

5. Übungsblatt

zur Vorlesung

Grundzüge der Informatik I

Abgabe über Ilias bis zum 10.5. 14:00 Uhr.
Besprechung in Kalenderwoche 20.

Aufgabe 1 *Mastertheorem (2 + 2 + 2 + 2 Punkte)*

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

$$\text{a) } T(n) = \begin{cases} 12 & n = 1 \\ 2 \cdot T(\frac{n}{2}) + 12 & \text{sonst} \end{cases}$$

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

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

$$\text{d) } T(n) = \begin{cases} 1 & n = 1 \\ 2 \cdot T(\frac{n}{4}) + \sqrt{n} & \text{sonst} \end{cases}$$

Geben Sie in allen Teilaufgaben an, welchen Teil des Theorems Sie nutzen, und überprüfen Sie stets, ob die nötige Voraussetzung erfüllt ist.

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

Wir betrachten den Aktienkurs der Firma *Informatik AG*. Der Wert einer Aktie über n Zeiteinheiten wird dabei in dem Array A gespeichert. Wir möchten Wissen zu welchen Zeiten wir die Aktie kaufen und verkaufen sollten, um den maximalen Gewinn zu erzielen. Kauf und Verkauf dürfen dabei aber nur einmal passieren.

Formal ist also ein Array $A[1..n]$ von positiven ganzen Zahlen gegeben und wir wollen die größte Wertdifferenz $A[i] - A[j]$ für $j \leq i$ bestimmen.

- a) Betrachten Sie zwei benachbarte Teilarrays $A[i..j]$ und $A[j + 1..\ell]$ mit $1 \leq i \leq j < \ell \leq n$. Welche Informationen über die Teilarrays werden benötigt, um die maximale Differenz zweier Elemente in $A[i..\ell]$ in konstanter Zeit zu berechnen? Erklären Sie auch wie diese Berechnung funktioniert.
- b) Entwickeln Sie einen Teile-und-Herrsche Algorithmus, der Ihre Überlegungen aus Aufgabenteil a) verwendet, um die größte Wertdifferenz zu berechnen. Geben Sie eine Implementierung Ihres Algorithmus in Pseudocode an und kommentieren Sie diesen. Für die volle Punktzahl wird ein Algorithmus erwartet, dessen Worst-Case-Laufzeit durch $\mathcal{O}(n)$ beschränkt ist.
- c) Analysieren Sie die Laufzeit Ihres Algorithmus. Stellen Sie hierzu eine Rekursionsgleichung für die Laufzeit Ihres Algorithmus auf und lösen Sie diese. Sie dürfen das Mastertheorem verwenden.
- d) Zeigen Sie die Korrektheit Ihres Algorithmus.