

**Präsenzaufgabe 3.1**

Überführen Sie Ihren Pseudocode der Zahlenraten-Aufgabe des letzten Blattes in ein funktionsfähiges Javaprogramm.

**Aufgabe 3.1** (*4 Punkte*).

Beschreiben Sie einen Algorithmus in Pseudocode für folgenden Primzahltest der Zahlen 1 bis 100:

Jede zu testende positive ganze Zahl  $n$  wird probeweise durch jede Zahl zwischen zwei und 100 geteilt und es wird geprüft, ob diese Division ohne Rest durchgeführt werden kann. Alle gefundenen Primzahlen sollen ausgegeben werden.

**Aufgabe 3.2** (*5 Punkte*).

Klammern Sie folgende Ausdrücke so, dass sie zu den Java Präzedenzregeln passend abgearbeitet werden.

$3 / 5 * - 2 == 3 + 5 - 2$

$false || 8 \% - 3 == 7 * 2 \&\& true$

**Aufgabe 3.3** (*4 Punkte*).

Welche Ausgaben produzieren folgende Zeilen und warum?

```
System.out.println(0.3 + 0.3 + 0.3);  
System.out.println('a' + 'b' + 'c' + "!");  
System.out.println(9/2);  
System.out.println("Rechnung: " + 3 + -1 + 5);
```

**Aufgabe 3.4** (9 Punkte).

Schreiben Sie ein Javaprogramm, das die Zahlen von 1 bis 100 auf der Konsole ausgibt. Allerdings soll immer, wenn eine Zahl durch 3 teilbar ist, diese Zahl durch `grumpy` ersetzt werden. Analog: ist eine Zahl durch 5 teilbar, wird `cat` ausgegeben.

Die Ausgabe von 10 bis 15 sähe folgendermaßen aus:

```
cat
11
grumpy
13
14
grumpycat
```