



Prof. Dr. Sebastian Wild
Dr. Nikolaus Glombiewski

Übungen zur Vorlesung
Effiziente Algorithmen

Präsenzübung 0

Aufgabe: Komplexität

a) Beweisen oder widerlegen Sie die folgenden Aussagen:

i) $\frac{1}{6}n^3 + \frac{1}{2}n^2 + \frac{1}{3}n \in \mathcal{O}(n^3)$

ii) $4n^3 + 2n + 1 \in \Theta(6n^3 + n + 12)$

iii) $n \log_2(n) \in \mathcal{O}(n^2)$

iv) $n \cdot \sqrt{\log n} \in \Omega(n \cdot \log(n^2))$

v) Aus $f(n) \in \Theta(n)$ folgt $2^{f(n)} \in \mathcal{O}(2^n)$

b) Implementieren Sie den in der Vorlesung vorgestellten Algorithmus *Bubblesort* in Java für ein ganzzahliges Array (`int []`).

Testen Sie die Laufzeit Ihrer Implementierung, indem Sie ein zufälliges Array der Größe n generieren und anschließend den Algorithmus auf das Array anwenden.

Führen Sie das Experiment mehrfach aus und variieren Sie n , indem Sie die Größe schrittweise verdoppeln. Vergleichen Sie anschließend die Ergebnisse mit der Methode `Arrays.sort` der Java Standardbibliothek.