

Übungsblatt 3

Abgabe bis Dienstag, den 16. Mai um 12:00 Uhr

Aufgabe 1 (4 Punkte)

Zeigen Sie, dass $\log_a n = \Theta(\log_b n)$ für alle $a, b > 1$. Benutzen Sie dabei für den Beweis direkt die Definitionen aus Vorlesung 3a. Sie dürfen für diese Aufgabe *nicht* mit Hilfe von Grenzwerten argumentieren.

Aufgabe 2 (2 + 2 + 2 + 2 = 8 Punkte)

Gegeben die folgenden fünf Funktionen von $\mathbb{N} \rightarrow \mathbb{R}^+$:

$$n \mapsto n \cdot \log_2 n, \quad n \mapsto n, \quad n \mapsto 2^n, \quad n \mapsto n^2, \quad n \mapsto n^{\log_2 3}$$

Ordnen Sie die Funktionen in der Reihe „aufsteigender Größenordnung“ an, also so in einer Reihe f_1, f_2, f_3, f_4, f_5 , dass gilt $f_i = O(f_{i+1})$, für $i = 1, \dots, 4$. Bestimmen Sie dabei außerdem für jedes $i = 1, \dots, 4$, ob nur gilt $f_i = O(f_{i+1})$ oder auch $f_i = \Theta(f_{i+1})$.

Wichtiger Hinweis: Sie können dabei wahlweise über die Definitionen aus Vorlesung 3a oder über den Grenzwert, wie in Vorlesung 3b erklärt, argumentieren. Allerdings sollten sie für mindestens ein i über die Definitionen argumentieren und für mindestens ein i über den Grenzwert.

Aufgabe 3 (2 + 3 + 3 = 8 Punkte)

Beschreiben Sie in einem Satz, was die folgende Funktion berechnet und bestimmen Sie die Laufzeit als $\Theta(\dots)$ in Abhängigkeit von n , wobei das \dots so einfach wie möglich sein sollte. Dabei ist hier und im Folgenden n stets eine natürliche Zahl.

```
def quad(n):  
    result = 0  
    for i in range(0, n):  
        for j in range(i, n):  
            result += 1  
    return 2 * result - n
```

[bitte in Zeit $\Theta(n^{1/\log n})$ wenden]

Tun Sie dann das Gleiche für die folgende Funktion.

```
def id(n):
    counts = [0] * n
    for i in range(0, n):
        counts[random.randint(0, n-1)] += 1
    result = 0
    for x in range(0, len(counts)):
        for i in range(0, counts[x]):
            result += 1
    return result
```

Und auch für die folgende Funktion! Die Funktion *index* sucht dabei den gegebenen Wert in dem gegebenen Feld und liefert den Index zurück, bei dem der Wert (zuerst) gefunden wurde.

```
def rev(n):
    array = [0] * n;
    for i in range(0, n):
        array[i] = i + 1
    for i in range(0, n):
        idx = array.index(n - i)
        array[idx] = i + 1;
    return array
```

Wichtiger Hinweis: sämtliche Aufgaben auf diesem Übungsblatt lassen sich elegant und mit wenig Schreibaufwand lösen. Wenn Sie anfangen, seitenlang zu rechnen, denken Sie zu kompliziert oder übersehen einen einfacheren Weg oder haben etwas noch nicht verstanden.

Committen Sie Ihre Lösung in das SVN, in einen neuen Unterordner *blatt-03*. Bezüglich des Formates gelten die Anweisungen vom letzten Übungsblatt. Beachten Sie auch bitte die Wünsche Ihres Tutors bzw. Ihrer Tutorin, falls er bzw. sie bei der letzten Korrektur welche geäußert hat.

Committen Sie in diesem Unterordner außerdem wie gehabt eine Textdatei *erfahrungen.txt*. Beschreiben Sie dort wie gewohnt in ein paar Sätzen Ihre Erfahrungen mit diesem Übungsblatt und den Vorlesungen dazu. Sie sollen keinen Roman schreiben, aber 2-3 Sätze mit nicht-trivialem Inhalt dürfen es schon sein.

Ist der Placebo-Effekt Einbildung?