Theoretische Informatik I

Blatt 2, 26.10.2004, Abgabe 02.11.2004 in der Vorlesung

Aufgabe 5. Gegeben sei der Vektor $\mathbf{b} = (b_1, \ldots, b_n)^t$ und $A \in \mathbf{Z}^{m \times n}$ nach der Koordinatenmethode als Array $B[1:3k]$. Schreibe ein Programm, welches den Vektor $\mathbf{c} := A\mathbf{b}$ in $O(k)$ Schritten berechnet.

Aufgabe 6. Sei X Zufallsvariable mit $\mathbf{Pr}[X = k] = (1-p)^{k-1} \cdot p$ und $p \in]0,1[$. Es gilt $\sum_{k=1}^{\infty}(1-p)^{k-1} \cdot p = 1$. X ist *geometrisch* verteilt. Bestimme $\mathbf{E}[X] = \sum_{k=1}^{\infty} k \cdot \mathbf{Pr}[X = k]$.

$\mathbf{Pr}[X = k]$ ist die **Ws**, dass bei der k-ten unabh. Wiederholung eines zufälligen Y mit $\mathbf{Pr}[Y = 1] = p$ erstmals das Ereignis $[Y = 1]$ auftritt.
Bestimme $\mathbf{E}[X] = \sum_{k=1}^{\infty} k \cdot \mathbf{Pr}[X = k]$.

Hinweis: $-\frac{1}{p^2} = \frac{\partial}{\partial p}(\frac{1}{p}) = \frac{\partial}{\partial p}(\frac{1}{1-(1-p)}) = -\sum_{k=1}^{\infty}(k-1)(1-p)^{k-1}$.

`MERGE-SORT`$(A[n : n+l-1])$

1. `MERGE-SORT`$(A[n : n + \lceil l/2 \rceil - 1])$, `MERGE-SORT`$(A[n + \lceil l/2 \rceil : n+l-1])$
2. `MERGE`$(A[n : n + \lceil l/2 \rceil - 1], A[n + \lceil l/2 \rceil : n+l-1])$.

Die Anzahl der Vergleiche über $\mathbf{Z} \cup \{+\infty\}$ bei Ausführung von `MERGE-SORT` ist eine Funktion $M(l)$ der Listenlänge l. Warum ?

Aufgabe 7. Bestimme $M(2), \ldots, M(6)$ mit Begründung. Zeige:
Die `INPUT`-Liste $A[1 : 2^d]$ für $n = 1, l = 2^d$ erfordert bei `MERGE-SORT` nur $\leq (d-1)2^d + 1 - k$ Vergleiche über $\mathbf{Z}$, wenn die $k+1$ grössten Einträge entweder in der ersten oder in der zweiten Hälfte von $A[1 : 2^d]$ liegen.

Aufgabe 8. Charakterisiere diejenigen Listen $A[1 : 2^d]$, für die `MERGE-SORT` genau $(d-1)2^d + 1$ Vergleiche über $\mathbf{Z}$ benötigt, mit Beweis.