



Algorithmen und Datenstrukturen

Sommersemester 2022

Übungsblatt 2

Abgabe: Dienstag, 10. Mai, 2022, 10:00 Uhr

Aufgabe 1: \mathcal{O} -Notation

(9 Punkte)

Beweisen oder widerlegen Sie die folgenden Aussagen anhand der *Mengendefinition* der \mathcal{O} -Notation (Vorlesungsfolien Woche 2, Folie 11 und 12).

- (a) $2n^3 + 4n^2 + 7\sqrt{n} \in \mathcal{O}(n^3)$ (1 Punkt)
- (b) $n \cdot \log_3(n) \in \omega(n \cdot \log_5(n))$ (2 Punkte)
- (c) $2^n \in o(n!)$ (2 Punkte)
- (d) $2 \log_2(n^2) \in \Omega((\log_2 n)^2)$ (2 Punkte)
- (e) $\max\{f(n), g(n)\} \in \Theta(f(n) + g(n))$ für nicht negative Funktionen f und g . (2 Punkte)

Aufgabe 2: Sortieren nach Asymptotischem Wachstum (6 Punkte)

Sortieren Sie folgende Funktionen nach asymptotischem Wachstum. Schreiben Sie $g <_{\mathcal{O}} f$ falls $g \in \mathcal{O}(f)$ und $f \notin \mathcal{O}(g)$. Schreiben Sie $g =_{\mathcal{O}} f$ falls $f \in \mathcal{O}(g)$ und $g \in \mathcal{O}(f)$ (kein Beweis nötig).

\sqrt{n}	2^n	$n!$	$\log(n^3)$
3^n	n^{100}	$\log(\sqrt{n})$	$(\log n)^2$
$\log n$	$10^{100}n$	$(n+1)!$	$n \log n$
$2^{(n^2)}$	n^n	$\sqrt{\log n}$	$(2^n)^2$

Aufgabe 3: Stabiles Sortieren

(5 Punkte)

Ein Sortieralgorithmus heißt stabil, falls Elemente mit gleichem Schlüssel ihre ursprüngliche Reihenfolge beibehalten. Angenommen man möchte folgende Tupel bzgl. deren ganzzahligen Schlüssel sortieren.

[(3, "blue"), (1, "green"), (1, "red"), (7, "gray"), (4, "yellow"), (3, "orange"), (4, "white"), (3, "black")]

Ein *stabiler* Sortieralgorithmus muss dann folgenden Output generieren:

[(1, "green"), (1, "red"), (3, "blue"), (3, "orange"), (3, "black"), (4, "yellow"), (4, "white"), (7, "gray")]

Ein Sortieralgorithmus wäre nicht stabil (bzgl. diesem Sortierschlüssel), wenn er z.B. Folgendes ausgibt:

[(1, "red"), (1, "green"), (3, "black"), (3, "blue"), (3, "orange"), (4, "yellow"), (4, "white"), (7, "gray")]

- (a) Der Sortieralgorithmus `QuickSort` ist *nicht* stabil. Zeigen Sie dies anhand eines geeigneten Beispiels. (1 Punkt)
- (b) Beschreiben Sie eine Methode, welche jeden vergleichsbasierten Sortieralgorithmus stabil macht ohne die *asymptotische* Laufzeit zu ändern. Begründen Sie warum. (4 Punkte)