

Aufgabenblatt zur 1. Übung

Zeitraum: 19.04. bis 23.04.2010

Achtung: Die Ersatzvorlesung für Freitag, dem 23.04.10, findet am Mittwoch, dem 21.04.10, in der 5. DS im POT/0081 (siehe auch Vorlesungsankündigung auf der Homepage) statt.

1. Aufgabe (AGS 11.1):

Programmieren Sie eine Funktion `f` in Haskell, die eine Liste von `Int`-Zahlen als Eingabe nimmt und eine Liste von `Int`-Zahlen mit folgenden Eigenschaften liefert:

- Die Ergebnisliste soll nur Zahlen größer Null enthalten.
- Die Zahlen der Ergebnisliste erscheinen in umgekehrter Reihenfolge bezüglich der Zahlen der Ausgangsliste.

2. Aufgabe (AGS 11.2):

Eine Liste `s = [e1, e2, ...]` bestehe aus Funktionswerten $f(i) = e_i$ mit $i \in \mathbb{N}^+$.

`f` sei wie folgt definiert:

$$f(1) = 1, f(2) = 1, f(3) = 1$$

$$f(i + 3) = f(i) + f(i + 1) + f(i + 2) \text{ für } (i \geq 1)$$

- Geben Sie eine Haskell-Funktion `f1` an, die die oben definierte Liste erzeugt.
- Schreiben Sie eine Haskell-Funktion `p`, die von der Liste `s` das n -te Element ermittelt.
- Stellen Sie die Operationsfolgen für die Ausgabe des zweiten Elementes von `s` bei Benutzung der Auswertungsstrategien „call by value“, „call by name“ und „call by need“ (lazy evaluation) gegenüber.

3. Aufgabe (AGS 11.4):

- Programmieren Sie eine Funktion `filter :: [Int] -> [Int]`, die aus einer Liste von Zahlen diejenigen herausstreicht, welche ungerade sind.
- Programmieren Sie eine Funktion `count :: [Int] -> Int`, die eine Liste aus (a) als Argument nimmt und die Anzahl der darin enthaltenen Elemente liefert.
- Programmieren Sie eine Funktion `check :: [Int] -> Bool`, die genau dann den Wahrheitswert `True` liefert, wenn die Eingabeliste mindestens 5 gerade Zahlen enthält. Verwenden Sie dazu die Funktionen `filter` und `count`.

4. Aufgabe (AGS 11.3*):

Gegeben sei der Listentyp $[l_1, l_2, \dots, l_n]$ mit l_1, l_2, \dots, l_n jeweils vom Typ `[Int]`. Es soll von jeder Liste l dieses Typs die Länge der längsten Liste, also $\max\{\text{length}(l_i) \mid 1 \leq i \leq n\}$, als Ergebnis berechnet werden.

- Geben Sie ein Beispiel für eine Liste dieses Listentyps an, und nennen Sie das zugehörige Ergebnis.
- Schreiben Sie in Haskell eine Funktion `max_length :: [[Int]] -> Int`, die diese Aufgabe erfüllt, und geben Sie abschließend einen Funktionsaufruf an. Wenn Sie in `max_length` Hilfsfunktionen nutzen, müssen Sie für diese die Typdeklarationen und den Programmcode aufschreiben. Hilfestellung: Ermitteln sie zunächst aus der Liste von Listen die Liste der Längen dieser Listen.