

Übungen zur Vorlesung
Datenstrukturen, Algorithmen und Programmierung 2 (DAP2)
Sommersemester 2007

Blatt 8

Aufgabe 8.1 (5 Punkte)

Gegeben sei der folgende Algorithmus:

Algo($A[1], \dots, A[n]$) // $A[1], \dots, A[n] \in \mathbb{N}$

```
1: for  $i = n$  downto 2 do
2:   for  $j = 1$  to  $i - 1$  do
3:     if  $A[j] > A[j + 1]$  then
4:       vertausche  $A[j]$  und  $A[j + 1]$ ;
5:     end if
6:   end for
7: end for
```

Ist dieser Algorithmus ein stabiler Sortieralgorithmus?

Gib eine n -elementige Eingabe ganzer Zahlen an, die zu einer maximalen Anzahl von Vertauschungsoperationen in Zeile 4 führt. Zu wie vielen Vertauschungsoperationen führt diese Eingabe?

Ist dieser Algorithmus ein adaptiver Sortieralgorithmus? Falls nein, gib, wenn möglich, eine Modifikation des Algorithmus an, so dass dieser adaptiv ist.

Aufgabe 8.2 (5 Punkte)

In der Vorlesung ist das Mergesort-Verfahren zum Sortieren vorgestellt worden. Benötigt wird bei diesem Verfahren eine Hilfsfunktion **MERGE**. Beschreibe in Pseudocode die Hilfsfunktion **MERGE**($a_1, \dots, a_k, b_1, \dots, b_m$). Dabei darf davon ausgegangen werden, dass a_1, \dots, a_k und b_1, \dots, b_m bereits sortiert vorliegen. Die Funktion liefert ein Array mit einer korrekt sortierten Folge der Zahlen $a_1, \dots, a_k, b_1, \dots, b_m$ zurück.

In der Vorlesung wurde angegeben, dass die Funktion **MERGE** mit höchstens $k + m - 1$ Vergleichen auskommt. Wie muss für gleichlange Teillisten $a_1 \dots a_k$ und $b_1 \dots b_k$ die Ordnung der Gesamtliste aussehen, damit der worst case $2k - 1$ erreicht wird? Was ist der best case, also die Mindestanzahl von Vergleichen?

Aufgabe 8.3 (5 Punkte)

Wende den MERGESORT-Algorithmus aus der Vorlesung (Algorithmus 4.3.1, Skript S. 98) auf folgende Eingabe an:

99, 41, 62, 13, 84, 35, 96, 57, 28, 79, 1

Aufgabe 8.4 (5 Punkte)

Der worst case von Quicksort mit Zerlegungsstrategie 1 tritt auf, wenn die Elemente im Array aufsteigend sortiert sind, die Laufzeit beträgt $\Omega(n^2)$.

Für $n = 32$ wäre dies z. B. das Array $(1, 2, 3, 4, \dots, 31, 32)$. Finde ein Array mit den gleichen 32 Zahlen, das bei Quicksort mit Zerlegungsstrategie 1 zum best case führt.

Wie sieht der best case bei allgemeinen n aus? Begründe, warum dies der best case ist.