

Übungsblatt 5– EIPNF WS 18/19

Besprechung des Übungsblatts erfolgt am 04.12. & 5.12.

Aufgabe 1:

Gegeben ist der Code aus der Vorlesung. Um die Funktionalität der Klasse `RotatingRect` zu vervollständigen, muss ihre Animation noch fertiggestellt werden. Dazu soll das Rechteck abhängig vom `animationCounter` rotiert werden. Dazu sind folgende Schritte nötig.

1. Zeichne das Rechteck so, dass dessen Mittelpunkt auf dem Ursprung des Koordinatensystems liegt.
2. Rotiere das Rechteck abhängig vom `animationCounter`. Denke daran: Wir rotieren mit einem Faktor gleich dem Bruchteil von $2 \cdot \pi$ ($2 \cdot \pi$ entspricht einer Drehung um 360°)
3. Verschiebe das Rechteck an seine Zielkoordinaten.
4. Deklariere und erzeuge ein Objekt dieser Klasse und lasse es dir anzeigen.

```
void setup () {
  size(600,400);
}

void draw () {
  background(0);
  // Aufruf von .animate()
}

void keyPressed () {
  println(„DOWN: “ + key);
  // Aufruf von .start(key)
}

void keyReleased () {
  println(„UP: “ + key);
  // Aufruf von .stop(key)
}

// Neuer Tab: Interfaces -----

interface Triggerable {
  void start(char keyInput);
  void stop(char keyInput);
}

interface Animateable {
  void animate();
}
```

```
// Neuer Tab: Animation -----
class Animation implements Animateable, Triggerable {
    char triggerKey;
    int animationCounter;

    public Animation (char triggerKey) {
        this.triggerKey = triggerKey;
        animationCounter = 0;
    }

    void start (char keyInput) {
        if (keyInput == triggerKey && animationCounter == 0) {
            animationCounter = 1;
        }
    }

    void stop (char keyInput) {
        if (keyInput == triggerKey) {
            animationCounter = 0;
        }
    }

    void animate () {}
}

```

```
// Neuer Tab: RotatingRect -----
class RotatingRect extends Animation {
    int x,y;

    public RotatingRect (int x, int y, char triggerKey) {
        super(triggerKey);
        this.x = x;
        this.y = y;
    }

    void animate () {
        if (animationCounter>0) {
            animationCounter++;

            // verschiebe Rechteck
            // rotiere Rechteck
            // zeichne Rechteck
        }
    }
}

```

Aufgabe 2:

Schreibe nun eine Klasse `ColorCircle`, die eine Kind der Klasse `Animation` ist. Diese soll auf Tastendruck einen Kreis zeichnen der stets mit einer zufälligen Farbe gefüllt wird.

1. Führe bei Bedarf neue Felder ein.
2. Schreibe Konstruktor.
3. Überschreibe `animate`.