

Prof. Dr. Sándor Fekete
 Nils Schweer

Algorithmen und Datenstrukturen Übung 3 vom 02.12.2009

Abgabe der Lösungen am Mittwoch, den 16.12.09, vor der Abteilung *Algorithmik*.
 Bitte die Blätter vorne deutlich mit eigenem **Namen** und **Gruppennummer** versehen!

Aufgabe 1 (O-Notation):

In der Vorlesung wurde die O -Notation zur Abschätzung der Laufzeit eingeführt.

- a) Suche für die folgenden Funktionen jeweils geeignete Konstanten c (bzw. c_1 und c_2) und n_0 und zeige mit Hilfe dieser Konstanten, dass die jeweilige Funktion in der angegebenen Klasse liegt.

$$f_1(n) = \frac{n^{14}}{4^n} \in O(1)$$

$$f_2(n) = 2n^2 + 3n + 1 \in O(n^3)$$

$$f_3(n) = \sum_{i=1}^n i \in \Theta(n^2)$$

- b) Kreuze an, in welchen Klassen die jeweilige Funktion liegt.

$f(n)$	$O(1)$	$O(n)$	$O(2^n)$	$\Omega(1)$	$\Omega(n)$	$\Omega(2^n)$	$\Theta(1)$	$\Theta(n)$	$\Theta(2^n)$
42									
$5n + 100$									
$n \log n$									
n^n									
$\sum_{i=0}^n q^i, q < 1$									

(9+7 Punkte)

Aufgabe 2 (O-Notation):

Beweise Satz 3.12 aus der Vorlesung: Seien $f, g : \mathbb{N} \mapsto \mathbb{R}$ zwei Funktionen. Dann gilt:

- (i) $f \in \Theta(g) \Leftrightarrow g \in \Theta(f)$
- (ii) $f \in \Theta(g) \Leftrightarrow f \in O(g)$ und $f \in \Omega(g)$
- (iii) $f \in O(g) \Leftrightarrow g \in \Omega(f)$

(5+5+5 Punkte)

Aufgabe 3 (O-Notation):

Gegeben seien folgende zwei Algorithmen:

- Algorithmus 1 mit einer Laufzeit von $O(f(n))$
- Algorithmus 2 mit einer Laufzeit von $O(g(n))$

a) Sei der Algorithmus 3 wie folgt definiert:

```
BEGIN  
  Führe Algorithmus 1 aus  
  Führe Algorithmus 2 aus  
END
```

Zusätzlich gelte $f(n) \in \Omega(g(n))$.

Welche Laufzeit hat der Algorithmus 3? Nutze dabei aus, dass $f(n) \in \Omega(g(n))$ gilt. Beweise Deine Aussage!

b) Sei der Algorithmus 4 wie folgt definiert:

```
BEGIN  
  IF  $n \geq 30$   
    Führe Algorithmus 1 aus  
  ELSE  
    Führe Algorithmus 2 aus  
  END IF  
END
```

Welche Laufzeit hat der Algorithmus 4? Beweise Deine Aussage!

(8+8 Punkte)

Aufgabe 4 (Datenstrukturen):

- a) In der Vorlesung wurden die Operationen LIST-INSERT(L,x) und LIST-DELETE(L,x) für doppelt verkettete Listen vorgestellt. Können diese Operation für einfach verkettete Listen auch mit einer Laufzeit von $O(1)$ durchgeführt werden?
- b) Seien S_1 und S_2 zwei Mengen von Elementen. Die Operation UNION soll diese beiden Mengen vereinigen und $S_1 = S_1 \cup S_2$ zurückgeben. Verwende eine geeignete Datenstruktur, so dass die Operation in $O(1)$ ausgeführt werden kann. (Gib die Operation UNION in Pseudocode an.)

(6+7 Punkte)