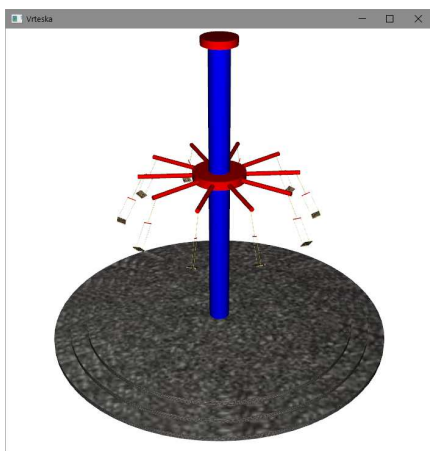
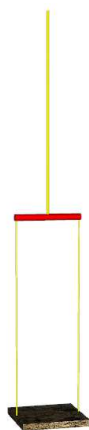


ISPIT IZ RAČUNARSKE GRAFIKE *praktični deo*

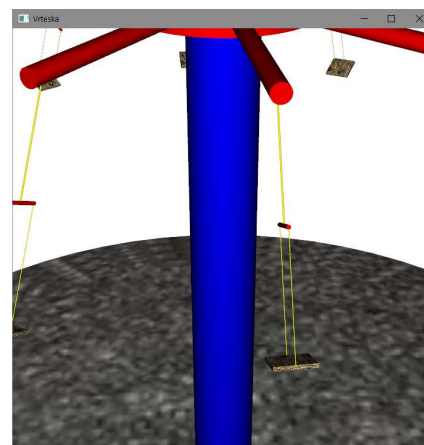
- 1) [50] Napisati program koji koristi grafičku biblioteku JavaFX i prikazuje animiranu vrtešku sa priložene slike 1. Na horizontalnim crvenim šipkama, koje su radijalno orijentisane, ravnomerno raspoređene i pričvršćene za crveni prsten na vertikalnom plavom stubu, nalaze se ljuljaške na žutim užadima sa crvenom horizontalnom prečkom, prema slici 2. Sedište ljuljaške je od drveta, a podijum sastavljen od 3 diska je od metala. Dimenzije usvojiti da model približno odgovara slikama, pri čemu je vertikalni plavi stub dužine 350, poluprečnika 12. Prsten klizi naviše, odnosno naniže, po vertikalnom stubu kad se vrteska ubrzava (strelica naviše), odnosno usporava (strelica naniže), respektivno. U donjem položaju prstena sedišta ljuljaški su malo iznad podijuma, vrteška rotira malom brzinom i ljuljaške nemaju otklon prema vertikalnoj osi (uže ljuljaške je vertikalno), dok se sa povećanjem brzine povećava otklon ljuljaški. Pri maksimalnoj brzini prsten dolazi do gornjeg crvenog diska-graničnika, a otklon ljuljaški je oko 60° u odnosu na vertikalnu osu. Točkićem miša omogućiti efekat zoom-in/zoom-out (slika 3). Priložen je demonstracioni .jar fajl i slike za potrebne teksture.



Slika 1



Slika 2



Slika 3

Napomene:

1. Praktični deo ispita, odnosno izrada zadatka 1 traje 150 minuta.
2. Rešenje zadatka se predaje u obliku NetBeans projekta u predviđenom folderu na računaru.
3. Dozvoljena je upotreba literature koja je stavljena na raspolaganje i pristup *Oracle* sajtu.
4. Nije dozvoljeno uz sebe imati mobilni telefon, bez obzira da li je uključen ili isključen.

