

# Übungen zur Vorlesung Sequenzanalyse I

Universität Bielefeld, Wintersemester 2010/2011  
Dipl.-Inform Peter Husemann · Dr. Roland Wittler

<http://wiki.techfak.uni-bielefeld.de/gi/Teaching/2010winter/SequenzAnalyse>

**Blatt 1 vom 15.10.2010**

**Abgabe in einer Woche vor Beginn der Vorlesung.**

## Organisatorisches

1. Bitte registriere Dich im eKVV für die Vorlesung *und* für die Übungen. Weiterhin trage Dich in eine der Übungslisten ein, die am TechFak-Brett hängen, sofern Du das nicht heute in der Vorlesung gemacht hast.
2. Das Skript wird in der ersten Übung ausgeteilt. Auf der Homepage der Veranstaltung steht eine elektronische Version des Skriptes zum Download bereit.
3. Die Übungszettel werden in der Regel wöchentlich in der Vorlesung ausgegeben. Die Abgaben zu den Übungszetteln werden zu Beginn der nächsten Vorlesung eingesammelt oder können *vor* dieser Vorlesung in den Briefkasten bei U10-151 geworfen werden.
4. Auch wenn die Bearbeitung der Aufgaben zu zweit erfolgen kann, müssen die Lösungen zu den Aufgaben separat und unter eigenem Namen abgegeben werden.
5. Bitte den Tutorientermin bzw. den Tutor, sowie Deinen Namen deutlich auf der Abgabe vermerken. Zettel, die nicht zugeordnet werden können, werden auch nicht bewertet.
6. Die Tutoren dürfen Punkte wieder abziehen, wenn abgegebene Lösungen im Tutorium nicht vorgerechnet / vorgetragen werden können.
7. Erfolgreiches Lösen der Übungsaufgaben (Bestehensgrenze 50%) ist Voraussetzung für die Teilnahme an der Abschlussklausur oder der abschließenden mündlichen Prüfung (Portfolioleistung).  
Die A&D Prüfung muss ebenfalls vor der Abschlussprüfung bestanden sein.

**Aufgabe 1** Gegeben sind zwei Algorithmen  $A_1$  und  $A_2$ . Bei einer Eingabgröße von  $n$  braucht  $A_1$  die folgende Anzahl an Rechenschritten:  $f_1 = 3 \cdot (n^2 + 3)$ . Algorithmus  $A_2$  benötigt  $f_2 = 100 + n \log_2(n)$  Schritte. (2 Punkte)

1. Stelle die Funktionen grafisch in einem gemeinsamen Koordinatensystem dar und gib die Komplexitätsklasse der Algorithmen in der  $\mathcal{O}$ -Notation an.
2. Welcher Algorithmus hat die asymptotisch schlechtere Laufzeit? Ab welchem  $n_0$  wird das deutlich, mit der Vorgabe, dass  $c = 1$  ist?

**Aufgabe 2** Gegeben sei ein Alphabet  $\Sigma = \{0, 1\}$  zur Darstellung von Binärzahlen. (2 Punkte)

1. Wie ist  $\Sigma^8$  definiert?
2. Schreibe alle Worte  $w \in \Sigma^3$  auf.

**Aufgabe 3** Gegeben sei das Wort  $s = \text{mississippi}$ . (2 Punkte)

1. Gib alle Substrings der Länge 7, sowie eine Subsequenz der Länge 5 von  $s$  an.
2. Erweitere die folgende Menge  $M = \{\text{mississippi}\}$ , so dass sie suffixabgeschlossen ist.