



Praktikum zu
**Einführung in die Informatik für
LogWings, WiMas und MedPhys**
Wintersemester 2021/22

Übungsblatt 8
Besprechung:
13.–17.12.2021
(KW 50)

Vorbereitende Aufgaben

Aufgabe 8.1: Manuelle Sortierung

Gegeben ist folgendes Array:

| | | | | | |
|---|----|---|---|----|---|
| 7 | 12 | 3 | 2 | 21 | 9 |
|---|----|---|---|----|---|

In der Vorlesung (Kapitel 5.1) haben Sie den Algorithmus **Selectionsort** zum Sortieren von Arrays kennengelernt. Sortieren Sie das Array nun aufsteigend nach diesem Algorithmus verfahren. Notieren Sie dabei **jede** Tauschoperation:

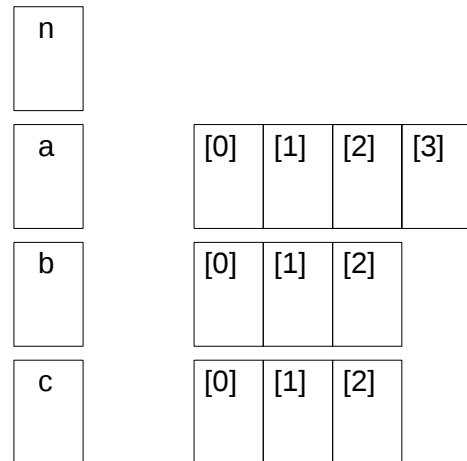
| | | | | | |
|--|--|--|--|--|--|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

Präsenzaufgaben

Aufgabe 8.2: Speicherverwaltung für Felder

In dieser Aufgabe wollen wir uns mit der Speicherverwaltung im Zusammenhang mit Arrays beschäftigen. Betrachten Sie das folgende Programmfragment auf Seite 2. Es deklariert und initialisiert eine ganzzahlige Variable **n** und drei Felder von ganzen Zahlen **a**, **b** und **c**. Gehen Sie den Programmtext zeilenweise durch und tragen Sie in das nebenstehende Diagramm ein, wie die einzelnen Speicherzellen im Laufe des Programms belegt und geändert werden.

```
1 int n;  
2 int[] a;  
3 a = new int[4];  
4 for (int i = 0; i < a.length; i++) {  
5     a[i] = i;  
6 }  
7 n = a.length - 1;  
8 int[] b;  
9 b = new int[n];  
10 for (int i = 0; i < b.length; i++) {  
11     b[i] = a[i] + 10;  
12 }  
13 int[] c = new int[b.length];  
14 c = b;
```



Aufgabe 8.3: Arrays als Parameter

In dieser Aufgabe wollen wir lernen, Arrays als Parameter zu nutzen. Legen Sie dazu eine Klasse **UseArrays** an.

- Schreiben Sie eine Funktion **printArray**, die den Inhalt eines übergebenen **int**-Array ausgibt. Der Funktionskopf soll wie folgt aussehen: **public static void printArray(int[] array)**
- Implementieren Sie eine Funktion mit dem Namen **swap**, die ein **int**-Array, sowie zwei Indizes entgegen nimmt und die Elemente im Array an den entsprechenden Indizes miteinander tauscht. Die Funktion besitzt keine Rückgabe.
- Verwenden Sie anschließend die Funktion **swap** in einer **main**-Methode, um das erste mit dem letzten Element eines Arrays zu tauschen. Geben Sie das Array vor und nach dem Aufruf von **swap** mit der Funktion **printArray** aus.
- Was fällt Ihnen auf, wenn Sie das Ergebnis mit den Erkenntnissen zu primitiven Datentypen als Parameter aus Aufgabe 6.5 vergleichen?

Aufgabe 8.4: Bibliothek erstellen

In dieser Aufgabe wollen wir eine Klasse programmieren, mit der Sie Arrays mit verschiedenen Eigenschaften erzeugen können. Erstellen Sie eine neue Klasse namens **ArrayGenerator**. Diese Klasse soll keine **main**-Methode erhalten! Implementieren Sie die folgenden beiden Funktionen in dieser Klasse.

- a) Die Funktionen **generateAscendingArray** soll ein **int**-Array der Länge **length** erzeugen und es aufsteigend mit den Werten von 0 bis **length**−1 befüllen.
- b) Die Funktion **generateRandomArray** soll ein zufällig gefülltes Array mit der Länge **length** erzeugen.

Dafür müssen Sie die Klasse **Random** aus dem Paket **java.util** importieren. Verwenden Sie dafür: **import java.util.Random;** noch vor der Definition der Klasse. Anschließend können Sie mit **Random randomNumbers = new Random();** einen neuen Zufallsgenerator erzeugen und danach mit **randomNumbers.nextInt(n)** eine zufällige Zahl aus dem halboffenen Intervall $[0, n)$ erhalten. Verwenden Sie als n die Länge des erzeugten Arrays.

Aufgabe 8.5: Bibliothek verwenden

In dieser Aufgabe wollen wir nun die Funktionen der soeben definierten Klasse verwenden. Eine Klasse ohne **main**-Methode, wie Ihre **ArrayGenerator**-Klasse, nennt man auch **Bibliothek**. Bibliotheken besitzen meistens sehr hilfreiche Funktionen. Diese nützlichen Funktionen aus der eben geschriebenen Bibliothek wollen wir nun in der Klasse **UseArrays** benutzen. Anschließend erweitern wir die Klasse **UseArrays** noch um zwei weitere Funktionen.

- a) Verwenden Sie in der **main**-Methode der Klasse **UseArrays** die soeben von Ihnen definierten Funktionen, um ein aufsteigendes und ein zufälliges Array der Länge 10 auszugeben.
- b) Erweitern Sie nun die Klasse **UseArrays** um die Funktion **arrayMin**, die das Minimum eines übergebenen **int**-Arrays zurückgibt. Rufen Sie diese Funktion für ein zufällig generiertes Array auf.
- c) Schreiben Sie die Funktion **average**, die den Mittelwert eines übergebenen **int**-Arrays berechnet und als **double** zurückgibt. Rufen Sie auch diese Funktion für ein zufällig generiertes Array auf.