Prof. J. Křetínský
M. Weininger, T. Meggendorfer

Technische Universität München

KEINE ABGABE

Einführung in die theoretische Informatik

Sommersemester 2018 – Übungsblatt 5

Selbstständige Vorbereitung

Bereiten Sie sich auf die Tutorgruppen selbstständig vor, indem Sie die Aufgaben 5.1 bis 5.4 ansehen. Insbesondere Aufgabe 5.2 sollten Sie lösen können, da Aufgabe 5.6 darauf aufbaut.
Lösen Sie außerdem Aufgabe 4.4, sofern Sie es noch nicht getan haben, da Aufgabe 5.5 darauf aufbaut.

AUFGABE 5.1. (*Wichtige Begriffe*) Stufe A

Überprüfen Sie, dass Sie die Folgenden Begriffe korrekt definieren können.

- MyHill-Nerode-Relation
- Äquivalenzklasse
- Kanonischer Minimalautomat

AUFGABE 5.2. (*Myhill-Nerode-Relation und Äquivalenzklassen*) Stufe B

Sei $\mathsf{L} = \mathsf{L}(\mathsf{a}^*\mathsf{b}^*\mathsf{c}^*)$ über dem Alphabet $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$.

(a) Entscheiden Sie, welche der folgenden Äquivalenzen wahr sind und begründen Sie Ihre Antwort:

- $\varepsilon \overset{?}{\equiv}_{\mathsf{L}} \mathsf{a}$
- $\mathsf{b} \overset{?}{\equiv}_{\mathsf{L}} \mathsf{c}$
- $\mathsf{abc} \overset{?}{\equiv}_{\mathsf{L}} \mathsf{cba}$

(b) Sei $v = \mathsf{aababc}$. Geben Sie ein Wort $u \neq v$ an, so dass $u \equiv_{\mathsf{L}} v$.

(c) Geben Sie die Mengen $[\mathsf{ab}]_{\mathsf{L}}$, $[\mathsf{bc}]_{\mathsf{L}}$ und $[\mathsf{ca}]_{\mathsf{L}}$ an.

(d) Finden Sie nun L', so dass $\mathsf{c} \equiv_{\mathsf{L}'} \mathsf{ba}$, $\mathsf{c} \not\equiv_{\mathsf{L}'} \mathsf{ab}$ und $\mathsf{aba} \equiv_{\mathsf{L}'} \mathsf{bab}$. Weiterhin soll $\varepsilon, \mathsf{aba} \in \mathsf{L}'$ gelten.

AUFGABE 5.3. (*Anwendungsbeispiel kontextfreie Grammatiken*) Stufe B

Betrachten Sie die folgende Grammatik $\mathsf{G} = (\{J, D, T, N, N', Z, A, S, E, U, B, C, V, U', B'\}$, $\{;, \{, \}, (,), =, a, b, \ldots, y, z, 0, 1, \ldots, 8, 9, +, -, \cdot, /, \%, !, <, >, \&\&, ||\}, \mathsf{P}, J)$ mit den Produktionen $\mathsf{P} :=$

$J \to DS \mid S$
$D \to TN; D \mid TN;$
$T \to int$
$N \to AN'$
$N' \to AN' \mid ZN' \mid A \mid Z$
$A \to a \mid b \mid c \mid \cdots \mid y \mid z$
$Z \to 0 \mid 1 \mid \cdots \mid 8 \mid 9$
$S \to SS \mid ; \mid \{S\} \mid N = E; \mid N = read(); \mid write(E); \mid if(C)\ S\ else\ S \mid while(C)\ S$
$E \to Z \mid N \mid (E) \mid UE \mid EBE$
$U \to -$
$B \to - \mid + \mid \cdot \mid / \mid \%$
$C \to true \mid false \mid (C) \mid EVE \mid U'(C) \mid CB'C$
$V \to == \mid != \mid <= \mid < \mid >= \mid >$
$U' \to !$
$B' \to \&\& \mid ||$

(a) Was für eine Sprache erzeugt diese Grammatik?

(b) Beurteilen Sie die folgende Aussagen: Alle Worte in $\mathsf{L}(\mathsf{G})$ können zu einem funktionierenden Programm compiliert werden.

(c) Geben Sie ein gültiges Wort in der Sprache an, das alle Nichtterminale mindestens einmal verwendet.

(d) Zeichnen Sie den Syntaxbaum für das in Teilaufgabe (c) gefundene Wort.

AUFGABE 5.4. (*Syntaxbaum, Rechts-/Linksableitung bei kontextfreien Grammatiken*) Stufe B + D

Mit $\mathsf{G} = (\{\mathsf{S}, E, O, A, B, X\}, \{\mathsf{a}, \mathsf{b}\}, \mathsf{P}, \mathsf{S})$ sei die CFG mit folgenden Produktionen P bezeichnet:

$$\begin{array}{lcl} \mathsf{S} & \to & E \mid O \\ E & \to & AB \mid BA \\ A & \to & XAX \mid \mathsf{a} \\ B & \to & XBX \mid \mathsf{b} \\ O & \to & XXO \mid X \\ X & \to & \mathsf{a} \mid \mathsf{b} \end{array}$$

(a) Geben Sie für jedes der folgenden Wörter jeweils eine Linksableitung und eine Rechtsableitung zzgl. des entsprechenden Syntaxbaums an:

(i) abaaaa (ii) babab (iii) aabbaaba

Diskutieren Sie Gemeinsamkeiten und Unterschiede der Link- und Rechtssableitungen.

