

# Prüfung zur Lehrveranstaltung

## Datenstrukturen und Algorithmen

### Stoffsemester WS 2002/2003

Es sind keinerlei Unterlagen oder Hilfsmittel erlaubt. Es dürfen nur einzelne, lose Blätter verwendet werden! Auf jedem Blatt muß der Name und die Matrikelnummer angegeben werden! Reine Arbeitszeit sind 90 Minuten.

(1) (5 Punkte).

- (a) Definieren Sie die  $\Theta$ -Notation und erläutern Sie sie mit eigenen Worten. Verwenden Sie eine Skizze zur Erklärung.
- (b) Geben Sie eine Funktion  $f(n)$  an, die  $f(n) \notin O(n)$  und  $n \log n \notin O(f(n))$  erfüllt!
- (c) Wie verhalten sich die beiden Funktionsklassen  $O(n^{ln 5})$  und  $O(5^{ln n})$  zueinander? Beweisen Sie Ihre Aussagen!

(2) (5 Punkte). Beweisen oder widerlegen Sie die folgenden Sätze. Verwenden Sie dabei keine Beispiele, sondern eine folgerichtige Argumentation!

- (a) Ein Algorithmus, der einen Input der Größe  $n$  in  $O(n \log n)$  Zeit bearbeitet, benötigt  $O(n \log n)$  Speicher.
- (b) Ein Algorithmus, der für einen Input der Größe  $n$  zur Verarbeitung  $\Theta(n^2)$  Speicher benötigt, hat eine Laufzeit von  $\Omega(n \log n)$ .
- (c) Es existiert ein Sortieralgorithmus, der  $n$  beliebige Zahlen mit  $O(n^3)$  Vergleichen sortiert.
- (d) Jeder Algorithmus, der für einen Input der Größe  $n$  insgesamt  $\Theta(n)$  Speicher benötigt, ist schneller als ein Algorithmus, der  $\Omega(n^3)$  Speicher benötigt.
- (e) Es gibt einen Algorithmus, der für einen Input der Größe  $n$  eine Zeit von  $O(n \log n)$  und einen Speicher von  $\Omega(n^2)$  benötigt.

(3) (5 Punkte). Geben Sie Häufigkeiten für die Buchstaben A, B, C, D, E, F, G und H an, sodaß 10, 11, 011, 010, 001, 0001, 00001 und 00000 eine optimale Codierung für diese acht Buchstaben ist. Begründen Sie Ihre Antwort!

(4) (5 Punkte). Gegeben sei eine  $n \times n$  Matrix  $A$  (ein zweidimensionales Feld  $A[1 \dots n, 1 \dots n]$ ), die lauter Floating-Point Zahlen enthält. Sowohl die Spalten als auch die Zeilen sind aufsteigend sortiert.

Entwerfen Sie einen möglichst schnellen Algorithmus, der für eine Zahl  $x$  entscheidet, ob diese in der Matrix  $A$  enthalten ist. Analysieren Sie Ihren Algorithmus hinsichtlich Zeit- und Speicherbedarf. Sie erhalten bis zu einem halben Punkt für einen Algorithmus mit  $O(n^2)$  Zeit, bis zu 5 Punkte für Lösungen mit  $O(n \log n)$  Zeit und bis zu 6 Punkte für Lösungen mit linearer Laufzeit.

*Viel Erfolg!*