Theoretische Informatik 1 / Algorithmentheorie

Wintersemester 2012/13

Prof. Dr. Georg Schnitger,
Dipl. Inf. Bert Besser,
Dipl. Inf. Matthias Poloczek

Arbeitsgruppe Theoretische Informatik, Institut für Informatik

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Blatt 2

Ausgabe: 25.10.2012
Abgabe: 01.11.2012 **vor** der Vorlesung

2.1. Aufgabe (4+4) *Mergesort*

Es sei das folgende ganzzahlige Array A gegeben: $A = [2, 4, 7, 12, 3, 5, 6, 9]$

a.) **Mische** die Teilfelder $A[1..4]$ und $A[5..8]$, indem Du die in der Vorlesung vorgestellte Funktion `Merge(...)` benutzt. Diese Funktion ist auch auf S. 24 im Skript beschrieben. Gib Deine Zwischenschritte an.

b.) In der Vorlesung wurde auch der nicht rekursive Mergesort besprochen. **Gib** für ein Feld der Größe sechzehn die Reihenfolge der Mischvorgänge **an**, jeweils beschrieben durch linken und rechten Endpunkt im Feld, d.h. (i, j) mischt die Teilsequenzen mit den Indizes $i, \ldots, \lfloor (j+i)/2 \rfloor$ und $\lceil (j+i)/2 \rceil, \ldots, j$.

Beispiel: Für Größe vier ist $(1, 2), (3, 4), (1, 4)$ die entsprechende Reihenfolge.

2.2. Aufgabe (8) *Interessante Angebote*

Eine Suchmaschine holt eine große Anzahl n von Angeboten für eine Kundenanfrage ein. Ein Angebot $A_i = (q_i, p_i)$ wird durch seine Qualität $q_i \in I\!R^+$ und den Preis $p_i \in I\!R^+$ beschrieben ($I\!R^+$ ist die Menge der positiven reellen Zahlen). Die Qualität drückt aus, wie nah das Angebot an die Wünsche des Kunden kommt. Hohe Werte bedeuten eine hohe Übereinstimmung.

Ein Angebot $A_i = (q_i, p_i)$ heißt *uninteressant*, wenn es ein alternatives Angebot $A_j = (q_j, p_j)$ gibt, so dass die Qualität von A_j mindestens so hoch ist wie die von A_i und A_j echt weniger kostet $((q_i \leq q_j) \wedge (p_j < p_i))$ oder falls die Qualität von A_j besser ist als die von A_i und A_j nicht mehr kostet als A_i $((q_i < q_j) \wedge (p_j \leq p_i))$. Um den Kunden nicht mit einer Flut von sinnlosen Daten zu überschwemmen, soll ein Algorithmus alle uninteressanten Angebote herausfiltern.

Beschreibe einen Algorithmus, der in Laufzeit $O(n \cdot \log(n))$ alle Angebote ausgibt, die nicht als uninteressant eingestuft sind.

2.3. Aufgabe (8) *Median-Bestimmung*

Gegeben sind zwei Mengen X, Y von je n Zahlen, so dass sich keine der insgesamt $2n$ Zahlen gleichen. Es soll der Median m, d.h. das n-kleinste Element, aller $2n$ Zahlen bestimmt werden. Wir haben die Möglichkeit, das k-kleinste Element für eine beliebige der beiden Mengen und für beliebiges k in konstanter Zeit anzufragen.

Gib einen Algorithmus an, der den Median mit höchstens $O(\log_2 n)$ Anfragen bestimmt.