4. Aufgabenblatt zu Funktionale Programmierung vom 19.11.2003, fällig: 26.11.2003

Themen: *Funktionen auf Zahlen, Zeichen(reihen), Tupeln und Listen*

Für dieses Aufgabenblatt sollen Sie Haskell-Rechenvorschriften zur Lösung der im weiteren angegebenen Aufgabenstellungen entwickeln und für die Abgabe in einer Datei namens `Aufgabe4.lhs` in Ihrem Homeverzeichnis ablegen. Mithin, Sie sollen ein literate Script entwickeln. Kommentieren Sie Ihre Programme wieder aussagekräftig und benutzen Sie, wo sinnvoll, Hilfsfunktionen und Konstanten.

- Schreiben Sie eine Haskell-Rechenvorschrift

  ```
  compr :: String->[(Int,Char)]
  ```

  die einen (endlichen) String mit aufeinanderfolgenden gleichen Zeichen komprimiert. Zum Beispiel soll der Aufruf `compr "aaabbbcccca"` als Ergebnis `[(3,'a'),(3,'b'),(4,'c'),(1,'a')]` liefern. In der Ergebnisliste dürfen nie zwei Paare für dasselbe Zeichen direkt nebeneinander stehen. Eine weitere Funktion

  ```
  uncompr::[(Int,Char)]->String
  ```

  soll einen mit `compr` komprimierten String wieder entkomprimieren.

- Schreiben Sie eine Haskell-Rechenvorschrift

  ```
  sums::[Integer]->[Integer]
  ```

  deren n-te Zahl in der Ergebnisliste gleich der Summe der letzten n Zahlen in der endlichen Argumentliste ist. Das Ergebnis von `sums` angewendet auf das Argument `[1..4]` soll `[4,7,9,10]` sein.

- Schreiben Sie eine Haskell-Rechenvorschrift

  ```
  sumBoth::[Integer]->[Integer]->Integer
  ```

  die die Summe der Zahlen berechnet, die in der ersten oder in der zweiten der jeweils endlichen Argumentlisten, aber nicht in beiden vorkommen. Mehrfache Vorkommen zählen nur einfach. Z. B. soll `sumBoth [1,2,1,2,3] [3,2,4]` als Ergebnis $1+4$ bzw. 5 liefern, da 1 und 4 in der jeweils anderen Liste nicht vorkommen und doppelte Vorkommen nicht mehrfach addiert werden.

- Sei $\Sigma = \{a, b, c, \ldots, z, A, B, C, \ldots, Z\}$ der schon bekannte Zeichenvorrat der Klein- und Großbuchstaben. Schreiben Sie eine Haskell-Rechenvorschrift

```
anzSigmaSymbole :: String->[[(Char,Int)]]
```

die in einer endlichen Eingabezeichenreihe s die Anzahl jedes in s vorkommenden Zeichens aus Σ bestimmt und in Form einer Liste von Listen von Paaren ausgibt. Die erste Komponente der Ergebnisliste soll für Kleinbuchstaben, die zweite Komponente für Großbuchstaben sein, die dritte für Klein- und Großbuchstaben. Weiters sollen alle Komponentenlisten nach den ersten Komponenten der Paare alphabetisch aufsteigend geordnet sein und nur Paare enthalten, deren zweite Komponente von 0 verschieden ist. Analog gilt das für die dritte Teilliste, die in entsprechender Weise die Anzahl von Vorkommen von Vokalen zusammenfasst, unabhängig davon ob groß oder klein. Erste Komponente der Paare der dritten Teilliste sollen dabei Großbuchstaben sein. Ein Beispiel: Angewendet auf die Zeichenreihe `aaObeZ-Z/saABEyebO!` soll `anzSigmaSymbole` die Ausgabe

```
[ [(a,3),(b,2),(e,2),(s,1),(y,1)],
  [(A,1),(B,1),(E,1),(O,2)(Z,2)] ],
  [(A,4),(E,3),(O,2)] ]
```

liefern.