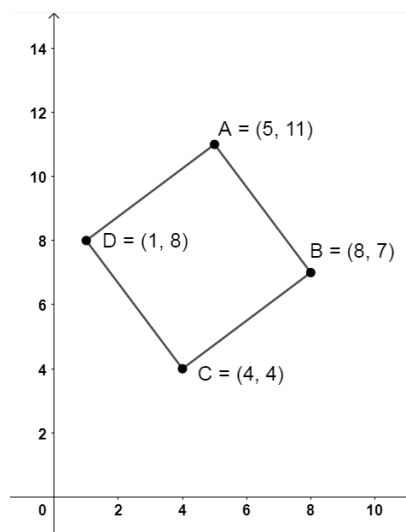


## RAČUNARSKA GRAFIKA

### *prvi kolokvijum – teoretski deo*

- 1) [20] Moodle test
- 2) [10] Posmatraju se objekti u desnom 2D koordinatnom sistemu. Potrebno je izvršiti preslikavanje objekata iz pravougaonog prozora (*window*) prikazanog na slici u prikazni prozor (*viewport*), pri čemu se slika temena C nalazi u tački  $(0.5, 0.5)$  normalizovanog koordinatnog sistema prikaznog prozora, a slika temena B leži na pravoj  $y=0.5$ , desno od tačke C. Prikazni prozor zauzima gornju desnu četvrtinu površi uređaja sa normalizovanim koordinatama. Postaviti jednu matičnu jednačinu preslikavanja tačaka iz zadatog prozora u zadati prikazni prozor. U sve matrice elementarnih transformacija uvrstiti konkretne vrednosti. Nije potrebno množiti matrice.



#### **Napomene:**

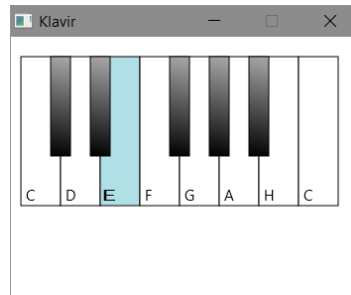
1. Izrada teoretskog dela (Moodle testa i zadatka) traje 20 minuta.
2. Rešenje 2. zadatka se predaje na ovom listu.
3. Nije dozvoljeno korišćenje literature i podsetnika u bilo kom obliku.
4. Nije dozvoljeno uz sebe imati mobilni telefon, bez obzira da li je uključen ili isključen.

#### **REŠENJE:**

## RAČUNARSKA GRAFIKA

### *prvi kolokvijum – praktični deo*

- 1) [70] Napisati program koji koristeći grafičku biblioteku JavaFX prikazuje animaciju pritiskanja dirki klavira. Klavir se sastoji od 8 belih i 5 crnih dirki, raspoređenih kao na slici. Svakoj beloj dirki je pripisana oznaka tona (C, D, E, F, G, A, H i C), koja se ispisuje u donjem levom uglu dirke. Program animira sviranje C-dur lestvice, pritiskanjem dirki sleva nadesno, a zatim i zdesna nalevo, uz neograničeno ponavljanje. Trajanje jednog ciklusa je 16s, pri čemu svaka nota traje isto. Pritisnuta dirka se boji plavom bojom. Oznaka pritisnute dirke se animirano povećava i smanjuje. Priložen je .jar fajl sa demonstracijom programa koji treba napisati.



#### **Napomene:**

1. Izrada praktičnog dela traje 100 minuta.
2. Rešenje zadatka se predaje u obliku .java fajla 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.

