# Datenstrukturen und Algorithmen

Altinger Harald
0630936

Note Title 22.10.2007

1. Die klein-$o$ Notation ist wie folgt definiert: $f(n) = o(g(n))$ wenn gilt $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$. Zeigen Sie daß $n^a = o(c^n)$ für alle $a > 0$ und $c > 1$.

$$n^a = o(c^n) \quad a > 0 \quad c > 1$$

$$\lim_{n\to\infty} \frac{n^a}{c^n} = \frac{\overbrace{n \cdot n \cdots n}^{a}}{\underbrace{c \cdot c \cdots c}_{n}} \to \frac{a \cdot n^{a-1}}{c^n \, \ln(c)}$$

De l'Hospital

$$\to \frac{[a^2] \, n^{a-2}}{c^n \cdot \ln(c)}$$ entsprechend oft differenzieren ($a$ mal)

$$\to \frac{[a^a] n^0}{(c^n \cdot \ln(c))^a} \xrightarrow[n\to\infty]{} 0 \quad \wedge \quad \frac{[a^a] \cdot n^0}{(c^n \cdot \ln(c)^a)} \to 0$$ über Einzwickung

2. Beweisen oder widerlegen Sie: Aus $f(n) = O(g(n))$ und $g(n) = O(h(n))$ folgt $f(n) = O(h(n))$.

$$f(n) = O(g(n)) = O(O(h(n)))$$

$$\left( \left. \begin{array}{l} n = O(n^2) \\ n^2 = O(n^3) \end{array} \right\} \Rightarrow n = O(n^3) \right)$$

$$f(n) \le c_1 \cdot g(n)$$

$$g(n) \le c_2 \cdot h(n) \Rightarrow f(n) \le \underbrace{c_1 \cdot (c_2}_{c} \cdot h(n)) \Rightarrow f(n) \le c \cdot h(n)$$

3. Zeigen Sie eine obere Schranke für folgende rekursive Zeitgleichung durch iteratives einsetzen:
$T(n) = T(n-a) + n$,
$T(n) = O(1)$ für $n \leq 1$,
wobei $a \in \mathbb{N}, a \geq 1$ eine Konstante ist.

Beispiel:

$T(n) = T(n-a) + n \quad | a = 1$

$T(1) = \Theta(1)$

$T(2) = T(2-1) + 2$

$= \Theta(1) + 2$

$T(3) = T(3-1) + 3$

$= \Theta(1) + 2 + 3 = \Theta(1) + 5$

$T(4) = T(4-1) + 4$

$= \Theta(1) + 5 + 4$

$T(n) = \Theta(1) + \left(\sum_{i=1}^{n} i\right) - 1$

$= \Theta(1) + \frac{n(n+1)}{2} - 1 = \Theta(n^2)$

Allgemein:

$T(n) = T(n-a) + n$

$T(n-a) = T((n-a)-a) + (n-a)$

$= T(n-2a) + n - a$

$T(n-2a) = T((n-2a)-a) + (n-2a)$

$= T(n-3a) + n - 2a$

$T(n-3a) = T((n-3a)-a) + (n-3a)$

$$= T(n-4a) + n - 3a$$

$$T(n-4a) = T((n-4a)-a) + (n-4a)$$

$$= T(n-5a) + n - 4a$$

$$\Rightarrow T(\underbrace{n-(k+1)a}) + \underbrace{n-ka}_{\Theta(1)}$$

$$n-(k+1)a \overset{!}{=} 1 \Rightarrow k = \frac{n-1}{a} - 1$$

$$k = \frac{n-1-a}{a}$$

$$T\left(n - \left(\frac{n-1}{a} - 1\right)a\right) =$$

$$T(\cancel{n} - (\cancel{n}-1-a) = T(-1-a) \; ?$$

4. Zeigen Sie eine obere Schranke für folgende rekursive Zeitgleichung durch iteratives einsetzen:
$T(n) = T(\lfloor\sqrt{n}\rfloor) + 1,$
$T(n) = O(1)$ für $n \leq 1$.
Hierbei ist $\lfloor\cdot\rfloor$ der floor-operator, d.h. $\lfloor x \rfloor = \max\{y \in \mathbb{N} | y \leq x\}$.

$$T(n) = \Theta(1) \quad \forall n \leq 1$$

$$T(n) = T(\lfloor\sqrt{n}\rfloor) + 1 \quad \Rightarrow \quad T(\lfloor n^{\frac{1}{2}} \rfloor) + 1$$

$$T(\lfloor\sqrt{n}\rfloor) = T(\lfloor\sqrt{\sqrt{n}}\rfloor \quad \Rightarrow \quad T(\lfloor n^{\frac{1}{4}} \rfloor) + 1$$

$$T(\lfloor n^{\frac{1}{4}} \rfloor) = T(\lfloor\sqrt{\sqrt{\sqrt{n}}}\rfloor) \Rightarrow T(n^{\frac{1}{8}}) + 1$$

$$T(\lfloor n^{\frac{1}{2^3}} \rfloor) + 1$$

$$T(\lfloor n^{\frac{1}{2^k}} \rfloor) + 1$$