**Theoretische Informatik**
D. Flumini, L. Keller, O. Stern

Zürcher Hochschule
für Angewandte Wissenschaften

# Übungsblatt 9

## Berechenbarkeit / Komplexitätstheorie

Abgabe: Kalenderwoche 21

**Aufgabe 1.**
Ist das Komplement $\overline{H}$ des Halteproblems semi-entscheidbar? Begründen Sie Ihre Antwort.

**10 Punkte**

**Aufgabe 2.**
Lösen Sie die folgenden Aufgaben zur $\mathcal{O}$-Notation:

(a) Geben Sie an ob $2n(5n + 3\log(n)) \in \mathcal{O}(n^2)$ gilt und begründen Sie.

(b) Begründen Sie, warum $13 \cdot 2^{n+5} \in \mathcal{O}(2^n)$ gilt.

(c) Begründen Sie, warum $n^2 \cdot 3^n \notin \mathcal{O}(n^5)$ gilt.

**9 Punkte**

**Aufgabe 3.**
Sortieren Sie die nachfolgende Funktionen gemäs ihrer $\mathcal{O}$-Klassen.

$f_1(n) = n^2 \cdot \log_2(n) + 42$
$f_2(n) = 2^n + n^4$
$f_3(n) = 2^{2 \cdot n}$
$f_4(n) = 2^{n+3}$
$f_5(n) = \sqrt{n^5}$

Es ist ausschliesslich die Nutzung der echten Teilmenge $\subset$ sowie die Gleichheit $=$ für die Beziehung zwischen den Mengen gültig. Die Beziehungen müssen weder begründet noch bewiesen werden.

Die Lösung soll nachfolgende Form aufweisen (Die Angaben haben nichts mit den oben angegebenen Funktionen zu tun):

$$\mathcal{O}(y_1(n)) = \mathcal{O}(y_2(n)) \subset \mathcal{O}(y_3(n)) = \mathcal{O}(y_4(n)) \subset \mathcal{O}(y_5(n))$$

**10 Punkte**

**Aufgabe 4.**
Begründen oder widerlegen Sie folgende Aussagen.

(a) Es gibt eine Funktion $f(n)$ und $g(n)$, so dass gleichzeitig $f(n) \in \mathcal{O}(g(n))$ und $f(n) \in \Omega(g(n))$ gilt.

(b) Es gibt voneinander verschiedene Funktionen $f(n)$ und $g(n)$, so dass $f(n) \in \mathcal{O}(n \cdot \log(n))$ und $g(n) \in \mathcal{O}(n \cdot \log(n))$ gilt.

(c) Ein Algorithmus welcher Daten in $\mathcal{O}(n \cdot \log(n))$ sortiert, sollte in allen Fällen einem Algorithmus mit $\mathcal{O}(n^2)$ vorgezogen werden.

**9 Punkte**

**Aufgabe 5.**
Das Untersummen-Problem ist wie folgt definiert:

**Gegeben:** Natürliche Zaheln $a_1, ..., a_n$ und $b$.

**Ausgabe:** Die Ausgabe ist JA, falls es eine Teilmenge $S \subseteq \{ a_1, ..., a_n \}$ gibt, so dass $\sum_{a \in S} = b$ gilt, und NEIN sonst.

(a) Beschreiben Sie informell einen Polynomzeit-Verifizierer für dieses Problem.

(b) Bestimmen Sie die Zeitkomplexität in $\mathcal{O}$-Notation dieses Verifizierers, wenn die Eingabe unär-codiert vorliegt (wie in der Vorlesung und den Übungen zu den Turingmaschinen besprochen).

**10 Punkte**