

Einführung in die Theoretische Informatik

Sommersemester 2023 – Übungsblatt 12

- Das Übungsblatt ist in zwei Teile gegliedert: den Vorbereitungsteil, den Sie vor der Übung selbstständig bearbeiten sollen, und den Übungs-/Nachbereitungsteil, der Aufgaben enthält, die in der Übung besprochen werden und von Ihnen anschließend zur Nachbereitung verwendet werden können.
- Das ist nicht das Hausaufgabenblatt! Die Hausaufgaben finden Sie auf einem separaten Blatt.

Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

Individualaufgabe Ü12.1. (Wichtige Begriffe)

Überprüfen Sie, dass Sie die folgenden Begriffe oder Notationen korrekt definieren können.

- Postsche Korrespondenzproblem (PCP)
- $time_M(w)$ und $ntime_M(w)$
- $TIME(f(n))$ und $NTIME(f(n))$
- Landau-Notation (oder auch \mathcal{O} -Notation): $\mathcal{O}(n), \mathcal{O}(2^n), \dots$
- P und NP
- $P \stackrel{?}{=} NP$
- Zertifikat und polynomiell beschränkter Verifikator
- NP-hart (oder auch auch: “NP-schwer”)
- NP-vollständig
- polynomielle Reduktion
- $A \leq_p B$ (sprich: “A ist polynomiell reduzierbar auf B”)
- Die Probleme: SAT und 3KNF-SAT

Individualaufgabe Ü12.2. (Unentscheidbare Typ-2 Probleme)

In der Vorlesung haben wir, beispielsweise, folgende Probleme für kontextfreie Grammatiken G_1, G_2 über einem Alphabet Σ kennengelernt:

- | | |
|--|-------------------------------------|
| ⟨1⟩ Ist $L(G_1) \cap L(G_2) = \emptyset$? | ⟨3⟩ Ist $L(G_1) \subseteq L(G_2)$? |
| ⟨2⟩ Ist $ L(G_1) \cap L(G_2) < \infty$? | ⟨4⟩ Ist $L(G_1) = L(G_2)$? |

Von ⟨1⟩ wurde in der Vorlesung gezeigt, dass es unentscheidbar ist. Zeigen Sie nun:

- | | |
|--------------------|--------------------|
| (a) ⟨3⟩ \leq ⟨4⟩ | (b) ⟨1⟩ \leq ⟨2⟩ |
|--------------------|--------------------|

Zur Vereinfachung dürfen Sie $\Sigma = \{a, b\}$ für das zu reduzierende Problem annehmen.

Übung und Nachbereitung

Übungsaufgabe Ü12.3. (Komplexe Fragen)

Entscheiden Sie jeweils, ob die folgenden Algorithmen korrekterweise das gegebene Problem in deterministisch, polynomieller Zeit entscheiden. Begründen Sie Ihre Antwort.

- (a) Gegeben: eine aussagenlogische Formel F . Frage: Ist F erfüllbar?

Algorithmus: Konstruiere die Wahrheitstabelle für F und werte F für jede Belegung aus der Tabelle aus.

- (b) Gegeben: Eine natürliche Zahl $n > 1$. Frage: Ist n prim?

Algorithmus: Überprüfe für alle Zahlen k zwischen 2 und $n - 1$ ob k n teilt. Wenn keine dieser Zahlen n teilt ist n prim.

Hinweis: Sie können annehmen das ein Teilbarkeitstest in polynomieller Zeit durchgeführt werden kann

- (c) Gegeben: Eine endlich Menge $S \subset \mathbb{N}$ mit n Elementen. Frage: Gilt für alle $(n-k)$ -elementigen Teilmengen K von S , dass $\sum_{s \in K} s > z$?

Algorithmus: Wiederhole, solange es noch neue $n - k$ -elementige Teilmengen von S gibt: Berechne die nächste solche Teilmenge K , summiere ihre Elemente und überprüfe ob diese Summe $> z$ ist. Wenn nein, antworte nein. Wenn es keine weitere $n - k$ -elementige Teilmengen von S gibt, antworte ja.

Hinweis: Sie dürfen annehmen, dass es einen Algorithmus gibt welcher die nächste Teilmenge in polynomieller Zeit berechnet.

- (d) Beantworten Sie Frage (c), wenn wir stattdessen n/k -elementige Teilmengen betrachten.

Hinweis: $\binom{n}{k} = \frac{n \cdot (n-1) \cdot \dots \cdot (n-k+1)}{k \cdot (k-1) \cdot \dots \cdot 1}$

Übungsaufgabe Ü12.4. (Guess What)

Entscheiden Sie jeweils, ob die folgenden Algorithmen korrekterweise das gegebene Problem in nichtdeterministisch, polynomieller Zeit semi-entscheiden. Falls ja, skizzieren Sie einen Beweis. Falls nein, beschreiben Sie den Fehler.

- (a) Gegeben: Rechtslineare Grammatik G für eine reguläre Sprache. Frage: ist $L(G)$ nicht leer?

Der Algorithmus ist eine NTM, die eine Ableitung in der Grammatik G rät. Genauer: Die Maschine beginnt beim Startsymbol. In jedem Schritt wählt die Maschine dann nichtdeterministisch ein Nichtterminal in der aktuellen Symbolkette und ersetzt es nichtdeterministisch durch eine passende Produktion. Falls die Symbolkette nur noch aus Terminalen besteht, wurde ein Wort in $L(G)$ gefunden und die Maschine antwortet positiv.

- (b) Gegeben: Grammatik G in Chomsky Normalform für eine kontextfreie Sprache. Frage: ist $L(G)$ nicht leer?

Der Algorithmus bleibt derselbe, wie in Aufgabe (a).

Übungsaufgabe Ü12.5. (SAT-Varianten)

Wir betrachten verschiedene Varianten von SAT, die auch NP-vollständig sind.

Zeigen Sie die NP-Vollständigkeit der folgenden Probleme A_1, A_2 , indem Sie jeweils $A_i \in \text{NP}$ zeigen und eine Reduktion $3\text{KNF-SAT} \leq_p A_i$ angeben.

(a) 3-OCC-KNF-SAT:

- **Eingabe:** Eine Formel F in KNF, bei der jede Variable höchstens dreimal auftritt.
- **Frage:** Ist F erfüllbar?

(b) Wir betrachten den ITE-Operator mit der Semantik $\text{ITE}(x, y, z) := (x \rightarrow y) \wedge (\neg x \rightarrow z)$.
Eine ITE-Formel genügt der folgenden Grammatik:

$$F \rightarrow \text{ITE}(F, F, F) \mid x \mid \text{true} \mid \text{false} \quad \text{für Variablen } x \in \mathcal{V}$$

ITE-SAT:

- **Eingabe:** Eine ITE-Formel F .
- **Frage:** Ist F erfüllbar?