Lehrstuhl für Programmiersprachen und Übersetzterkonstruktion
CUA Kiel, Informatik und Prakt. Mathematik
Prof. Dr. M. Hanus, Frank Huch

24118 Kiel
Christan-Albrechts-Platz 4
Tel.: 0431 / 880-7277

1. Übung „Übersetzerkonstruktion“

Abgabe am 22. Oktober in der Vorlesung oder bis spätestens Mittwoch Mittag im Hochhaus, Raum 704

In dieser ersten Übung sollen Sie sich mit dem Hugs-System vertraut machen.
Public domain Versionen und weitere Informationen zur Vorlesung und den Übungen finden Sie im WWW unter der Adresse:

`http://www.informatik.uni-kiel.de/~fhu/CBW01/`

Aufgabe 1 2 Punkte

Die Funktion `double` sei wie folgt definiert:

```
double x = x + x
```

Geben Sie alle möglichen Auswertungen von `double(double 3)` an.

Aufgabe 2 1+3 Punkte

In der Übung wurde die Fibonacci-Funktion intuitiv und mittels Akkumulatortechnik definiert. Berechnen Sie `fib1/fib2 5` und `fib1/fib2 30` und geben Sie die Zahl der benötigten Reduktionsschritte (entspr. Zeitbedarf) und der benötigten Zellen (entspr. Platzbedarf) an. Hugs gibt beides nach Eingabe der Anweisung '`:set +s`' aus.
Bestimmen Sie anhand der benötigten Reduktionsschritte die Zeitkomplexitäten der Implementierungen (`fib1/fib2` liegt in $O(??)$). Die einfachste Methode ist eine graphische Auswertung.

Aufgabe 3 2+1+1 Punkte

Programmieren Sie folgende Funktionen in Haskell und geben Sie jeweils auch den Typ der definierten Funktionen an:

a) Definieren Sie eine Funktion `rev`, welche eine Liste umdreht. Welchen Aufwand hat ihr Algorithmus?

   Optimieren Sie ihre Lösung mit Hilfe der Akkumulatortechnik auf lineare Komplexität (mit Begründung).

b) Definieren Sie eine zweistellige Funktion `stelle`, die feststellt, an welcher Position ein Element in einer Liste vorkommt. Falls das Element nicht in der Liste vorkommt, soll die Funktion den Wert 0 liefern.

Beispiel: `stelle 1 [2,1,3,1]` ergibt `2`

c) Definieren Sie eine Funktion `inconcat`, die eine Liste von Strings zu einem neuen String konkateniert und dabei zwischen je zwei Elemente eine Zeichenkette einfügt.

Beispiel: `inconcat "-" ["Christian","Albrechts","Universität"]`
ergibt `"Christian-Albrechts-Universität"`

Wichtige Interpreter-Kommandos des Hugs-Systems:

- `:l` *filename*

  Löschen aller vorhandenen Funktionsdefinitionen (außer Prelude) und Laden des Programms *filename.*

- `:r`

  Wiederholt die letzte Ladeoperation, somit werden alle Funktionsdefinitionen aktualisiert.

- `:q`

  Verlassen des Hugs-Systems.

- `:?`

  Komandoliste wird angezeigt.

- *Ausdruck* (z.B. `fac 3`)

  Start einer Funktionsberechnung.