



Übung 9

Aufgabe 1 :

Geben Sie drei Methoden an, die für ein Zahlenvektor $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ mit $n \geq 0$ die folgenden Werte berechnen (für das harmonische Mittel muss für alle i gelten: $x_i \neq 0$):

1. arithmetischen Mittelwert $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
2. harmonischen Mittelwert $\frac{n}{\sum_{i=1}^n \frac{1}{x_i}}$
3. Standardabweichung $\sigma(x) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$

Für $n = 0$ liefern alle Methoden den Wert 0. Geben Sie weiterhin eine Methode an, die für beliebig viele Eingabezahlen von der Kommandozeile in einem String-Feld, dessen String-Inhalte Zahlenwerte darstellen, einen Feld mit double-Zahlen anlegt und dieses als Resultat der Methode liefert.

Rufen Sie in `main()` die Umwandlungsmethode auf mit dem main-Argument `args`. Rufen Sie anschließend mit dem resultierenden double-Feld die drei statistischen Auswertemethoden auf und lassen sich das Ergebnis jeden Aufrufs auf dem Bildschirm ausgeben.

Hinweis: `Math.sqrt`

Beispielaufruf:

```
java Statistik 1 2 3 3 4 5
arithmetischer Mittelwert: 3.0
harmonischer Mittelwert: 2.18978102189781
Standardabweichung : 1.5811388300841898
```

Aufgabe 2 (1 Punkt) :

Dies ist eine Praktomataufgabe!

Geben Sie in einer Klasse `Zahlendarstellung` eine Methode `public static int[] ermittleZiffern(int x, int b)` an, die zu einer Zahl $x \geq 0, x \in \mathbb{N}$ und einer Basis b mit $b \in \mathbb{N}, b > 1$ alle Ziffern der Zahl x zur Basis b bestimmt und in einem int-Feld als Resultat liefert. Im Resultatfeld stehen an den Stellen mit kleineren Indizes die niederwertigsten Stellen von x , in `feld[0]` steht also die niederwertigste Ziffer. Ist die Basis b größer als 10, so nehmen Sie als 'Ziffern' jenseits der 9 die Zifferwerte $10, \dots, b-1$ (siehe die Ziffer 13 im letzten Beispiel unten). Beachten Sie, dass $x = 0$ möglich ist.

Sie dürfen keine String-Operationen verwenden! Hinweise:

- ganzzahlige Division und Modulo
- Betrachten Sie den Fall $x = 0$ als Sonderfall

Beispiele:

- `ermittleZiffern(4711, 10)` liefert das Feld $\{1,1,7,4\}$
- `ermittleZiffern(4711, 16)` liefert das Feld $\{7,6,2,1\}$
- `ermittleZiffern(4711, 2)` liefert das Feld $\{1,1,1,0,0,1,1,0,0,1,0,0,1\}$
- `ermittleZiffern(1234, 16)` liefert das Feld $\{2,13,4\}$

Beachten Sie, dass das Feld genauso groß sein soll, wie der Wert an Ziffern benötigt. Dies müssen Sie also für eine Zahl `x` erst ermitteln.

Hinweis: Im Feld werden laut Vorgabe die niederwertigsten Ziffern an der niederwertigsten Positionen im Feld gespeichert. Wenn Sie *testweise* ihr Ergebnisfeld auf dem Bildschirm ausgeben wollen, so macht es Sinn, den Inhalt in umgekehrter Reihenfolge auszugeben, die höchstwertigen Ziffern zuerst, also der normalen Darstellungsweise.

Aufgabe 3 :

Basierend auf dem Abstrakten Datentyp *Folge* mit dessen vorgegebenen Funktionen, definieren Sie eine rekursive Funktion $istenthalten : M^* \times M \rightarrow \mathbb{B}$, die für eine Folge f und ein Element a der Grundmenge mit $istenthalten(f, a)$ feststellt, ob das Element a in der Folge f enthalten ist. Sie können Elemente der Basismenge M mit dem Operator $=$ direkt vergleichen.

Testen Sie Ihre Lösung an folgendem Beispielaufwurf: $istenthalten(< a, b, c, d >, c)$. Geben Sie dabei alle Zwischenschritte in der Berechnung des Beispiels an.

Aufgabe 4 :

Geben Sie eine Funktion $exchangeAll : M^* \times M \times M \rightarrow M^*$ an, wobei $exchangeAll(f, a, b)$ alle Vorkommen von a in f durch b ersetzen soll. Sie können Elemente der Basismenge M mit dem Operator $=$ direkt vergleichen.

Testen Sie ihre Lösung an folgenden Beispielaufwurf: $exchangeAll(< 3, 4, 4, 5 >, 4, 6)$. Geben Sie dabei alle Zwischenschritte in der Berechnung des Beispiels an. Das Ergebnis muss $< 3, 6, 6, 5 >$ sein.

Aufgabe 5 :

Geben Sie zu folgender Problemstellung eine adäquate Klasse mit Namen **Studierender** in Java an, die zum jetzigen Zeitpunkt nur relevante Instanzvariablen enthalten soll. Diese Klasse wird in der nächsten Übung erweitert. Überlegen Sie sich, was jeweils geeignete Typen für die Instanzvariablen sind.

Ein Student / eine Studentin wird beschrieben durch einen Vornamen, einen Nachnamen, eine Matrikelnummer, die einzelnen Noten der bis jetzt bestandenen Prüfungen (auch 0 Prüfungen ist z.B. für Erstsemester möglich) sowie der Notendurchschnitt über alle diese bestandenen Prüfungen (dessen Wert als Spezialfall 0 ist, wenn bis jetzt noch keine Prüfung bestanden wurde). Mögliche Noten zu einer Prüfung sind 1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0, 5.0.

Aufgabe 6 :

Geben Sie zu folgender Problemstellung eine adäquate Klasse **Musikstueck** in Java an, die zum jetzigen Zeitpunkt nur relevante Instanzvariablen enthalten soll. Überlegen Sie sich, was jeweils geeignete Typen für die Instanzvariablen sind.

Ein elektronisches Musikstück soll in Java beschrieben werden. Zu einem Musikstück gehören n Interpreten ($n \in \mathbb{N}, n > 0$) und ein Titel. Weiterhin hat ein solches Musikstück m Töne ($m \in \mathbb{N}, m \geq 0$), wobei jeder Einzelton durch eine ganze Zahl dargestellt wird. Zusätzlich hat ein Musikstück eine Gesamtbewertung, ein ganzzahliger Wert zwischen 1-5, der auf n Einzelbewertungen beruht. Diese Einzelbewertungen selbst werden aber *nicht* gespeichert, sondern nur die Gesamtanzahl der Bewertungen und der Durchschnitt über alle Bewertungen. Wurden noch keine (Einzel-)Bewertungen abgegeben, so ist die Gesamtbewertung als Spezialfall 0.