

| | | | |
|----------|-------|------------|--|
| Name | | Matr.-Nr.: | |
| Σ | Note: | | |

DI Stefan Klampfl
04.03.2010 – Gruppe B

Prüfung zur Lehrveranstaltung 708.031 Datenstrukturen und Algorithmen

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 beträgt 90 Minuten.

1. Asymptotische Schranken (10 Punkte)

- a.) Definieren Sie in **eigenen**, klaren Worten **und** einer mathematischen Formulierung sowohl die Ω -Notation als auch die Θ -Notation (mit Skizze!).
- b.) Lösen Sie die folgende rekursive Zeitgleichung durch iteratives Einsetzen: $T(n) = 4T(\frac{n}{2}) + n^2$, mit $T(1) = O(1)$.
- c.) Geben Sie eine Funktion $f(n)$ an, sodass weder $f(n) = O(n \log n)$ noch $n^2 = O(f(n))$ ist. Beweisen Sie Ihre Behauptung!
- d.) Beweisen oder widerlegen Sie: Die Funktion $g(n) = e^{n\pi}$ ist $\Omega(6n^3)$.

2. Sortieralgorithmen (10 Punkte)

- a.) Was sind vergleichsbasierte Sortieralgorithmen? Erklären Sie ausführlich das Entscheidungsbaummodell und leiten Sie damit die untere Laufzeitschranke für vergleichende Sortierverfahren ab.
- b.) Erklären Sie die Funktionsweise von MergeSort in eigenen Worten. Geben Sie einen Pseudocode an und führen Sie eine Laufzeitanalyse durch. Wie groß ist der Speicherverbrauch dieses Algorithmus?
- c.) Was bedeuten die Begriffe *in-place* und *worst-case optimal* im Zusammenhang mit Sortierverfahren?

3. Optimale Kodierung (10 Punkte)

- a.) Erklären Sie die Begriffe *Entropie* und *präfix-frei* im Zusammenhang mit Kodieren.
- b.) Folgender Text ist zu kodieren: "WAIKIKI_LIN_HAWAII". Ermitteln Sie dazu eine Häufigkeitstabelle (Achtung: auch das Leerzeichen "_" ist ein Zeichen!) und konstruieren Sie daraus **Schritt für Schritt** den optimalen Codebaum (nach Huffman). Wieviele Bits können bei diesem Text gegenüber einem Code mit je 3 Bits/Zeichen eingespart werden?
- c.) Schreiben Sie einen Pseudocode für die Implementierung der Huffman-Kodierung mit Halden (mit Erklärung!) und leiten Sie die Laufzeit $T(n)$ ab.

4. Richtig oder Falsch (10 Punkte)

Stimmen die folgenden Aussagen? Beachten Sie, dass es nur bei richtiger Antwort **mit** richtiger Begründung Punkte gibt.

- a.) Eine ideale Hashfunktion liefert immer einen Index zurück, welcher zu keiner Kollision führt.
- b.) Alle Funktionen der Warteschlange mit Prioritäten, mit Hilfe von Halden implementiert, haben eine Laufzeit von $O(\log n)$.
- c.) Die Interpolationssuche ist ordnungsmäßig manchmal schneller als FastSearch.
- d.) Bei Hashtabellen mit Überlauferlisten kommt es immer zu Kollisionen, wenn der Belegungsfaktor $\alpha \geq 1$.
- e.) Bei einem Binärbaum, der in Nebenreihenfolge aufsteigend sortiert ist, steht das Maximum immer in der Wurzel.

Viel Erfolg!