Theoretische Informatik I

Blatt 6, 23.11.2004, Abgabe 30.11.2004 in der Vorlesung

Aufgabe 20. In der Auswahlphase von `Heap-Sort` ist $S = S_1$ tendenziell recht klein, so dass im allgemeinen $S < S_k$ gilt und die WHILE-Schleife nicht abbricht. Ändere die Auswahlphase so, dass S nicht mit S_k verglichen wird und die mittlere Anzahl der Vergleiche halbiert wird.
Hinweis: [Kn. 5.2.3, Aufgabe 18, Algorithm H]

Aufgabe 21. Gib ein Verfahren an, das mit möglichst wenigen Vergleichen folgendes leistet

`EINGABE` $S[1:n]$ ein Heap, j mit $1 \leq j \leq n$

`AUSGABE` Heap $S[1:n-1]$ aus dem S_j^{alt} entfernt wurde.

Wie viele Vergleiche in Abhängigkeit von j und n führt das Verfahren maximal aus?

Aufgabe 22. Betrachte die deterministische Prozedur `Quick-Sort`$(1, n)$. Die Eingabe $S_1, \ldots, S_n$ sei zufällig gleichverteilt permutiert. Zeige:
An der Stelle $(*)$ liegen bei festem r zufällige Permutationen $P_\ell = (S_1, \ldots, S_{r-1}) \in \text{Sym}\{1, \ldots, r-1\}$ und $P_r = (S_{r+1}, \ldots, S_n) \in \text{Sym}\{r+1, \ldots, n\}$ vor. Wie viele Eingabepermutationen werden bei festem r und t ($t =$ Anzahl der vorherigen Austausche) auf das Paar (P_ℓ, P_r) abgebildet?

Aufgabe 23. Matrix–Vektor AND/OR Produkt

INPUT $\quad A = [a_{i,j}]_{\substack{1\le i\le n\\ 1\le j\le m}} \in \{0,1\}^{n\times m}, \;\; \mathbf{b} = (b_1, \ldots, b_m)^t \in \{0,1\}^m$

#A sei gegeben durch die Listen $S_1(A), ..., S_m(A)$ und $Z_1(A), ..., Z_n(A)$.

 # berechne für $i = 1, ..., n$: $c_i = \bigvee_{j=1,...,m} (a_{i,j} \wedge b_j)$.

1. $z := 0,\; c_1 := c_2 := \cdots := c_n := 0,\;\; t := \sum_{i=1}^m \#S_i(A),\;\; j := 1$
2. WHILE $z < n\ln(t/n)$ and $j \le m$ DO

   [IF $b_j = 1$ THEN FOR all $i \in S_j(A)$ DO $[c_i := 1,\; z := z+1,]$

   $j := j+1$, IF $j > m$ THEN terminate]
3. FOR all i with $c_i = 0$ DO FOR all $j \in Z_i(A)$ DO

   IF $b_j = 1$ THEN $c_i := 1$

OUTPUT $c_1, \ldots, c_n$.

Dabei sei $Z_i(A) = \{j \mid a_{i,j} = 1\}$, $S_j(A) = \{i \mid a_{i,j} = 1\}$.

Aufgabe 13. Die Wahrscheinlichkeit $\mathbf{WS}(A)$ sei eine Funktion von $(\#S_1(A), ..., \#S_m(A))$. Damit sind die $\#S_j(A)$ 1-Einträge von Spalte j zufällig verteilt und unabhängig von den anderen Spalten.

1. Bestimme: $\mathbf{E}[\#Z_i(A)]$ für $i = 1, ..., n$
2. Zeige: Die mittlere Laufzeit des Algorithmus ist $O(n\ln(t/n))$.
3. Optimiere die Schranke $n\ln(t/n)$ in Schritt 3 über $n\ln(t/n) - ns$ mit $s \in \mathbb{N}$, so dass die erwartete Schrittzahl minimal wird.