

Tinpro01-7 Practicumopdracht 3

W. Oele

14 maart 2016

Inleiding

In deze opdracht hou je je met verschillende zaken bezig:

- Werken met bestaande bibliotheken.
- Partieel parameteriseren.
- Oefenen met diverse ingebouwde functies zoals `filter`, `map`, de `fold` functies, etc.

Opdracht 1a : partieel parameteriseren

Schrijf de functie:

```
differentieer :: (Double->Double)->Double->Double->Double
differentieer f p x
```

Deze functie differentieert de functie f numeriek in punt x met een precisie p .

- Hint 1: bestudeer het differentiequotiënt.
- Hint 2: kies voor p een klein getal, maar niet te klein, bijv. $\frac{1}{10.000}$

Opdracht 1b : partieel parameteriseren

Schrijf de functie:

```
integreer :: (Double->Double)->Double->Double->Double->Double
integreer f a b p
```

Deze functie integreert de functie f op het interval a, b met een precisie p .

- Hint 1: bestudeer wat men verstaat onder Riemannintegratie.
- Hint 2: kies voor p een klein getal, maar niet te klein, bijv. $\frac{1}{10.000}$

Opdracht 2: werken met bibliotheken

Zoek op internet naar de module `Data.List`. Deze is te vinden in de `hackage` database. Importeer deze module in je programma. Functies kunnen ook worden opgezocht in de `hackage` database m.b.v. een daarvoor geschikte zoekmachine zoals <http://www.haskell.org/hoogle/>.

Schrijf de functie:

```
dubbelen :: [a] -> [a]
dubbelen s
```

Deze functie heeft een lijst als invoer en levert als uitvoer een lijst met uitsluitend die elementen die meer dan één keer voorkomen in de lijst. Voorbeeld:

```
ghci>dubbelen "aabbbcdeeeef"
ghci>"abe"
```

Gebruik de functies uit `Data.List` om bovenstaande functie op te bouwen.

Opdracht 3: hogere orde functies

Uit het eerste jaar is bij de module kansrekening het pokerspel doorgerekend:

Poker 5 stenen met gelijke ogen

Four of a kind 4 stenen met gelijke ogen

Three of a kind 3 stenen met gelijke ogen

Full house 3 stenen met gelijke ogen en nog eens twee stenen met gelijke ogen, bijv. 44554

Two pair Twee paar met gelijke ogen, bijv. 35453

One pair Een paar met gelijke ogen, bijv. 12344

Straight De reeks 1,2,3,4,5 of 2,3,4,5,6

Bust De overgebleven worpen

Schrijf een aantal functies, waarmee je de kansen kunt berekenen op bovenstaande uitkomsten. Gebruik naar eigen inzicht functies zoals `map`, `filter`, `any`, `all`, `foldr`, `foldl`, etc. om uit een lijst met (lijsten van) *alle* mogelijke worpen alleen die worpen te halen die een three of a kind, poker, etc. vertegenwoordigen.

- Hint: Probeer deze opdracht *niet* op te lossen door individuele functies te schrijven voor poker, full house, etc. Schrijf, in plaats daarvan, een kleine “bibliotheek” van functies die je kunt hergebruiken bij het oplossen van meerdere varianten.

Wat suggesties voor enkele functies:

De lijst met lijsten van iedere worp bouwen:

```
s = [1..6]
stenen = [[a,b,c,d,e] | a<-s,b<-s,c<-s,d<-s,e<-s]
```

Onderstaande functie retourneert het aantal voorkomens van c in een lijst:

```
count :: Integer -> [Integer] -> Integer
count c [] = 0
count c (x:xs)
  | c==x = 1 + (count c xs)
  | otherwise = count c xs
```

Onderstaande functie converteert een lijst in een aantal tuples met voorkomens:

```
convert list = ([a,b,c,d,e,f],list) where
  a = count 1 list
  b = count 2 list
  c = count 3 list
  d = count 4 list
  e = count 5 list
  f = count 6 list
```