Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2021 – Hausaufgabenblatt 1

Abgabe: 26.04.2021, 12:00 CEST

- Beachten Sie die Abgabemodalitäten auf der Vorlesungswebsite!
- Sei $\Phi := \{\{1\},\{2,3\},\{4,5\},\{6\}\}$. Nach dem Abgabedatum werden wir für jede Menge $A \in \Phi$ eine zufällige Aufgabe $a \in A$ wählen und korrigieren, d.h. Aufgaben H1.1 und H1.6 werden sicher korrigiert, während zwischen Aufgaben H1.2 und H1.3 eine Münze entscheidet.
- Es werden diese Aufgaben korrigiert: H1.1, H1.2, H1.4, H1.6
- Wenn Sie einen Beweis aufstellen, von dem Sie wissen, das einzelne Schritte problematisch oder unvollständig sind, merken Sie dies bitte in Ihrer Lösung an, damit wir das bei der Korrektur positiv berücksichtigen können.
- Es gilt $0 \in \mathbb{N}$.

Aufgabe H1.1. (*DFA/NFA Konstruktion*) 1+1+1 Punkte

Diese Hausaufgabe wird mit Automata Tutor bearbeitet und abgegeben. Falls Sie es noch nicht bereits gemacht haben, folgen Sie den Schritten in Ü1.2, um ein Konto zu erstellen. Achten Sie darauf, dass Sie sich, wie dort beschrieben, mit Ihrer TUM-Kennung anmelden. Ansonsten können wir Ihnen die Punkte nicht gutschreiben.

Bearbeiten Sie die Hausaufgaben H1.1 (a–c). **Achtung:** Während Sie für die Aufgaben aus dem Übungsblatt beliebig viele Versuche hatten, haben Sie für jede Hausaufgabe nur 5 Versuche. Sie bekommen nur dann einen Punkt, wenn Sie die Aufgabe nach 5 Versuchen vollständig (also mit 10/10 Punkten) gelöst haben.

Aufgabe H1.2. (*Sprachen*) 1+1+1+1 Punkte

Sei $\Sigma := \{a, b\}$, $A := \{a, ba, aab\}$, $B := \{\varepsilon, ab\}$. Bestimmen Sie folgende Mengen:

(a) $AB \setminus A$ (b) $A^3\emptyset$ (c) AB^0 (d) $\emptyset^* \times B^2$

Aufgabe H1.3. (*Lug und Trug*) 1+1+1+1 Punkte

Sei $\Sigma \neq \emptyset$ ein Alphabet, und $A, B, C \subseteq \Sigma^*$ Sprachen über Σ. Folgende Aussagen sind falsch; geben Sie jeweils ein Gegenbeispiel an.

(a) $A^+ \setminus A = A^2A^*$ (c) $|AB| = |BA|$

(b) $A \subseteq B \Leftrightarrow A^* \subseteq B^*$ (d) $AB \setminus AC = A(B \setminus C)$

Aufgabe H1.4. (*Teilbarkeit*) 2+3 Punkte

Der kleine Theodor lernt gerade in der Schule, wie mal Zahlen malnimmt und teilt. Damit er mehr Zeit zum Malen hat, will er seine Hausaufgaben vollautomatisch lösen.

Sei $\Sigma := \{\mathtt{0}, ..., \mathtt{9}\}$. Für ein Wort $w_1w_2...w_n = w \in \Sigma^*$ bezeichnen wir mit $(w)_{10} := \sum_{i=1}^{n} w_i \cdot 10^{n-i}$ die Zahl, die man erhält, wenn man w zur Basis 10 interpretiert (d.h. $(x)_{10} \in \mathbb{N}$ und es gilt z.B. $(\mathtt{42})_{10} = 42$ oder $(\varepsilon)_{10} = 0$).

Sei $k \in \mathbb{N}, k \geq 2$. Wir definieren die Sprache der durch k teilbaren Dezimalzahlen als $L_k := \{w \in \Sigma^* : (w)_{10} = i \cdot k,\ i \in \mathbb{N}\}$.

(a) Geben Sie einen deterministischen endlichen Automaten (DFA) an, der L_3 akzeptiert, und begründen Sie, dass ihr Automat korrekt ist, also genau die Wörter in L_3 akzeptiert.

