Theoretische Grundlagen der Informatik

Tutorium 7 — Wintersemester 2014/15

Moritz Klammler — 2. Dezember 2014

Anmerkungen zum 2. Übungsblatt

- Aufgaben 1 & 2 (Automatenkonstruktion)
  - Es gab vor allem Flüchtigkeitsfehler.
  - Könnt ihr die Algorithmen in Software implementieren?
- Aufgaben 3 & 4 (Nerode-Relation)
  - Die Lösungen waren idR sehr gut.
- Aufgaben 5 & 6 (Turingmaschinen)
  - Bitte ignoriert Aufgabe 5.

Ergebnisse der Umfrage

Es wurden insgesamt 8 Fragbögen abgegeben.

Wie empfindest Du das Tutorium?

- 0 Ich lerne wenig, weil ich unterfordert bin und mich häufig langweile.
- 1 Ich lerne etwas, langweile mich aber gelegentlich.
- 5 Ich lerne viel. Das Niveau passt für mich.
- 2 Ich lerne etwas, aber es fällt mir gelegentlich schwer zu folgen.
- 0 Ich lerne wenig, weil ich überfordert bin und häufig nicht folgen kann.

Wie häufig sollten die folgenden Dinge im Tutorium eingesetzt werden?

Ding	seltener	gleich oft	häufiger
Stoffzusammenfassung	0	5	3
Beispiele vorrechnen	0	4	4
Einzelarbeit	1	7	0
Gruppenarbeit	3	5	0
Lernspiele	2	5	1

Wie häufig besuchst Du folgende Veranstaltungen aus TGI?

Veranstaltung	selten	gelegentlich	regelmäßig
Vorlesung	2	4	2
Übung	2	4	2
Tutorium	0	1	7

Was studierst Du?

6 Informatik

1 Informationswirtschaft

1 Mathematik

Was wolltest Du außerdem noch anmerken?

- Die VL zieht sich sehr in die Länge, das Tutorium fasst alles in gutem Tempo zusammen.
- Programmieren finde ich nicht so interessant, aber sonst eine gute Zusammenstellung von Beispielen.

Tagesthemen

- Polynomielle Transformation
- $\mathcal{NP}$-Vollständigkeit
- Satz von Cook

Many-to-One-Reduzierbarkeit

Sei Σ ein Alphabet und $B \subseteq \Sigma^*$ **(semi-)entscheidbar**, sowie $A \subseteq \Sigma^*$.

Wenn es eine **berechenbare** Funktion $f : \Sigma^* \to \Sigma^*$ gibt, sodass für alle $w \in \Sigma^*$ gilt

$$w \in A \Leftrightarrow f(w) \in B$$

dann ist auch A (semi-)entscheidbar.

Man schreibt diesfalls $A \leq_{\mathrm{m}} B$.

Polynomielle Transformation

Sei Σ ein Alphabet und $B \subseteq \Sigma^*$ **in Polynomialzit entscheidbar**, sowie $A \subseteq \Sigma^*$.

Wenn es eine **in Polynomialzit berechenbare** Funktion $f : \Sigma^* \to_{\mathrm{p}} \Sigma^*$ gibt, sodass für alle $w \in \Sigma^*$ gilt

$$w \in A \Leftrightarrow f(w) \in B$$

dann ist auch A in Polynomialzit entscheidbar.

Man schreibt diesfalls $A \leq_{\mathrm{p}} B$ oder $A \propto B$.

$\mathcal{NP}$-Vollständigkeit

Eine Sprache L ist **$\mathcal{NP}$-vollständig** genau dann wenn

- $L \in \mathcal{NP}$
- $\forall L' \in \mathcal{NP} : L' \leq_{\mathrm{p}} L$

Die Menge aller $\mathcal{NP}$-vollständigen Sprachen heißt $\mathcal{NPC}$.

Erfüllbarkeitsproblem der Aussagenlogik (SAT)

Gegeben eine Variablenmenge $\{v_1, \ldots, v_n\}$ für $n \in \mathbb{N}$ und eine Klauselmenge $\{c_1, \ldots, c_m\}$ für $m \in \mathbb{N}$ aus Disjunktionen von (potentiell negierten) Variablen.

Existiert eine Wahrheitsbelegung der Variablen, sodass alle Klauseln erfüllt werden?

Falls jede Klausel eine Disjunktion aus genau 3 (potentiell negierten) Variablen ist, spricht man von 3-SAT.

$$\underbrace{(v_1 \vee v_2 \vee \neg v_3)}_{c_1} \wedge \underbrace{(\neg v_2 \vee v_4 \wedge v_5)}_{c_2} \wedge \cdots \wedge \underbrace{(v_3 \vee \neg v_n \vee v_1)}_{c_m}$$

Satz von Cook

$$3\text{-SAT} \in \mathcal{NPC}$$

Korollar Um zu zeigen, dass eine Sprache $L \in \mathcal{NP}$ auch $\mathcal{NP}$-vollständig ist, reduzieren wir 3-SAT polynomiell auf L (dh wir zeigen 3-SAT $\leq_p L$).