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

05.Mai 2009

Philipps Universität Marburg

Übungen zu „Parallelität in funktionalen Programmiersprachen“, Sommer 2009

Nr. 3, Abgabe der Aufgaben: 12.Mai 2009 vor der Vorlesung

Aufgaben

3.1 Paralleles Quicksort

7 Punkte

(a) Parallelisieren Sie die abgebildete Quicksort Funktion mit den Kombinatoren `par` und `pseq`. Achten Sie auf sinnvolle Auswertungsgrade. / 1

(b) Erstellen Sie eine optimierte Version `qsortD :: (Ord a) ⇒ Int → [a] → [a]` mit einem zusätzlichen Parameter `depth` zur Tiefenkontrolle. Wenn `depth` Ebenen parallel expandiert sind, soll das Programm sequentiell weiterarbeiten. / 2

(c) Erstellen Sie eine weitere Version `qsortL :: (Ord a) ⇒ Int → [a] → [a]`, welche nur Listen einer bestimmten Mindestlänge weiter parallel sortiert. Bei jedem Rekursionsaufruf soll also die Länge der Inputliste abgeschätzt werden. / 2

(d) Vergleichen Sie Laufzeiten der verschiedenen Versionen mit Hilfe von zufällig generierten Eingaben. Variieren Sie die Anzahl der Prozessoren, die Listenlänge sowie die Parameter zur Tiefen- und Längenkontrolle. / 2

```
import System.Random (mkStdGen, randoms)

qsort :: (Ord a) ⇒ [a] → [a]
qsort (x:xs) = lesser ++ x:greater
    where lesser  = qsort [y | y ← xs, y <  x]
          greater = qsort [y | y ← xs, y ≥ x]
qsort _ = []

rnd :: Int → [Int]
rnd k = let result = take k (randoms (mkStdGen 1))
        in length result `pseq` result
```

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) Listentransformation `map :: (a → b) → [a] → [b]`

(b) Listenfaltung `foldr1 :: (a → a → a) → [a] → a`

(c) Listenprädikat `any :: (a → Bool) → [a] → Bool`

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