Technische Universität München
Institut für Informatik
Lehrstuhl für Effiziente Algorithmen
Volker Heun

SS 1999
Übungsblatt 4
8. Juni 1999

Grundlegende Algorithmen

Abgabetermin: 15. Juni

Aufgabe 1

Ein Sortierverfahren $\mathcal{S}$ heißt *stabil*, wenn Datensätze mit gleichen Schlüsseln ihre ursprüngliche Reihenfolge beibehalten. D.h. wenn für eine Folge von zu sortierenden Schlüsseln $k_1, \ldots, k_n$ für alle $i < j \in [1 : n]$ mit $k_i = k_j$ gilt, daß $\pi(i) < \pi(j)$ (wobei π die mit $\mathcal{S}$ gefundene Sortierung ist, also $k_{\pi(1)} \leq \cdots \leq k_{\pi(n)}$).

Welche der untersuchten Sortierverfahren (Mergesort, Heapsort, Quicksort) sind stabil?

Aufgabe 2

Kann man jedes Sortierverfahren so modifizieren, daß es stabil sortiert?

Aufgabe 3

Betrachten Sie das folgende Problem: *Gegeben sei eine Folge* $(a_1, \ldots, a_n)$ *ganzer Zahlen* ($\in \mathbb{Z}$). *Bestimme eine Teilfolge* $(a_i, \ldots, a_j)$ *mit* $1 \leq i \leq j \leq n$, *so daß*

$$\sum_{\lambda=i}^{j} a_\lambda \geq \max_{1 \leq \mu \leq \nu \leq n} \sum_{\lambda=\mu}^{\nu} a_\lambda.$$

Entwerfen Sie einen Divide-and-Conquer-Algorithmus der obiges Problem in Zeit $O(n \log(n))$ löst. Nehmen Sie dazu an, daß die Folge $(a_1, \ldots, a_n)$ in einem Feld der Größe n abgespeichert ist.

Aufgabe 4

Geben Sie einen Divide-and-Conquer-Algorithmus zur Bestimmung des Elementes mit Rang k in einer Folge $(a_1, \ldots, a_n)$ an, dessen Divide-Schritt der Aufteilungsprozedur in Quicksort stark ähnelt.