

Übungen zur Vorlesung  
**Algorithmen und Datenstrukturen**  
WiSe 2018/19  
Blatt 3

Wichtige Hinweise:

- > Falls Sie bei der Bearbeitung einer Aufgabe größere Schwierigkeiten hatten und deswegen die Bearbeitung abgebrochen haben, so versuchen Sie bitte Ihre Schwierigkeiten in Form von Fragen festzuhalten. Bringen Sie Ihre Fragen einfach zur Vorlesung oder zur Übung mit!
- > Kursraum: <https://elearning.uni-regensburg.de/course/view.php?id=9228>

**Aufgabe 1:**

Sei  $f(n)$  die  $n$ .te Fibonacci-Zahl, die wie folgt definiert ist:

$$f : \mathbb{N} \rightarrow \mathbb{N} \text{ mit } f(1) = 1, f(2) = 1 \text{ und } f(n) = f(n-1) + f(n-2) \text{ für } n \geq 3$$

Zeigen Sie mit Hilfe vollständiger Induktion, dass gilt:

$$f(n) = \frac{\phi^n - \hat{\phi}^n}{\sqrt{5}} \text{ mit } \phi = \frac{1 + \sqrt{5}}{2}, \hat{\phi} = \frac{1 - \sqrt{5}}{2}$$

Folgern Sie, dass eine rekursive Implementierung zur Berechnung der Fibonacci-Zahlen die Laufzeit  $T(n) = \Theta(\phi^n)$  hat.

**Aufgabe 2:**

Entwerfen und implementieren Sie einen Algorithmus mit Laufzeit  $O(n^3)$  für das MaxTeilSum-2d-Problem:

- Eingabe:  $a_{ij}$  mit  $a_{ij} \in \mathbb{Z}$  für  $i, j \in \{1, \dots, n\}, n \in \mathbb{N}$  ( $n^2$  ganze Zahlen)
- Ausgabe:

$$s = \max_{1 \leq i_1 \leq i_2 \leq n, 1 \leq j_1 \leq j_2 \leq n} \sum_{i_1 \leq i \leq i_2, j_1 \leq j \leq j_2} a_{ij} \text{ (maximale 2-dimensionale Teilsumme)}$$

**Aufgabe 3:**

Beweisen Sie folgende Aussagen mit Hilfe der Master-Methode:

1. Sei  $T(1) = 1, T(n) = T(n/2) + 1$  für alle  $n > 1$ , dann:  $T(n) = \Theta(\log n)$  (Binäre Suche)
2. Sei  $T(1) = 1, T(n) = 3T(n/4) + n \log n$  für alle  $n > 1$ , dann:  $T(n) = \Theta(n \log n)$

3. Sei  $T(1) = 1, T(n) = 7T(n/2) + n^2$  für alle  $n > 1$ , dann:  $T(n) = \Theta(n^{2,81})$

**Aufgabe 4:**

Demonstrieren Sie die Funktionsweise der Sortieralgorithmen InsertionSort, BubbleSort und SelectionSort anhand des Feldes  $a[] = \{-5, 13, -32, 7, -3, 17, 23, 12, -35, 19\}$ . Überprüfen Sie Ihre Demonstration mit Hilfe eines C, C++, Java oder C#-Programms, das die einzelnen Sortierv Verfahren implementiert und die wesentlichen Zwischenfolgen zum Vergleich ausgibt.