

Übungen zu Funktionaler Programmierung

Übungsblatt 5

Ausgabe: 17.11.2017, **Abgabe:** 24.11.2017 – 16:00 Uhr, **Block:** 3

Aufgabe 5.1 (2 Punkte) *Unendliche Listen*

Schreiben Sie die angegebene Liste `solutions :: [(Int, Int, Int)]` so um, dass sie *alle* Lösungen der Gleichung $2x^3 + 5y + 2 = z^2$ enthält. Die Liste wird dadurch unendlich lang.

```
solutions :: [(Int, Int, Int)]
solutions = [ (x,y,z)
  | z <- [0..100]
  , y <- [0..100]
  , x <- [0..100]
  , 2*x^3 + 5*y + 2 == z^2
  ]
```

Aufgabe 5.2 (3 Punkte) *Zahlen als Datentypen*

Diese Aufgabe bezieht sich auf die in der Vorlesung vorgestellten rekursiven Datentypen `Nat`, `Int'` und `PosNat`.

- Definieren Sie eine Konstante `drei = 3` für den Datentyp `Int'` in Haskell.
- Erweitern Sie die Datentypen für Zahlen um einen Datentyp für rationale Zahlen. Basieren Sie den Datentyp nur auf den Datentypen `Nat`, `Int'` und `PosNat`.
- Definieren Sie eine Konstante $c = -\frac{3}{2}$ für Ihren Datentyp in Haskell.

Aufgabe 5.3 (4 Punkte) *Rekursive Datentypen*

Definieren Sie folgende Haskell-Funktionen.

- `natLength :: [a] -> Nat`, wie `length` für den Datentyp `Nat`.
- `natDrop :: Nat -> [a] -> [a]`, wie `drop` für den Datentyp `Nat` anstatt `Int`.
- `colistIndex :: Colist a -> Int -> a`, wie `(!!)` für `Colist a` anstatt `[a]`.
- `streamTake :: Int -> Stream a -> [a]`, wie `take` für `Stream a` anstatt `[a]`.

Aufgabe 5.4 (3 Punkte) *Modellierung*

Gegeben seien folgende Datentypen:

```
type ID = Int
data Bank = Bank [(ID,Account)] deriving Show
data Account = Account { balance :: Int, owner :: Client } deriving Show
data Client = Client
  { name :: String
  , surname :: String
  , address :: String
  } deriving Show
```

Definieren Sie folgende Funktionen.

- a) `credit :: Int -> ID -> Bank -> Bank` – Addiert den angegebenen Betrag auf das angegebene Konto.
- b) `debit :: Int -> ID -> Bank -> Bank` – Subtrahiert den angegebenen Betrag von dem angegebenen Konto.
- c) `transfer :: Int -> ID -> ID -> Bank -> Bank` – Überweist den angegebenen Betrag vom ersten Konto auf das zweite.