

# Informatik I: Einführung in die Programmierung

Prof. Dr. Peter Thiemann  
Tim Schulte, Christoph-Simon Senjak  
Wintersemester 2018/2019

Universität Freiburg  
Institut für Informatik

## Übungsblatt 3

Abgabe: Dienstag, 6.11.2018, 20:00 Uhr

### Aufgabe 3.1 (Boolsche Ausdrücke; Datei: `bool.txt`; 3 Punkte)

Beschreiben Sie die Wertemengen (alle möglichen Belegungen für  $x$  bzw.  $y$ ) für welche die folgenden Python-Funktionen `True` zurückgeben.

- (a) 

```
def foo(x: float) -> bool:
    return not x >= 35
```
- (b) 

```
def bar(x: int) -> bool:
    if x > 3 and x % 3 == 0:
        return True
```
- (c) 

```
def baz(x: int, y: float):
    if not x or y and x == 35:
        print("True")
```

### Aufgabe 3.2 (Schaltjahre; Datei: `leap.txt`; 5 Punkte)

Schreiben Sie eine Funktion `leapyear(year: int) -> bool` welche für ein gegebenes Jahr `year` berechnet, ob dieses Jahr ein Schaltjahr (im gregorianischer Kalender) ist oder nicht. Ein Jahr ist ein Schaltjahr, wenn das Jahr ganzzahlig durch 4 teilbar ist, ohne dabei ganzzahlig durch 100 teilbar zu sein, es sei denn, das Jahr ist außerdem durch 400 teilbar. Beispielsweise ist 1997 kein Schaltjahr, 1996 ist ein Schaltjahr. 1900 ist kein Schaltjahr, 2000 ist ein Schaltjahr.

### Aufgabe 3.3 (Schulnoten; Datei: `grading.txt`; Punkte: 4+3+3)

Im Folgenden geht es darum, einen Bewertungsschlüssel zur Verteilung von Schulnoten für eine Klausur mit 60 erreichbaren Punkten zu implementieren. Gültige Notenstufen sind 1.0, 1.5, ..., 5.5 und 6.0. Idee hierbei ist, dass Schülerinnen mit einer Punktzahl von 20% der erzielbaren Punkte (das sind 12 Punkte) oder weniger in jedem Fall die Note 6.0 erhalten, Schülerinnen mit einer Punktzahl von mindestens 95% (also 57 Punkten) in jedem Fall eine 1.0 erzielen. Ansonsten wird folgendes Verfahren angewendet: für eine Punktzahl  $12 < p < 57$  wird zunächst ein „exakter“ Notenwert mit der Formel  $-\frac{5}{45} \cdot (p - 57) + 1$  berechnet. Dieser wird dann zur nächstgelegenen Notenstufe gerundet.

- (a) Implementieren Sie die oben beschriebene Benotungsvorschrift als Funktion `lineargrade(p: float) -> float`, die bei Eingabe einer Punktzahl  $p \geq 0$  die zugehörige Note zurückgibt. Testen Sie Ihre Funktion an geeigneten Beispielen, also z.B. mit:

```
>>> abs(lineargrade(0) - 6.0) < 1e-10
True
>>> abs(lineargrade(36.5) - 3.5) < 1e-10
True
```

```
>>> abs(lineargrade(37) - 3.0) < 1e-10
True
>>> abs(lineargrade(60.0) - 1.0) < 1e-10
True
```

- (b) Implementieren Sie eine Funktion `passed(p: float) -> bool` welche für eine Punktzahl  $p$  genau dann `True` zurückgibt, wenn die Klausur nach obigem Bewertungsschlüssel mindestens eine Note von 4.0 erzielt.

```
>>> passed(0)
False
>>> passed(27.5)
False
>>> passed(28)
True
>>> passed(33)
True
```

- (c) Schreiben Sie die folgende Funktion so um, dass lediglich ein einziges `if`-Statement verwendet wird. Verwenden Sie hierzu mehrere `elif`-Anweisungen.

```
def mark(grade: float):
    if grade >= 6.0:
        print("F")
    else:
        if grade >= 5.0:
            print("E")
        else:
            if grade >= 4.0:
                print("D")
            else:
                if grade >= 3.0:
                    print("C")
                else:
                    if grade >= 2.0:
                        print("B")
                    else:
                        print("A")
```

**Aufgabe 3.4** (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet03` eine Textdatei `erfahrungen.txt` an. Notieren Sie in dieser Datei kurz Ihre Erfahrungen beim Bearbeiten der Übungsaufgaben (Probleme, Bezug zur Vorlesung, Interessantes, benötigter Zeitaufwand, etc.).