Algorithmen und
Berechnungskomplexität I
Wintersemester 2017/18

Abgabe: 07.11.2017, 12.00 Uhr
Besprechung: KW 46

PD Dr. Elmar Langetepe
David Kübel
Institut für Informatik

Übungsblatt 3

Aufgabe 3.1: (4 Punkte)

Eine $m \times n$ Matrix $M[.][.]$ heißt *monoton*, wenn für alle $1 \leq k < \ell \leq m$ und $1 \leq i < j \leq n$ gilt, dass

$$M[k][i] < M[k][j] \Rightarrow M[\ell][i] < M[\ell][j].$$

Das bedeutet wenn in der k-ten Zeile das i-te Element kleiner ist als das j-te Element, dann ist in jeder Zeile unterhalb der k-ten Zeile das i-te Element ebenfalls kleiner als das j-te.

Geben Sie einen *Divide- and Conquer* Algorithmus an, der in jeder Zeile die Position des linkesten Zeilenmaximums bestimmt, und geben Sie dessen Laufzeit in O-Notation an. Die Laufzeit des Algorithmus soll in $O(n + m + n \log m)$ liegen.

Es darf angenommen werden, dass die Matrixeinträge schon initialisiert sind. Für die Laufzeit sind lediglich die Rechenoperationen des Algorithmus, wie z. B. Vergleiche der Form "Gilt $M[a][b] < M[c][d]$?" relevant.

Hinweis: Beweisen Sie zunächst, dass der Wert "Spaltenindex des linkesten Maximums in Zeile k" monoton in k wächst.

Aufgabe 3.2: (4 Punkte)

Sei eine rekursive Kostenfunktion T definiert durch

$$T(n) = T(n-1) + 2 \cdot n + 3$$

und $T(0) = 0$. Beweisen Sie mittels vollständiger Induktion über n, dass $T(n) \in \Theta(n^2)$.

Hinweis: Stellen Sie zunächst eine Vermutung für einen geschlossenen Ausdruck von $T(n)$ auf und beweisen Sie diesen via Induktion. Zeigen Sie damit, dass $T(n) \in \Theta(n^2)$.

Aufgabe 3.3: (4 Punkte)

Finden Sie für die folgende rekursive Funktion $S : \mathbb{N} \to \mathbb{N}$ eine geschlossene Form und beweisen Sie Ihre Behauptung:

$$S(1) = 1$$
$$S(n) = \sum_{i=1}^{n-1} i \cdot S(i) \qquad \text{für } n > 1.$$

Aufgabe 3.4: (1+1+1+1 Punkte)

Verwenden Sie das Master-Theorem, um für die folgenden Rekursionsgleichungen asymptotische Schranken anzugeben:

(a) $T(n) = 2T(\frac{n}{4}) + 1$

(b) $T(n) = 2T(\frac{n}{4}) + \sqrt{n}$

(c) $T(n) = 2T(\frac{n}{4}) + n$

(d) $T(n) = 2T(\frac{n}{4}) + n^2$

Aufgabe 3.5:

(1+2+4+1 Zusatzpunkte)

Diese Aufgabe zeigt ein Verfahren zum Lösen (komplizierterer) Rekursionsgleichung mit Methoden der Linearen Algebra. Betrachten Sie die folgende Rekursionslgeichung $f_n = f_{n-1} + f_{n-2}$ mit $f_0 = f_1 = 1$.

(a) Finden Sie eine geeignete Matrix $A \in \mathbb{R}_{2\times 2}$, sodass $\begin{pmatrix} f_n \\ f_{n-1} \end{pmatrix} = A \cdot \begin{pmatrix} f_{n-1} \\ f_{n-2} \end{pmatrix}$ für $n \geq 2$.

(b) Beweisen Sie $f_n = (1,0) \cdot A^{n-1} \cdot \begin{pmatrix} f_1 \\ f_0 \end{pmatrix}$ für $n \geq 1$ mittels vollständiger Induktion.

(c) Berechnen Sie eine Diagonalisierung von A, d.h. berechnen Sie die Matrix $S \in \mathbb{R}_{2\times 2}$ und ihre Inverse S^{-1}, sodass $D = S^{-1} \cdot A \cdot S$ eine Diagonalmatrix mit den Eigenwerten von A auf der Diagonale ist.

(d) Leiten Sie eine geschlossene Form für f_n her.