## Rešenja zadatka

prvi kolokvijum – teoretski deo

Kompozitna matrica:

$$M = T(4, 4) \times R(\arcsin(3/5)) \times S(0.5/5, 0.5/5) \times T(-0.5, -0.5)$$

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -4 & -4 & 1 \end{bmatrix} \times \begin{bmatrix} 4/5 & -3/5 & 0 \\ 3/5 & 4/5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0.5 & 0.5 & 1 \end{bmatrix}$$

Matrična jednačina:

$$P' = P \times M$$

$$[x' \quad y' \quad 1] = [x \quad y \quad 1] \times M$$

# Rešenja zadatka

## prvi kolokvijum – praktični deo

1)

```
package RG_1718_k1;

import javafx.animation.*;
import javafx.application.Application;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.paint.Color;
import javafx.scene.paint.CycleMethod;
import javafx.scene.paint.LinearGradient;
import javafx.scene.paint.Stop;
import javafx.scene.shape.Rectangle;
import javafx.scene.text.Text;
import javafx.stage.Stage;
import javafx.util.Duration;

class Klavir extends Group {
    private static final double T = 1000;
    private String[] oznake = {"C", "D", "E", "F", "G", "A", "H", "C"};

    public Klavir(){

        for(int i=0; i<8; i++){
            Rectangle dirka = new Rectangle(10+40*i, 20, 40, 150);
            dirka.setStroke(Color.BLACK);
            dirka.setFill(Color.WHITE);
            Text oznaka = new Text(oznake[i]);
            oznaka.setTranslateX(15+40*i);
            oznaka.setTranslateY(165);
            getChildren().addAll(dirka, oznaka);

            KeyValue dirka_iskljucena = new KeyValue(dirka.fillProperty(), Color.WHITE);
            KeyValue dirka_ukljucena = new KeyValue(dirka.fillProperty(), Color.POWDERBLUE);
            KeyValue oznaka_iskljucena = new KeyValue(oznaka.scaleXProperty(), 1);
            KeyValue oznaka_ukljucena = new KeyValue(oznaka.scaleXProperty(), 2);

            Timeline animacija = new Timeline(
                new KeyFrame(Duration.millis(0), dirka_iskljucena, oznaka_iskljucena),
                new KeyFrame(Duration.millis(i*T), dirka_iskljucena, oznaka_iskljucena),
                new KeyFrame(Duration.millis(i*T+T/5), dirka_ukljucena, oznaka_ukljucena),
                new KeyFrame(Duration.millis((i+1)*T-T/5), dirka_ukljucena, oznaka_ukljucena),
                new KeyFrame(Duration.millis((i+1)*T), dirka_iskljucena, oznaka_iskljucena),
                new KeyFrame(Duration.millis(16*T-(i+1)*T), dirka_iskljucena, oznaka_iskljucena),
                new KeyFrame(Duration.millis(16*T-(i+1)*T+T/5), dirka_ukljucena, oznaka_ukljucena),
                new KeyFrame(Duration.millis(16*T-i*T-T/5), dirka_ukljucena, oznaka_ukljucena),
                new KeyFrame(Duration.millis(16*T-i*T), dirka_iskljucena, oznaka_iskljucena),
                new KeyFrame(Duration.millis(16*T), dirka_iskljucena, oznaka_iskljucena));

            animacija.setCycleCount(Timeline.INDEFINITE);
            animacija.play();
        }

        for(int i=0; i<6; i++){
            if(i==2) continue;
            Rectangle dirka = new Rectangle(40+40*i, 20, 20, 100);
            dirka.setStroke(Color.BLACK);
            Stop[] stanice = {new Stop(1, Color.BLACK), new Stop(0, Color.DARKGRAY)};
            dirka.setFill(new LinearGradient(1,0,0,1,true, CycleMethod.NO_CYCLE, stanice));
            getChildren().add(dirka);
        }
    }
}

public class K1_2018 extends Application {
    @Override
    public void start(Stage primaryStage) {
        Group koren = new Group();
        koren.getChildren().add(new Klavir());
        Scene scena = new Scene(koren, 340, 250);
        primaryStage.setTitle("Klavir");
        primaryStage.setScene(scena);
        primaryStage.setResizable(false);
        primaryStage.show();
    }
    public static void main(String[] args) { launch(args); }
}
```