



# Programmierung für Mathematik HS10

## Übung 4

### 1 Aufgabe: Fehler finden

#### 1.1 Lernziele

1. Verschiedenartige Fehler in einem Programmcode finden und beheben können.
2. Programmstil verbessern können.

#### 1.2 Aufgabenstellung

1. Lesen Sie die folgende Klasse `Person` genau durch.

```
1 public class Person
2     public String name;
3     public long age;
4     public double weight;
5
6     public void setPerson(String name, int age, double weight){
7         this.name = name
8         this.age = age;
9         this.weight = weight;
10    }
11    public void setPerson(String name){
12        this.name = name;
13        this.age = 0;
14        this.weight = 0;
15    }
16    public String sayWord (String word){
17        System.out.println(word);
18    }
19    public String sayWord (String word){
20        System.out.println(word.toUpperCase());
21    }
```

```
22     public void countTo(int number) {
23         int i = 1;
24         while (i<=number) {
25             System.out.println(i);
26         }
27     }
28     public void inverseNarrate(String[] story) {
29         for (int i=story.length; i>=0; i--){
30             System.out.println(story[i]);
31         }
32     }
33     ...
34 }
```

**Listing 1:** Person Klasse

2. Verändern Sie den Code so, dass die Klasse korrekt kompilieren würde.
3. Beheben Sie alle Fehler, die zu Laufzeit auftreten würden (runtime error).
4. Verbessern Sie den Programmstil

## 2 Aufgabe: geometrische Figuren

### 2.1 Lernziele

1. Kontrollstrukturen üben.
2. Mit Hilfsmethoden programmieren können.

### 2.2 Aufgabenstellung

1. Schreiben Sie eine Klasse, welche Rechtecke und gleichschenklige Dreiecke für eine beliebige Grösse  $n \in \mathbb{N}_{\geq 2}$  auf der Kommandozeile ausgeben soll (die Höhe soll jeweils  $n$  Zeilen betragen.) Beispiel für  $n = 4$ :

```
****      bzw.      *
*  *          *  *
*  *          *    *
****          *      *
                * * * * *
```

(d.h. die Klasse soll die Methoden `printTriangle` und `printRectangle` besitzen, welche als Übergabeparameter eine Zahl erwarten, die die Grösse der Figuren bestimmt).

Verwenden Sie Hilfsmethoden, um Codeduplizität zu vermeiden. Die Hilfsmethoden sollen `private` sein, da sie nur von den Methoden der Klasse selber verwendet werden sollen.

2. Testen Sie Ihre Klasse mit einem TestDriver

### 3 Aufgabe: Polynome von beliebigem Grad

#### 3.1 Lernziele

1. Mit eindimensionalen Arrays rechnen können.

#### 3.2 Aufgabenstellung

1. Schreiben Sie eine Java-Klasse `Polynomial`, welche ein Polynom  $p(x) = a_0 + a_1x + \dots + a_nx^n$  für beliebiges  $n \in \mathbb{N}$  darstellen soll, indem die Koeffizienten von  $p$  gespeichert werden. (Tipp: benutzen Sie ein Array um die Koeffizienten des Polynoms zu speichern.)
2. Ergänzen Sie die Klasse um die Methode `toString`, welche das Polynom (d.h. zum Beispiel  $p(x) = 2 + 4x + 0x^2 - 6x^3$ ) als String zurückgeben soll.
3. Ergänzen Sie Ihre Klasse `Polynomial` um die Methode `getValueOf(double x)`, welche den Wert des Polynoms an der Stelle  $x$  berechnen soll. Implementieren Sie die Methode so, dass der Rechenaufwand möglichst gering ist, d.h. dass die Methode möglichst effizient ist. (Tipp: Beachten Sie dazu vor allem die Berechnung von  $x^m$ ).
4. Ergänzen Sie die Klasse um die Methode `getDerivative`, welche die Ableitung  $p'(x)$  berechnen und als `return` Wert zurückgeben soll.
5. Ergänzen Sie Ihre Klasse um die Methode `getIntegral`, welche das Integral  $F(x) = \int_0^x p(t)dt$  berechnen und als `return` Wert zurückgeben soll.