

Hinweis: Die Übungsaufgaben 5 und 6 sind Bonusaufgaben.

1. Aufgabe (10 Punkte)

Schreiben Sie eine Funktion in Haskell, die eine Kette von Zeichen in eine Zahl (Float) umwandelt. Dezimalpunkt und Vorzeichen muss verarbeitet werden.

Beispiele der möglichen Eingaben sind: „123“, „34.23“, „+123“, „-56.23“.

2. Aufgabe (2 Punkte)

Wir haben in der Vorlesung die SKI-Kombinatoren besprochen.

Es gibt auch andere Kombinatoren in der Literatur, wie die Kombinatoren C und B mit der jeweiligen Definition:

$C\ a\ b\ c \rightarrow a\ c\ b$

$B\ a\ b\ c \rightarrow a\ (b\ c)$

Zeigen Sie dass B wie folgt definiert werden kann :

$B = S\ (K\ S)\ K$

3. Aufgabe (2 Punkte)

Reduzieren Sie den Ausdruck: $(S(KK)(SII))I$

4. Aufgabe (6 Punkte)

Definieren Sie ein folgenden Datentyp

```
data Expr = Num Integer | Sum Expr Expr | Sub Expr Expr | Mul Expr Expr |  
Divi Expr Expr  
deriving (Show,Eq)
```

Schreiben Sie einen Parser der einen String wie z.B. "1+(1+2)" in einen Ausdruck (Expression, Expr) umwandelt. Das Minus-Zeichen wird nur für die Subtraktion verwendet.

5. Bonusaufgabe (4+6 Punkte)

- a) Schreiben Sie eine rekursive Funktion **wechsel**, die bei Angabe einer Geldmenge das optimale Wechselgeld, d.h. die minimale Anzahl von Münzen, für die Geldmenge bei gegebenen Münzen berechnet. Sie können folgende konstante Hilfsfunktion verwenden.

```
coins :: [Int]  
coins = [200, 100, 50, 20, 10, 5, 2, 1]
```

- b) Schreiben Sie die obige Funktion mit Hilfe der unfold Methode.

6. Bonusaufgabe (3+3 Punkte)

- a) Benutzen Sie eine Fold-Funktion um eine eigene Funktion **myMin** zu definieren. Die Funktion soll das kleinste Element einer Liste berechnen.
- b) Benutzen Sie eine Fold-Funktion um eine eigene Funktion **abflachen** zu definieren. Die Funktion soll eine Liste von Listen zu einer Liste zusammenfassen. Verwenden Sie dazu eine Faltungsfunktion. Beispiel:
abflachen [[1,2,3], [4,5,6], [7]] wird zu [1,2,3,4,5,6,7] ausgewertet.

Wichtige Hinweise:

- 1) Verwenden Sie geeignete Namen für Ihre Variablen und Funktionsnamen, die den semantischen Inhalt der Variablen oder die Semantik der Funktionen wiedergeben.
- 2) Verwenden Sie vorgegebene Funktionsnamen, falls diese angegeben werden.
- 3) Kommentieren Sie Ihre Programme.
- 4) Verwenden Sie geeignete lokale Funktionen und Hilfsfunktionen in Ihren Funktionsdefinitionen.
- 5) Schreiben Sie in alle Funktionen die entsprechende Signatur.