Prof. Dr. Christian Sohler
Nicole Funk, Dr. Matthijs Ebbens, Sebastian Zaun

Sommersemester 2023

4. Übungsblatt

zur Vorlesung

Grundzüge der Informatik I

Abgabe über Ilias bis zum 3.5. 14:00 Uhr.
Besprechung in Kalenderwoche 19.

Aufgabe 1 *Binäre Suche (2 Punkte)*

Betrachten Sie das folgende Feld von Zahlen:

$$A = [6, 3, 36, 99, 2, 39, 12, 45, 23, 17, 24]$$

Strukturieren Sie die Elemente in dem Feld um, sodass Binäre Suche anwendbar ist.

Führen Sie anschließend die aus der Vorlesung bekannte Methode BinäreSuche(A,12,1,11) aus, um in dem Feld $A[1, \ldots, 11]$ nach dem Wert 12 zu suchen und geben Sie die Ausgabe, sowie die besuchten Teilfelder des Algorithmus an.

Aufgabe 2 *Teile und Herrsche (4 + 2 + 3 Punkte)*

Gegeben sei ein aufsteigend sortiertes Feld A aus n natürlichen Zahlen. Bei der Übertragung des Feldes ist ein Fehler unterlaufen. Die Diagnose des Fehlers besagt, dass es nach der Übertragung maximal einen Wert gibt, welcher verfälscht wurde. Dies bedeutet, dass die betroffene Stelle des Feldes einen größeren Wert als ursprünglich haben kann. Dadurch kann es sein, dass das Feld nach der Übertragung nicht mehr aufsteigend sortiert ist. Also kann binäre Suche auf dem Feld nicht mehr angewandt werden.

Trotz dieses Fehlers sollen aber alle unverfälschten Werte von Feld A in Laufzeit $O(\log n)$ gesucht werden können.

a) Entwickeln Sie einen Teile und Herrsche Algorithmus, welcher das Feld A, dessen Länge n und eine natürliche Zahl x bekommt, und in Laufzeit $O(\log n)$ die Position von x im Feld A zurückgibt, sofern x ein unverfälschter Wert ist. Sollte x durch den Übertragungsfehler in A entstanden sein, kann der Algorithmus in Laufzeit $O(\log n)$ die Position von x ausgeben, oder einen Fehler ausgeben.

b) Beginnen Sie die Laufzeitanalyse Ihres Algorithmus, indem Sie eine Rekursionsgleichung für die Laufzeit herleiten und angeben.

c) Beweisen Sie die Korrektheit Ihres Algorithmus.

Aufgabe 3 *Rekursionsgleichungen (3 + 3 + 3 Punkte)*

Geben Sie für die folgenden Rekursionsgleichungen eine möglichst einfache und langsam wachsene Funktion $g(n)$ an, sodass $T(n) \in O(g(n))$ ist.

Gehen Sie dabei wie folgt vor:

Lösen Sie die Rekursionsgleichungen auf, um eine nicht rekursive Funktion zu erhalten, welche die Laufzeit in Abhängikeit von n beschreibt und beweisen Sie deren Korrektheit mittels Induktion. Drücken Sie nun Ihre bewiesene Funktion durch O-Notation aus.

a) $T(n) = \begin{cases} 1 & n = 1 \\ 1 \cdot T(\frac{n}{4}) + 4 & \text{sonst} \end{cases}$

b) $T(n) = \begin{cases} 1 & n = 1 \\ 2 \cdot T(\frac{n}{2}) + n^2 & \text{sonst} \end{cases}$

c) $T(n) = \begin{cases} 1 & n = 1 \\ 3 \cdot T(\frac{n}{3}) + n & \text{sonst} \end{cases}$