(b) Bestimmen Sie die von $\mathsf{L}(\mathsf{G})$ erzeugte Sprache und beweisen Sie, dass die Grammatik tatsächlich die angegebene Sprache erzeugt.

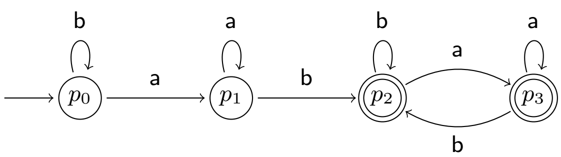
AUFGABE 5.5. (*Erweiterter Minimierungsalgorithmus*) Stufe C

In der Vorlesung haben Sie ein Verfahren zur Minimierung von DFAs gesehen. Wir erweitern diesen Algorithmus, damit er zusätzlich das unterscheidende Wort w für ein Paar inäquivalenter Zuständen berechnet.

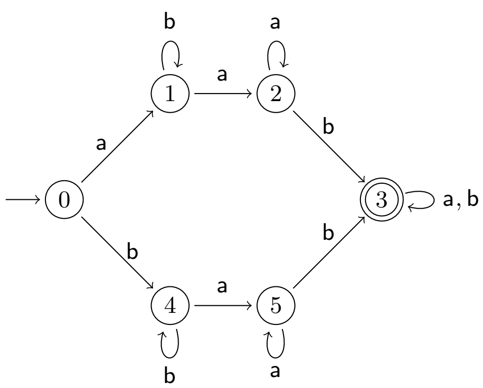
(a) Erweitern Sie den Minimierungsalgorithmus so, dass er für jedes Paar von inäquivalenten Zuständen $q, p \in \mathsf{Q}$ eines DFAs $\mathsf{D} = (\mathsf{Q}, \Sigma, \delta, \mathsf{q}_0, \mathsf{F})$ ein kürzestes Wort $w_{\{q,p\}}$ mit $\delta(q, w_{\{q,p\}}) \in \mathsf{F} \Leftrightarrow \delta(p, w_{\{q,p\}}) \notin \mathsf{F}$ berechnet.

(b) Verwenden Sie Ihren erweiterten Minimierungsalgorithmus, um die folgenden DFAs zu minimieren.

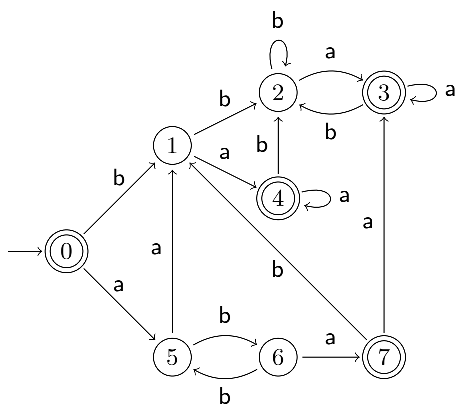
(i) DFA D_1:



(ii) DFA D_2:



(iii) DFA D_3:



AUFGABE 5.6. (*MyHill-Nerode-Relation*) Stufe D

Entscheiden Sie, ob folgende Sprachen über dem Alphabet $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$ regulär sind. Bestimmen Sie hierzu die Äquivalenzklassen der dazugehörigen Myhill-Nerode-Relation. Falls die Sprache regulär ist, zählen Sie alle Äquivalenzklassen auf und zeichnen Sie den kanonischen Minimalautomat $M_{\mathsf{L}} = (\Sigma^*/\equiv_{\mathsf{L}}, \Sigma, \delta_{\mathsf{L}}, [\varepsilon]_{\equiv_{\mathsf{L}}}, F_{\mathsf{L}})$. Falls die Sprache nicht regulär ist, reicht es eine unendliche Menge von Äquivalenzklassen zu bestimmen.

