

# Übungen zu Funktionaler Programmierung

## Übungsblatt 9

**Ausgabe:** 9.12.2016, **Abgabe:** 16.12.2016 - 12:00 Uhr

**Aufgabe 9.1** (3 Punkte) Schreiben Sie Traversierungsfunktionen für Bintree.

1. Die Funktion `preorderB` soll die Knoten des Baumes als Liste in Hauptreihenfolge (pre-order) ausgeben.
2. Die Funktion `postorderB` soll die Knoten des Baumes als Liste in Nebenreihenfolge (post-order) ausgeben.

**Aufgabe 9.2** (3 Punkte) Bestimmen Sie die Typen der folgenden Ausdrücke. Die Aufgabe soll mithilfe von Typinferenzregeln gelöst werden.

1. `zipWith (+)`
2. `\x -> \f -> f x`

**Aufgabe 9.3** (6 Punkte)

1. Modellieren folgende Eigenschaften mit Datentypen.
  - Eine Bank führt eine Liste von Konten.
  - Ein Konto hat einen Kontostand und einen Kunden als Besitzer.
  - Für einen Kunden werden die Daten Vorname, Name und Adresse (String) gespeichert.

2. Legen Sie eine Beispielbank an mit mindestens zwei Konten.

3. Definieren Sie folgende Funktionen. Benutzen Sie für ID den Typ `Int` (`type ID = Int`).

**credit :: Int -> ID -> Bank -> Bank** Addiert den angegebenen Betrag auf das angegebene Konto. Hinweis: Schauen Sie sich die Funktion `updList` an.

**debit :: Int -> ID -> Bank -> Bank** Subtrahiert den angegebenen Betrag von dem angegebenen Konto.

**transfer :: Int -> ID -> ID -> Bank -> Bank** Überweist den angegebenen Betrag vom ersten Konto auf das zweite.