Prof. Dr. R. Loogen, M. Dieterle
Fachbereich Mathematik und Informatik
Hans-Meerwein-Straße
D-35032 Marburg

2. November 2010

Philipps Universität Marburg

Übungen zu „Parallelität in funktionalen Programmiersprachen“, WS 2010

Nr. 3, Abgabe der Aufgaben: 9. November 2010 vor der Vorlesung

---

Aufgaben

3.1 Paralleles 'Divide and Conquer' — 7 Punkte

Ähnlich wie `parMap` eine parallele Implementierung der **`map`** Funktion liefert, sollen Sie eine allgemeine Parallelisierung von 'Divide and Conquer' Problemen definieren. Ausgangspunkt der Parallelisierung ist die Funktion `dc`:

```
dc :: (a → Bool) → (a → b) → (a → [a]) → ([b] → b) → a → b
dc trivial solve divide combine x
  | trivial x = solve x
  | otherwise = combine children
    where children = map (dc trivial solve divide combine) (divide x)
```

(a) Schreiben Sie eine simple erste parallele Variante `parDC`. / 1

(b) Erweitern Sie die Funktion `parDC` um eine parallele Tiefenkontrolle, analog zur Aufgabe 2.1 e): `parDC_D :: Int → (a → Bool) → ...` / 1

(c) Bei Aufgabe 2.1 sollten sie die **`length`** Funktion benutzen, um einen höheren Auswertungsgrad der parallelen Berechnungen zu erzwingen. Erweitern Sie `parDC_D` um eine Parameterfunktion vom Typ `(b → c)` und setzen Sie diese analog zu Aufgabe 2.1 zur Transformation des Auswertungsgrades ein. / 2
`parDC_D_E :: Int → (b → c) → (a → Bool) → ...`

(d) Formulieren Sie Aufgabe 2.1 e) als Instanz von `parDC_D_E`. Alternativ können Sie auch ein paralleles Quick- oder Merge-Sort als Instanz von `parDC_D_E` definieren. / 3

3.2 Bedarfsgesteuerte Auswertung und Striktheit — 5 Punkte

Erläutern Sie die Auswirkung der Striktheitseigenschaften der Parameterfunktion auf die Listenauswertung bei den folgenden Haskell-Funktionen:

(a) Präfix **`takeWhile`** `:: (a → Bool) → [a] → [a]` / 1

(b) Listenzusammenführung **`zipWith`** `:: (a → b → c) → [a] → [b] → [c]` / 2

(c) Listenfaltung **`foldl1`** `:: (a → a → a) → [a] → a` / 2

Geben Sie zu jeder Funktion Beispielaufrufe an, bei denen die Listenparameter nicht bzw. nicht vollständig ausgewertet werden müssen. Erläutern Sie für Ihre Beispiele jeweils die Auswertung des Ergebnisses.