(b) Geben Sie einen DFA an, der L_k akzeptiert, und beweisen Sie seine Korrektheit. *Anmerkung:* k ist ein Parameter, Sie müssen also die Zustände und die Übergangsfunktion in Abhängigkeit von k angeben.

Hinweis: Es genügt, Teil (b) sinnvoll zu bearbeiten. Die Punkte für (a) erhalten Sie dann automatisch.

Aufgabe H1.5. (*Buchstabenfhler*) 2+2+1 Punkte

Der Superschurke Dr. Evilsparza hat sich ins Campusnetzwerk gehackt und es ist ihm gelungen, aus mehreren Programmen Fragmente zu löschen! Sie sind Teil des Cybersecurityteams, das versucht das Ausmaß des Schadens zu bestimmen und die beschädigten Programme zu reparieren.

Sei $\Sigma := \{a, b\}$ ein Alphabet und $w \in \Sigma^*$ ein Wort. Mit w^R bezeichnen wir die Umdrehung von w, also $(w_1 w_2 ... w_n)^R := w_n w_{n-1} ... w_1$ (hier sind $w_1, w_2, ..., w_n \in \Sigma$) die Buchstaben von w. Beispielsweise gilt also $(abbaaa)^R = aaabba$.

(a) Sei $L := \{w \in \Sigma^* : w = w^R\}$ die Menge der Palindrome über Σ. Wir definieren folgende kontextfreie Grammatik $G = (\{S\}, \Sigma, P, S)$ mit Produktionen $P := \{S \to aSa \mid bSb \mid \varepsilon\}$.

Gilt $L(G) = L$? Wenn ja, beweisen Sie es, wenn nein, geben Sie ein Gegenbeispiel an und „reparieren" Sie G, konstruieren Sie also eine Grammatik G' mit $L(G') = L$, indem Sie neue Produktionen zu G hinzufügen.

(b) Sei G eine beliebige Grammatik, und G' eine Grammatik, die durch das Entfernen von einer beliebigen Produktion aus G entsteht. Beweisen Sie $L(G') \subseteq L(G)$, oder widerlegen Sie es durch Angabe eines Gegenbeispiels.

(c) Sei G wieder eine beliebige Grammatik, und G' eine Grammatik, die man erhält, wenn man von G ein beliebiges Nichtterminalzeichen aus der rechten Seite einer beliebigen Produktion löscht. Beweisen oder widerlegen Sie: $L(G') \subseteq L(G)$

Bonusaufgabe H1.6. (*Thue-Morse-Sequenz*) 1+4 Bonuspunkte

Die Thue-Morse-Sequenz über dem Alphabet $\Sigma := \{0, 1\}$ kann wie folgt definiert werden. Wir setzen $w_0 := 0$, und für $n \in \mathbb{N}$ definieren wir $w_{n+1} := w_n \overline{w_n}$, wobei $\overline{w}$ das Einerkomplement des Binärwortes $w \in \Sigma^*$ bezeichnet, z.B. $\overline{100111} = 011000$. Die ersten Folgenglieder sind also $w_0 = 0$, $w_1 = 01$, $w_2 = 0110$, und $w_3 = 01101001$.

(a) Überprüfen Sie $(w_5)_2 \bmod 7 = 4$, wobei $(w)_2$ – analog zu Aufgabe H1.4 – den Wert von w zur Basis 2 bezeichnet. Geben Sie Ihren vollständigen Rechenweg an.

(b) Bestimmen Sie $(w_{2021})_2 \bmod 19$. Beschreiben Sie ihr Vorgehen.

Hinweis: Für Teilaufgabe (b) dürfen sie einen Computer zu Hilfe nehmen. Falls Sie dies tun, beschreiben Sie bitte Ihren Ansatz in *natürlicher Sprache*. Sie können (müssen aber nicht), ihrer Lösung Programmcode beifügen, jedoch steht es den Korrektoren frei, diesen zu ignorieren. Zur Kontrolle: $(w_{42})_2 \bmod 19 = 2, (w_{43})_2 \bmod 19 = 10$.

