

# Aufgaben Programmierkurs

## Übungsblatt 3



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

von Uli Fahrner & Dennis Albrecht

Wintersemester 2014/2015

### Aufgabe 1 Multiple Choice

Kreuze zu jeder Antwort an, ob sie zutrifft (w) oder nicht (f).

#### Aufgabe 1.1 Arrays (Stufe 1)

w f

- ☐ ☐ Ein bestehendes Array kann in seiner Länge verändert werden.
- ☐ ☐ meinArray[0] liefert einen gültigen Wert.
- ☐ ☐ meinArray[meinArray.length] liefert einen gültigen Wert.
- ☐ ☐ Eine foreach-Schleife gibt die Elemente eines String-Arrays in alphabetischer Reihenfolge aus.

#### Aufgabe 1.2 Funktionen (Stufe 2)

w f

- ☐ ☐ Eine Funktion mit Rückgabewert muss auch einen Wert zurückgeben.
- ☐ ☐ Eine void-Funktion kann Werte zurückgeben.
- ☐ ☐ Eine Funktion kann auch mehrere Werte zurückgeben.

### Aufgabe 2 Theoriefragen

#### Aufgabe 2.1 Funktionsköpfe (Stufe 1)

Welche der folgende Funktionsköpfe sind gültig.

- |  |  |
|--|--|
| a) <code>public static void f1()</code>        | d) <code>public static int f4(int a, int b)</code>               |
| b) <code>public static f2()</code>             | e) <code>public static String f5(String, char)</code>            |
| c) <code>public static boolean f3(x, y)</code> | f) <code>public static String[] f6(String r, String... c)</code> |

---

## Aufgabe 3 Arrays

---

### Aufgabe 3.1 Ein Array anlegen (Stufe 1)

---

- a) Schreibe ein Programm, das zuerst ein Array für 10 Elemente vom Typ `char` allokiert. Anschließend sollen diese 10 Elemente mit Benutzereingaben gefüllt werden. D.h. der Benutzer soll in der Lage sein nach und nach Zeichen einzugeben, welche in dem Array abgelegt werden. Achte dabei darauf, dass die Größe des Arrays nicht überschritten wird. Gib die gespeicherten Zeichen anschließend nacheinander aus.
- b) Teste dein Programm und gib der Reihe nach die folgenden Zeichen ein:  
l, n, f, o, r, m, a, t, i, k.

---

## Aufgabe 4 Funktionen

---

### Aufgabe 4.1 Taschenrechner Reloaded (Stufe 1)

---

In der vorigen Übung sollte ein simpler Taschenrechner programmiert werden. Nun wollen wir diesen Taschenrechner erweitern. Schreibe zuerst jeweils eine Funktion für jede der angebotenen Grundrechenarten (`public static int plus(3, 5)` soll beispielsweise 8 liefern). Es ist dir überlassen, ob du bei der Division eine Ganzzahl- oder Fließkommadivision machst. Im Anschluss sollen die Rechenanweisungen in der `switch-case`-Anweisung oder der `if`-Struktur durch Funktionsaufrufe ersetzt werden. Vergiss nicht, das Ergebnis auszugeben.

---

### Aufgabe 4.2 Taschenrechner Reloaded II (Stufe 1)

---

Erweitere den Taschenrechner nun noch um einen weiteren Operator. Dabei soll auch das Potenzieren möglich sein. Dabei gilt  $a^n = a \cdot a \cdot a \cdot \dots \cdot a$  (d.h.  $a$  wird  $n$ -mal mit sich selbst malgenommen). Zum Aufruf der Potenzier-Funktion soll der Benutzer `^` (das kleine „Dach“ links oben auf der Tastatur) eingeben.

---

### Aufgabe 4.3 Tabellen Ausgabe (Stufe 2)

---

Schreibe ein Programm, welches eine Tabelle von Potenzen ausgibt.

n	$n^2$	$n^3$	...
1	1	1	
2	4	8	
3	9	27	
⋮			

Dabei soll die Größe (also die höchste Potenz und die höchste Basis) variabel, also von einer Benutzereingabe abhängig sein. Zum Potenzieren kann die Funktion des Taschenrechners wiederverwendet werden. Wenn beide Klassen im selben Ordner liegen, kann die Potenz-Funktion ähnlich der Input-Funktionen direkt aus dem neuen Tabellen-Programm genutzt werden. Je nach vergebenen Namen kann der Funktionsaufruf etwa `Taschenrechner.potenz(basis, exponent)` lauten.

---

### Aufgabe 4.4 Funktionen mit beliebig vielen Parameters (Stufe 2)

---

Schreibe eine Funktion die beliebig viele Parameter vom Type `double` übergeben bekommt und diese einfach nacheinander ausgibt.

---

## Aufgabe 5 Arrays und Funktionen

---

### Aufgabe 5.1 Summe und Durchschnitt eines Arrays (Stufe 3)

---

Ergänze im folgende Code die fehlenden Codestellen damit die beiden Funktionen die Summe und den Durchschnitt errechnen.

```
1 public class Arrayfunktionen {
2     public static void main(String[] args) {
3         int array[] = {2,4,6,8,10,12,14,16,18};
4         System.out.println(summe(array));
5         System.out.println(durchschnitt(array));
6     }
7
8     public static int summe(int[] werte) {
9         //To-Do Rechne die Summe des Arrays aus
10    }
11
12    public static int durchschnitt(int[] werte) {
13        //To-Do Berechne den Durchschnitt mit Hilfe der Funktion summe
14    }
15 }
```

---

### Aufgabe 5.2 Summe und Durchschnitt eines Arrays II (Stufe 2)

---

Schreibe deine Funktionen aus dem vorigen Aufgabenteil nun so um, dass sie kein Array entgegennehmen sondern beliebig viele Integerwerte. Was fällt im Bezug auf die übergebenen Zahlen auf.