

# Übungsblatt 1

## Propädeutikum Informatik

May 8, 2018

Alle Aufgaben auf diesem Blatt können in einer Klasse gelöst werden. Allerdings soll, wie in der Veranstaltung am Freitag eingeführt, die jeweilige Funktionalität in eigene Methoden ausgelagert werden und nur noch die Tests in der Main-Methode durchgeführt werden.

### 1 Minimum und Maximum

Schreibe jeweils eine Methode **min(int a, int b)**, die zwei Integer Zahlen als Argumente erhält und das Minimum zurück gibt. Und eine Methode **max(int a, int b)**, die zwei Integer Zahlen als Argumente erhält und das Maximum zurück gibt.

### 2 Division

In der Übung haben wir die Methoden Addition und Quadrieren geschrieben. Schreibe jetzt eine Methode **div(double a, double b)**, die zwei Double Zahlen als Eingabe bekommt und das Ergebnis  $a/b$  zurück gibt. Denk daran dass Division durch 0 nicht erlaubt ist. Die Methode soll in diesem Fall den Wert -0.0 zurück geben.

### 3 Mittelwert

Schreibe eine Methode **mittelwert(double a, double b, double c)**, die drei Double Zahlen als Eingabe bekommt und den Mittelwert also  $(1/3) \cdot (a+b+c)$  zurückgibt.

### 4 isIn Methode

Am Freitag haben wir Arrays als zusammengesetzte Datenstruktur kennengelernt. Schreibe eine Methode **isIn(int[] array, int a)**, welche als Eingabe ein unsortiertes Integer Array beliebiger Länge und einen Integer a erhalten soll. Die Methode soll prüfen, ob die Zahl a in dem Array enthalten ist und entsprechend true oder false zurückgeben.

## 5 Minimum und Maximum

In Aufgabe 1 haben wir zwei Methoden geschrieben, welche das Maximum bzw. Minimum von zwei Zahlen berechnet haben. Dies soll nun auf beliebig viele Zahlen erweitert werden. Schreibe zwei neue Methoden Minimum und Maximum welche nun ein Integer Array beliebiger Länge übergeben bekommen und davon das Minimum bzw. Maximum zurück geben.

## 6 Geometrisches Mittel

In dieser Aufgabe soll eine Funktion geschrieben werden, die das geometrische Mittel der Werte eines Double Arrays zurückgibt. Das geometrische Mittel von  $n$  Zahlen  $x_1, \dots, x_n$  ist definiert durch  $(x_1 * x_2 * \dots * x_n)^{(1/n)}$ .

## 7 Ackermann Funktion

Die Ackermann Funktion ist eine vor allem in der theoretischen Informatik oft benutzte rekursive Funktion. Die Ackermann Funktion hat folgende Bildungsvorschrift:

$$ack(a, b) = \begin{cases} b + 1 & \text{falls } a = 0 \\ ack(a - 1, 1) & \text{falls } a > 0 \text{ und } b = 0 \\ ack(a - 1, ack(a, b - 1)) & \text{sonst} \end{cases} \quad (1)$$

Versuche eine Methode **ack(int a, int b)** rekursiv in Java zu implementieren.