(a) Geben Sie einen deterministischen endlichen Automaten (DFA) an, der L_3 akzeptiert, und begründen Sie, dass ihr Automat korrekt ist, also genau die Wörter in L_3 akzeptiert.

(b) Geben Sie einen DFA an, der L_k akzeptiert, und beweisen Sie seine Korrektheit. *Anmerkung:* k ist ein Parameter, Sie müssen also die Zustände und die Übergangsfunktion in Abhängigkeit von k angeben.

Hinweis: Es genügt, Teil (b) sinnvoll zu bearbeiten. Die Punkte für (a) erhalten Sie dann automatisch.

Aufgabe H1.5. (*Buchstabenfhler*) 2+2+1 Punkte

Der Superschurke Dr. Evilsparza hat sich ins Campusnetzwerk gehackt und es ist ihm gelungen, aus mehreren Programmen Fragmente zu löschen! Sie sind Teil des Cybersecurityteams, das versucht das Ausmaß des Schadens zu bestimmen und die beschädigten Programme zu reparieren.

Sei $\Sigma := \{a, b\}$ ein Alphabet und $w \in \Sigma^*$ ein Wort. Mit w^R bezeichnen wir die Umdrehung von w, also $(w_1 w_2 ... w_n)^R := w_n w_{n-1} ... w_1$ (hier sind $w_1, w_2, ..., w_n \in \Sigma$) die Buchstaben von w. Beispielsweise gilt also $(abbaaa)^R = aaabba$.

(a) Sei $L := \{w \in \Sigma^* : w = w^R\}$ die Menge der Palindrome über Σ. Wir definieren folgende kontextfreie Grammatik $G = (\{S\}, \Sigma, P, S)$ mit Produktionen $P := \{S \to aSa \mid bSb \mid \varepsilon\}$.

Gilt $L(G) = L$? Wenn ja, beweisen Sie es, wenn nein, geben Sie ein Gegenbeispiel an und „reparieren" Sie G, konstruieren Sie also eine Grammatik G' mit $L(G') = L$, indem Sie neue Produktionen zu G hinzufügen.

(b) Sei G eine beliebige Grammatik, und G' eine Grammatik, die durch das Entfernen von einer beliebigen Produktion aus G entsteht. Beweisen Sie $L(G') \subseteq L(G)$, oder widerlegen Sie es durch Angabe eines Gegenbeispiels.

(c) Sei G wieder eine beliebige Grammatik, und G' eine Grammatik, die man erhält, wenn man von G ein beliebiges Nichtterminalzeichen aus der rechten Seite einer beliebigen Produktion löscht. Beweisen oder widerlegen Sie: $L(G') \subseteq L(G)$

Bonusaufgabe H1.6. (*Thue-Morse-Sequenz*) 1+4 Bonuspunkte

Die Thue-Morse-Sequenz über dem Alphabet $\Sigma := \{0, 1\}$ kann wie folgt definiert werden. Wir setzen $w_0 := 0$, und für $n \in \mathbb{N}$ definieren wir $w_{n+1} := w_n \overline{w_n}$, wobei $\overline{w}$ das Einerkomplement des Binärwortes $w \in \Sigma^*$ bezeichnet, z.B. $\overline{100111} = 011000$. Die ersten Folgenglieder sind also $w_0 = 0$, $w_1 = 01$, $w_2 = 0110$, und $w_3 = 01101001$.

(a) Überprüfen Sie $(w_5)_2 \bmod 7 = 4$, wobei $(w)_2$ – analog zu Aufgabe H1.4 – den Wert von w zur Basis 2 bezeichnet. Geben Sie Ihren vollständigen Rechenweg an.

(b) Bestimmen Sie $(w_{2021})_2 \bmod 19$. Beschreiben Sie ihr Vorgehen.

Hinweis: Für Teilaufgabe (b) dürfen sie einen Computer zu Hilfe nehmen. Falls Sie dies tun, beschreiben Sie bitte Ihren Ansatz in *natürlicher Sprache*. Sie können (müssen aber nicht), ihrer Lösung Programmcode beifügen, jedoch steht es den Korrektoren frei, diesen zu ignorieren. Zur Kontrolle: $(w_{42})_2 \bmod 19 = 2, (w_{43})_2 \bmod 19 = 10$.