Algorithmen und Berechnungskomplexität I WS 15/16

Universität Bonn, Institut für Informatik, Abteilung I

3. Aufgabenblatt zur Vorlesung **Abgabe: 10.11.** (12^{30})

- *Unter **www.i1.informatik.uni-bonn.de** werden Unterlagen zu Vorlesung und Übung bereitgestellt.*
- *Bearbeitung und Abgabe der Übungsblätter ist in festen Gruppen von bis zu 3 Personen erlaubt.*
- *Auf den Abgaben müssen die Namen aller Teilnehmer und die Nummer der Übungsgruppe (siehe Homepage) stehen.*
  Achtung: Abgaben ohne Nummer der Übungsgruppe werden nicht berücksichtigt!
- *Lösungen können am Tag der Abgabe bis zum Beginn der Vorlesung in den Postkasten im AVZ III eingeworfen werden.*

Aufgabe 9: Laufzeitbestimmung Dichtestes Punktepaar (4 Punkte)

Analysieren Sie die Gesamtlaufzeit des *Divide and Conquer*-Algorithmus, wie er in der Vorlesung vorgestellt wurde, zum Bestimmen eines dichtesten Punktepaares einer Menge von Punkten in der Ebene. Stellen Sie hierzu eine Rekursionsgleichung auf, die Sie lösen, um die Laufzeit des Verfahrens in Θ-Notation anzugeben. Sie können die vereinfachenden Annahmen aus der Vorlesung verwenden und folglich als Rekursionsgleichung Gleichung (2.4) im Vorlesungsskript zugrunde legen.

Aufgabe 10: Maximum einer monotonen Matrix (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 relevant sind lediglich die Rechenoperationen des Algorithmus, wie zum Beispiel Vergleiche der Form „Gilt $M[a][b] < M[c][d]$?“.

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

Aufgabe 11: Rekursionsgleichungen (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 $T(n) \in \Theta(n^2)$.

Aufgabe 12: Master-Theorem (4 Punkte)

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

a) $T(n) = 2T(n/4) + 1$.

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

c) $T(n) = 2T(n/4) + n$.

d) $T(n) = 2T(n/4) + n^2$.