

Übungen zu Informatik II (Sommersemester 2008) Musterlösung zu Aufgabenblatt 6

Aufgabe 22 (Präsenzaufgabe)

Die folgende iterative Variante von **GenericQuickSort** nutzt einen Keller, um die Feldgrenzen von noch zu sortierenden Teilfeldern abzulegen. Da immer zuerst das größere Teilfeld auf den Keller gelegt und somit das kleinere immer zuerst abgearbeitet wird, kann man zeigen, dass der Keller höchstens logarithmisch tief ist.

IterativeGenericQuickSort(a)

Vorbedingung: a ist ein ganzzahliges Feld.

Initialisiere den Keller mit den Feldgrenzen.

setze $s = \text{newLinkedStack}()$

$s.\text{pushItem}(0)$

$s.\text{pushItem}(|a| - 1)$

solange nicht $s.\text{isEmpty}()$

Bearbeite ein Feldsegment vom Keller.

setze $r = s.\text{popItem}()$

setze $l = s.\text{popItem}()$

falls $l < r$ ist

Partitioniere das Feld.

belege j nach Pivotwahlstrategie ($l \leq j \leq r$)

setze $j = \text{Partition}(a, l, j, r)$

Lege Feldsegmente nach ihrer Größe auf dem Keller ab.

falls $j - l < r - j$

$s.\text{pushItem}(l + 1)$

$s.\text{pushItem}(r)$

$s.\text{pushItem}(l)$

$s.\text{pushItem}(j - 1)$

sonst

$s.\text{pushItem}(l)$

$s.\text{pushItem}(j - 1)$

$s.\text{pushItem}(l + 1)$

$s.\text{pushItem}(r)$

Nachbedingung: $[a]$ ist eine geordnete Umordnung von $[\bar{a}]$.

Aufgabe 21 (Präsenzaufgabe)

EvalRPNExpression(a)

Vorbedingung: a ist ein Feld, dessen Inhalt ein arithmetischer Ausdruck in RPN ist.

```
setze  $s = \text{newLinkedStack}()$ 
für  $i = 0$  bis  $|a| - 1$ 
  falls isNumber( $a[i]$ )
     $s.\text{pushItem}(a[i])$ 
  sonst
    setze  $o_2 = s.\text{popItem}()$ 
    setze  $o_1 = s.\text{popItem}()$ 
     $s.\text{pushItem}(\text{eval}(a[i], o_1, o_2))$ 
setze  $\text{return} = s.\text{popItem}()$ 
```

Nachbedingung: return ist der Wert des in \bar{a} gespeicherten RPN-Ausdrucks und $[a] = [\bar{a}]$.

Aufgabe 23 (Hausaufgabe)

RadixSort(a)

Vorbedingung: a ist ein Feld natürlicher Zahlen $< 2^k$ für ein $k \in \mathbf{N}$.

ParaRadixSort($a, 0, |a| - 1, k - 1$)

Nachbedingung: $[a]$ ist eine geordnete Umordnung von $[\bar{a}]$.

ParaRadixSort(a, l, r, m)

Vorbedingung: a ist ein Feld natürlicher Zahlen $< 2^k$ für ein $k \in \mathbf{N}$, $0 \leq m < k$, $0 \leq l \leq r < |a|$.

```
falls  $m \geq 0$ 
  setze  $L = l - 1$  und  $R = r + 1$ 
  solange  $L < R$ 
    wiederhole setze  $L = L + 1$  bis  $L = r + 1$  oder  $a[L].\text{bit}(m) = 1$ 
    wiederhole setze  $R = R - 1$  bis  $R = l - 1$  oder  $a[R].\text{bit}(m) = 0$ 
    falls  $L < R$ 
      Swap( $a, L, R$ )
  ParaRadixSort( $a, l, R, m - 1$ )
  ParaRadixSort( $a, L, r, m - 1$ )
```

Nachbedingung:

1. $l = \bar{l}$, $r = \bar{r}$, $m = \bar{m}$, $a[i] = \bar{a}[i]$ für alle $i \in \{0, \dots, l - 1, r + 1, \dots, |a| - 1\}$,
2. $[a][l..r]$ ist eine Umordnung von $[\bar{a}][l..r]$,
3. $a[l] \bmod 2^{m+1} \leq a[l + 1] \bmod 2^{m+1} \leq \dots \leq a[r] \bmod 2^{m+1}$.

Aufgabe 24 (Hausaufgabe)

Voraussetzung: Wir betrachten **SimpleQuickSort** wie im Vorlesungsskript beschrieben. Sei $q(n)$ definiert wie im Skript. Sei $n \in \mathbf{N}$.

Behauptung: Die durchschnittliche Laufzeit von **SimpleQuickSort** für Felder der Länge n ist in $\Omega(n \log n)$.

Beweis: Sei $n \in \mathbf{N}$ mit $n > 1$. Dann gelten laut Voraussetzung folgende Ungleichungen:

$$n + \frac{2}{n} \sum_{i < n} q(i) \leq q(n) , \quad (1)$$

$$q(n) \leq n + 1 + \frac{2}{n} \sum_{i < n} q(i) . \quad (2)$$

Multiplizieren wir die Ungleichung (1) mit n , so erhalten wir:

$$n \cdot q(n) \geq n^2 + 2 \sum_{i < n} q(i) . \quad (3)$$

Ersetzen wir in Ungleichung (2) jedes n durch $n - 1$ und multiplizieren wir die Ungleichung mit $n - 1$, so erhalten wir

$$(n - 1) \cdot q(n - 1) \leq (n - 1) \left(n + \frac{2}{n - 1} \sum_{i < n - 1} q(i) \right) = n^2 - n + 2 \sum_{i < n - 1} q(i) . \quad (4)$$

Subtrahieren wir (4) von (3), so erhalten wir

$$n \cdot q(n) - (n - 1) \cdot q(n - 1) \geq n + 2q(n - 1) , \quad (5)$$

woraus sich

$$n \cdot q(n) \geq (n + 1) \cdot q(n - 1) + n \quad (6)$$

ergibt, was wiederum zu

$$\frac{q(n)}{n + 1} \geq \frac{q(n - 1)}{n} + \frac{1}{n + 1} \quad (7)$$

äquivalent ist. Durch wiederholtes Einsetzen ergibt sich daraus (formaler Beweis per Induktion):

$$\frac{q(n)}{n + 1} \geq \frac{q(1)}{2} + \sum_{i=3}^{n+1} \frac{1}{i} . \quad (8)$$

Daraus und aus einer früheren Überlegung, aus der wir $\sum_{i=3}^{n+1} \frac{1}{i} \in \theta(\log n)$ folgern können, folgt auch $q(n) \in \Omega(n \log n)$.