

Übungsblatt 4

Aufgabe 4.1: Arrays

Gegeben ist folgender Teil eines Java-Programms, das ein Array mit zufälligen Zahlen zwischen 1 und 49 füllt und diverse Auswertungen darüber durchführt.

Hinweis aus der Dokumentation <https://docs.oracle.com/javase/7/docs/api/java/util/Random.html>

```
public int nextInt(int n)
Returns a pseudorandom, uniformly distributed int value between 0 (inclusive) and the specified value (exclusive)
```

Beispiel: <http://www.javapractices.com/topic/TopicAction.do?id=62>

Gegebener Code:

```
import java.util.Random;

public class Lottozahlen {

    static int[] zahlen = new int[100];
    static int[] doppelteZahlen = new int[50];

    public static void main(String[] args) {
        // initialisiere mit Zufallszahlen zwischen 1 und 49
        initArray();
        // Ausgabe eines Arrays: Index und Wert
        ausgabeArray(zahlen);
        // Ausgabe von Maximal-, Minimal- und Durchschnittwert
        System.out.println("Maximalwert: " + maxWert());
        System.out.println("Minimalwert: " + minWert());
        System.out.println("Mittelwert: " + midWert());

        // Suche nach doppelten Zahlen
        sucheDoppelte();
        // Ausgabe wie oft die Zahlen zwischen 1 und 49 bei den 100
        // Zufallszahlen vorkommt
        System.out.println("Auswertung");
        ausgabeArray(doppelteZahlen);

        // Suche nach dem erstmaligen Vorkommen einer doppelten Zahl
        // Ausgabe der Position
        System.out.println("Erste doppelte Zahl bei: "
            + indexErsteDoppelte(zahlen));
    }
}
```

Implementieren Sie die fehlenden Methoden.

Empfohlenes Vorgehen: Eine Methode nach der anderen umsetzen. In der main-Methode die restlichen Methodenaufrufe auskommentieren.

Aufgabe 4.2: Messwerte

Jede Stunde misst eine Informatik-Student*in die Temperatur auf dem Balkon. Sie möchte die Werte automatisch auswerten, um herauszufinden, wann es am heißesten und wann es am kältesten war.

a) Die Stunden sollen einfach in 1, 2, 3, ..., 24 in einem int-Array gespeichert werden. Die zugehörigen Messwerte sollen in einem double-Array gespeichert werden. Sie können dieses Array einfach im Quellcode mit festen Werten initialisieren oder Random (siehe unten) verwenden.

Eine Methode findMax() soll den maximalen Wert ermitteln. Eine weitere Methode soll den minimalen Wert ermitteln. Beide sollen ausgegeben werden.

b) Es wäre natürlich schöner, wenn man zusätzlich noch die Stunde ausgegeben bekommt. Z.B.:

„Um 13 Uhr war es am wärmsten mit 27,3 °C“.

„Um 6 Uhr war es am kältesten mit 5,7 °C.“

Wie können Sie dies erreichen? Z.B. könnten weitere Methoden einen dieser Strings zurückgeben. Es gibt auch andere Lösungen.

c) Für „Experten zum Knobeln“

Die Messwerte kommen von einer Messtation, die pro Tag zu einem 24 Zeitpunkten zwei Werte sendet:

- Zeitstempel / Timestamp als Unix-TimeStamp als long-Werte

- Messwert als double-Wert

Die Werte sollen in einem zweidimensionalen long-Array gespeichert werden. Dazu werden die Messwerte mit 1000 Multipliziert und als long gespeichert.

Die Werte eines Tages sollen dann untereinander ausgegeben werden. Dabei wird der Unix Timestamp in ein Datumsformat gewandelt und die Temperatur wieder mit drei Nachkommastellen angegeben.

Hinweis: z.B. <https://stackoverflow.com/questions/17432735/convert-unix-time-stamp-to-date-in-java/27319755>