Rešenje zadatka

1)

```

package RG_16_17_11;
import javafx.application.Application;
import javafx.stage.Stage;
import javafx.scene.Scene;
import javafx.scene.SceneAntialiasing;
import javafx.scene.Group;
import javafx.scene.PerspectiveCamera;
import javafx.scene.shape.Box;
import javafx.scene.shape.Cylinder;
import javafx.scene.paint.Color;
import javafx.scene.paint.PhongMaterial;
import javafx.scene.transform.Rotate;
import javafx.scene.transform.Translate;
import javafx.animation.Interpolator;
import javafx.animation.RotateTransition;
import javafx.animation.Timeline;
import javafx.util.Duration;
import javafx.scene.image.Image;

class Materijali {
    final static PhongMaterial DRVO = new PhongMaterial();
    final static PhongMaterial METAL = new PhongMaterial();
    static {
        DRVO.setDiffuseMap(new Image("drvo.jpg"));
        METAL.setDiffuseMap(new Image("metal.jpg"));
    }
    final static PhongMaterial PLAVA = new PhongMaterial(Color.BLUE);
    final static PhongMaterial CRVENA = new PhongMaterial(Color.RED);
    final static PhongMaterial ZUTA = new PhongMaterial(Color.YELLOW);
}

class Ljuljaska extends Group {
    Ljuljaska() {
        Box sediste = new Box(10, 1, 10);
        sediste.setTranslateY(60);
        sediste.setMaterial(Materijali.DRVO);
        for (int i = 0; i < 2; i++) {
            Cylinder uze = new Cylinder(0.1, 30); uze.setMaterial(Materijali.ZUTA);
            uze.getTransforms().addAll(new Translate(i%2==1 ? 4.75 : -4.75, 45, 0));
            this.getChildren().add(uze);
        }
        Cylinder precka = new Cylinder(0.5, 10);
        precka.setMaterial(Materijali.CRVENA);
        precka.getTransforms().addAll(new Rotate(90, Rotate.Z_AXIS),
            new Translate(30, 0, 0));
        this.getChildren().add(precka);
        Cylinder uze = new Cylinder(0.2, 30); uze.setMaterial(Materijali.ZUTA);
        uze.getTransforms().addAll(new Translate(0, 15, 0));
        this.getChildren().addAll(sediste, uze);
    }
}

class Vrteska extends Group {
    final private Group nosacLjuljaski;
    final private RotateTransition okretanje;
    final private Rotate ugaoLjuljaski;
    Vrteska() {
        Cylinder stub = new Cylinder(12, 350);
        stub.setTranslateY(-175);
        stub.setMaterial(Materijali.PLAVA);
        for (int i = 0; i < 3; i++) {
            Cylinder postolja = new Cylinder(200 - i * 20, 6);
            postolja.setTranslateY(-3 - i * 6);
            postolja.setMaterial(Materijali.METAL);
            this.getChildren().add(postolja);
        }
        Cylinder vrh = new Cylinder(20, 8);
        vrh.setMaterial(Materijali.CRVENA);
        vrh.setTranslateY(-354);
        nosacLjuljaski = new Group();
        Cylinder prsten = new Cylinder(30, 10);
        prsten.setMaterial(Materijali.CRVENA);
        nosacLjuljaski.getChildren().add(prsten);
        ugaoLjuljaski = new Rotate();
        ugaoLjuljaski.setAxis(Rotate.Z_AXIS);

        for (int i = 0; i < 10; i++) {
            Cylinder nosac = new Cylinder(2.5, 60);
            nosac.getTransforms().addAll(new Rotate(i * 36, Rotate.Y_AXIS),
                new Rotate(-90, Rotate.X_AXIS),
                new Translate(0, 30 + 30, 0));
            nosac.setMaterial(Materijali.CRVENA);

            Ljuljaska ljuljaska = new Ljuljaska();
            ljuljaska.getTransforms().addAll(new Rotate(i * 36, Rotate.Y_AXIS),
                new Translate(0, 0, 85),
                new Rotate(90, Rotate.Y_AXIS),
                ugaoLjuljaski);

            nosacLjuljaski.getChildren().addAll(nosac, ljuljaska);
        }
        nosacLjuljaski.setTranslateY(-100);
        this.getChildren().addAll(stub, vrh, nosacLjuljaski);
        okretanje = new RotateTransition(Duration.seconds(20), nosacLjuljaski);
        okretanje.setAxis(Rotate.Y_AXIS);
        okretanje.setByAngle(-360);
        okretanje.setInterpolator(Interpolator.LINEAR);
        okretanje.setCycleCount(Timeline.INDEFINITE);
        okretanje.play();
    }
    public void ubrzaj() {
        if (okretanje.getRate() < 7) okretanje.setRate(okretanje.getRate() + 0.02);
        pomeriNosac(); zaljuljajStolice();
    }
    public void uspori() {
        if (okretanje.getRate() > 1) okretanje.setRate(okretanje.getRate() - 0.02);
        pomeriNosac(); zaljuljajStolice();
    }
    private void pomeriNosac() {
        nosacLjuljaski.setTranslateY(-100 - (345-100)*(okretanje.getRate()-1)/6);
    }
    private void zaljuljajStolice() {
        ugaoLjuljaski.setAngle((okretanje.getRate() - 1)/6*60);
    }
}

public class II_2017 extends Application {
    @Override public void start(Stage prozor) {
        Group koren = new Group();

        Vrteska vrteska = new Vrteska();
        vrteska.getTransforms().addAll(new Translate(0,200,0));

        PerspectiveCamera kamera = new PerspectiveCamera(true);
        kamera.setFarClip(2000);
        Group nosacKamere = new Group(kamera);
        nosacKamere.getTransforms().addAll(new Translate(0, -380, -800),
            new Rotate(-30, Rotate.X_AXIS));
        koren.getChildren().addAll(vrteska, nosacKamere);
        SceneAntialiasing glatko=SceneAntialiasing.BALANCED;
        Scene scena = new Scene(koren, 700, 700, true, glatko);
        scena.setOnKeyPressed(dog -> {
            switch (dog.getCode()) {
                case DOWN: vrteska.uspori(); break;
                case UP: vrteska.ubrzaj(); break;
            }
        });
        scena.setOnScroll(dog -> {
            double korakZ = dog.getDeltaY(); double z = kamera.getTranslateZ();
            kamera.setTranslateZ(z + korakZ);
        });
        scena.setCamera(kamera); prozor.setScene(scena); prozor.setTitle("Vrteska");
        prozor.show();
    }
    public static void main(String[] args) { launch(args); }
}

```