

ProInformatik II: Funktionale Programmierung

6. Übungsblatt (7. Tag)

1. Aufgabe

Definieren Sie eine Haskell-Funktion **flatten** :: [[a]] -> [a], welche eine Liste von Listen zu einer Liste kombiniert.

Anwendungsbeispiel:

flatten ["abc", "bcd", "abc"] => "abcbcdabc"

- a) Schreiben Sie zuerst Ihre Definition mit Hilfe der **foldr**-Funktion.
- b) Schreiben Sie eine zweite Definition mit der **foldl**-Funktion.
- c) Welche der beiden Lösungen ist besser? Warum?

2. Aufgabe

Definieren Sie die Summe der Quadrate aller Zahlen zwischen **1** und **n** unter Verwendung der **foldr** und **map**-Funktionen.

3. Aufgabe

Benutzen Sie die **foldr**-Funktion, um eine eigene **min**-Funktion zu definieren.

4. Aufgabe

Definieren Sie unter Verwendung der **foldl**-Funktion noch einmal die **bin2dec** Funktion aus der Vorlesung, die als Eingabe eine Liste von Bits bekommt und daraus die entsprechende Dezimal-Zahl berechnet.

5. Aufgabe

Ein Element einer Liste von **n** Objekten stellt die absolute Mehrheit der Liste dar, wenn das Element mindestens $\left(\frac{n}{2} + 1\right)$ mal in der Liste vorkommt. Definieren Sie

eine **majority**-Funktion, die mit linearem Aufwand das Majority-Element der Liste findet, wenn eines existiert oder sonst **Nothing** zurückgibt.

Die Signatur der Funktion soll wie folgt aussehen:

majority :: (Eq a) => [a] -> **Maybe** a