Karlsruher Institut für Technologie
Institut für Theoretische Informatik

Prof. Dr. Jörn Müller-Quade
Björn Kaidel, Sascha Witt

1. Tutorenblatt zu Algorithmen I im SoSe 2017

https://crypto.iti.kit.edu/index.php?id=algo-sose17
{bjoern.kaidel,sascha.witt}@kit.edu

Im Folgenden findet ihr einige unverbindliche Vorschläge zur Gestaltung des ersten Tutoriums.

1. Vorstellungsrunde & Organisatorisches

Kurze Vorstellungsrunde. Wie heißen deine Tutanden und was studieren sie? Gebe deinen Tutanden auch ein paar Infos über dich (Name, Mail, Was studierst du? In welchem Semester?)! Hinweise zur Bearbeitung/Abgabe der Übungsblätter und ähnliches. Weiße deine Tutanden insbesondere auf die folgenden Möglichkeiten hin:

- Die Vorlesungswebsite `https://crypto.iti.kit.edu/index.php?id=799`
- Das ILIAS-Forum für Diskussionen `https://ilias.studium.kit.edu/goto.php?target=crs_671935&client_id=produktiv`
- Besonders wichtig für uns: Die Mailingliste `https://lists.ira.uni-karlsruhe.de/mailman/listinfo/algorithmeni`
- Den anonymen Feedbackkasten auf der Website
- Bitte daraufhinweisen, dass sich der Feedbackkasten nicht eignet, um Fragen zu stellen!

2. Wiederholung

Eine Kurzzusammenfassung der Themen am Anfang des Tutoriums, damit allen die verwendeten Begriffe klar sind. Diese sollte aber nicht zuviel Zeit beanspruchen, da das Tutorium überwiegend Raum für die Beantwortung von Fragen und die Bearbeitung von Aufgaben geben sollte. Für diese Woche:

- Pseudocode (Was ist Pseudocode, worauf sollte geachtet werden?)
- $\mathcal{O}/\Omega/\Theta$-Kalkül (evtl kurze Wiederholung der Definition mit Grenzwert und Definition Quantoren)
- Schleifeninvarianten: Was ist eine Schleifeninvariante? Wie beweist man die Korrektheit von Schleifeninvarianten (Beweisidee)?
- Master-Theorem
- (Für die Multiplikation mit dem Algorithmus von Karatsuba und Ofman könnt ihr auf die Übung verweisen.)

3. Aufgaben

Für Aufgaben könnt ihr z.B. folgende Algorithmenbücher aus der Bibliothek ausleihen: Corman et al., Manber, Sedgewick, Mehlhorn/Sanders...

Aufgabe 1 (*Pseudocode*)

Gebt die gewünschte Funktionsweise eines einfachen Algorithmus vor und lasst die Studenten den Algorithmus in Pseudocode aufschreiben. (Beispiel: Der Algorithmus soll n Studenten in zwei Gruppen einteilen, je nachdem ob die Quersumme der Matrikelnummer (Matrikelnummer als Array gegeben) gerade oder ungerade ist.) Geht durch die Reihen und schaut, ob die Verwendung von Pseudocode grundsätzlich klar ist. Den Studenten soll klar werden, dass es bei der Verwendung von Pseudocode auf die Struktur und nicht so sehr auf die exakte Syntax ankommt.

Aufgabe 2 (*Laufzeitverhalten*)

Bestimmung der Laufzeiten von verschiedenen Algorithmen (gegeben in Pseudocode) im $\mathcal{O}/\Omega/\Theta$-Kalkül (worst-case, best-case, eventuell average-case). Außerdem könnt ihr Fragen wie Korrektheit unter der Verwendung von Invarianten und ähnliches besprechen.

Aufgabe 3 (*Schleifeninvariante, Korrektheit*)

Stellt einen einfachen Algorithmus vor (**nicht** die binäre Suche, da sie auf dem zweiten Aufgabenblatt thematisiert wird) und behandelt die folgenden Punkte:

- Beweis der Korrektheit mittels Schleifeninvariante. (Für die totale Korrektheit muss auch bewiesen werden, dass der Algorithmus überhaupt terminiert. Außerdem ist wichtig, dass die Studenten verstehen, dass der Beweis der Invarianteneigenschaft häufig nichts anderes als eine vollständige Induktion ist. Die Wahl der Invariante, so dass aus ihr die Korrektheit der Schleife folgt, erfordert natürlich Kreativität. Auch das sollte den Studenten klar werden.)

Aufgabe 4 (*Logarithmus & Landau-Notation*)

Wiederholt ein paar grundlegende Eigenschaften des Logarithmus, da er für gewöhnlich im 2. Semester noch „etwas ungewohnt“ ist. Wendet diese Regeln dann auch in (einfachen) Landau-Notationsaufgaben an.

$$\begin{aligned}\text{Regeln: } & \log(a \cdot b) = \log(a) + \log(b), \log(a/b) = \log(a) - \log(b), 1 = \log_a(a), a^{\log_a(b)} = b, ...\\ \text{Aufgaben: } & \log(10 \cdot n) \in \mathcal{O}(\log(n))\,, n^n \in \Theta\left(2^{n\log(n)}\right), ...\end{aligned}$$

Aufgabe 5 (*Landau-Notation*)

Beweise oder widerlege:

- $n \in \Theta(\sqrt{n})$, $\quad n^2 \in \mathrm{o}(n^3)$, $\quad n^3 \in \Omega\left(n^2\right)$ $\quad 2^{n+1} \in \Theta(2^n)$, $\quad \ldots$
- $f \sim g :\Leftrightarrow f \in \mathcal{O}(g)$ definiert eine Äquivalenzrelation auf der Menge der Funktionen $\mathbb{N} \to \mathbb{N}$
- $f \sim g :\Leftrightarrow f \in \Theta(g)$ definiert eine Äquivalenzrelation auf der Menge der Funktionen $\mathbb{N} \to \mathbb{N}$
- $\forall c_1, c_2 \in \mathbb{N}\ \forall f\colon \mathbb{N} \to \mathbb{N}\colon c_1 \cdot f(n) + c_2 \in \mathcal{O}(f(n))$
- $\forall c \in \mathbb{N}\ \forall f\colon \mathbb{N} \to \mathbb{N}\colon\ (f(n))^c \in \mathcal{O}(f(n))$
- ...

Sonstige Hinweise/Anmerkungen

Bitte gebt uns Rückmeldung, welche Aufgaben besondere Schwierigkeiten bereitet haben/welche Fehler gehäuft auftraten.
Die Übungspunkte werden auf ILIAS eingetragen, für jedes Tutorium haben wir dort einen eigenen Bereich angelegt. Björn hat euch auch schon als Kurstutor zum Kurs hinzugefügt. Falls nicht, bitte melden! Die Seite zu Algorithmen I findet Ihr unter `https://ilias.studium.kit.edu/goto.php?target=crs_671935&client_id=produktiv`. Bitte tragt die Punkte jeweils zeitnah ein.
Weiter gibt es einen Bereich, der für alle Tutoren (aber nicht für die Studierenden) zugänglich ist. Dort könnt ihr z.B. Material für das Tutorium austauschen, euch gegenseitig Fragen stellen und ähnliches.