(a) $\mathsf{L}_1 = \{\mathsf{a}^{2i} \mid i \in \mathbb{N}_0\}$
(b) $\mathsf{L}_2 = \{\mathsf{a}^i\mathsf{b}^i\mathsf{c}^i \mid i \in \mathbb{N}_0\}$
(c) $\mathsf{L}_3 = \{w \in \Sigma^* \mid |w|_{\mathsf{a}} = 2 \cdot |w|_{\mathsf{b}}\}$
(d) $\mathsf{L}_4 = \mathsf{L}((\mathsf{a}^*(\mathsf{b} \mid \mathsf{c}))^*)$
(e) $\mathsf{L}_5 = \{w\mathsf{c}w \mid w \in \{\mathsf{a}, \mathsf{b}\}^*\}$

AUFGABE 5.7. (*Lernen von regulären Sprachen*) Stufe C

Bilden Sie Gruppen zu jeweils vier Studierenden. Zwei Studierende nehmen die Rolle des **Teachers** ein und zwei die Rolle des **Learners** ein. Die Aufgabe des **Teachers** ist es, sich eine reguläre Sprache L auszudenken und Fragen zu dieser zu beantworten. Die **Learner** müssen diese Sprache lernen, d.h. einen Automaten für diese Sprache konstruieren. Hierbei dürfen die Learner aber nur die folgenden zwei Fragen stellen:

(a) **Membership:** Ist das Wort w in der Sprache L?
(b) **Equivalence:** Erkennt der gegebene Automat (DFA, NFA, ε-NFA) A die Sprache L?

Die Teacher dürfen nur mit **Ja** oder **Nein** antworten. Zusätzlich geben sie bei (2) bei der Antwort **Nein** ein unterscheidendes Wort w an. Ein unterscheidendes Wort hat die folgenden Eigenschaften: $w \in \mathsf{L} \wedge w \notin \mathsf{L}(A)$ *oder* $w \notin \mathsf{L} \wedge w \in \mathsf{L}(A)$.

(a) Diskutieren Sie zunächst, wie man ein unterscheidendes Wort für zwei Automaten finden kann bzw. wie man entscheiden kann, dass zwei Automaten gleich sind.
(b) Wenden Sie dann das Teacher-Learner-Verfahren so an, dass jedes Gruppenmitglied jede Rolle zweimal inne hatte.

Definition (Monoid)

Ein Monoid $\langle M, \circ, 1\rangle$ besteht aus einer (Träger-)Menge M, einer assoziativen Abbildung $\circ\colon M \times M \to M$ und einem bzgl. $\circ$ neutralen Element $1 \in M$ (d.h. $\forall m \in M.\ m \circ 1 = m = 1 \circ m$). Ist $\circ$ kommutativ, dann wird $\langle M, \circ, 1\rangle$ als kommutatives Monoid bezeichnet. Gilt $\forall m \in M.\ m \circ m = m$, so wird $\langle M, \circ, 1\rangle$ als **idempotentes** Monoid bezeichnet.

Definition (Kleene Algebra)

Eine *Kleene Algebra* $\langle K, +, \cdot, ^*, 0, 1\rangle$ besteht aus einer Trägermenge K, den binären Operationen $+\colon K \times K \to K$ (Addition), $\cdot\colon K \times K \to K$ (Multiplikation), der unären Operation $^*\colon K \to K$ (Stern) und zwei Konstanten $0, 1 \in K$. Mittels der Addition definiert man die binäre Relation $\sqsubseteq$ auf K durch

$$a \sqsubseteq b \overset{\text{Def}}{\longleftrightarrow} a + b = b$$

Wie üblich schreibt man kurz ab für $a \cdot b$. Um Klammern zu sparen, gilt "Stern vor Punkt vor Strich".
Eine Kleene Algebra erfüllt folgende Eigenschaften für alle $a, b, c, x \in K$:

Ax1: $\langle K, +, 0\rangle$ ist ein kommutatives und idempotentes Monoid.

Ax2: $\langle K, \cdot, 1\rangle$ ist ein Monoid.

Ax3: $a(b + c) = ab + ac$ und $(a + b)c = ac + bc$.

Ax4: $a0 = 0 = 0a$.

Ax5: $1 + aa^* \sqsubseteq a^*$ und $1 + a^*a \sqsubseteq a^*$.

Ax6: $b + ax \sqsubseteq x \to a^*b \sqsubseteq x$ und $b + xa \sqsubseteq x \to ba^* \sqsubseteq x$.

AUFGABE 5.8. (*Abstraktion von Problemen*) Stufe E

In dieser Aufgabe abstrahieren wir von der konkreten Interpretation von regulären Ausdrücken als Konstruktionsbeschreibungen regulärer Sprachen. Ziel ist es den Zusammenhang mit Pfadproblemen im Bereich der Informatik und dem Lösen linearer Gleichungssysteme zu verdeutlichen.

(a) Sei Σ ein Alphabet. Beweisen Sie, dass $\langle 2^{\Sigma^*}, \cup, \circ, ^*, \emptyset, \{\varepsilon\}\rangle$ eine Kleene Algebra ist für

$$LL' := L \circ L' := \{ww' \mid w \in L, w' \in L'\} \qquad \text{und} \qquad L^* := \bigcup_{k \in \mathbb{N}_0} L^k$$

(b) Die Addition und das Minimum auf $\mathbb{R}$ seien auf $[-\infty, \infty] = \mathbb{R} \cup \{\pm\infty\}$ wie folgt erweitert:

$$\infty + a = \infty \quad -\infty + a = -\infty \quad -\infty + \infty = \infty$$

$$\min(a, \infty) = a \quad \min(a, -\infty) = -\infty \quad \min(-\infty, \infty) = -\infty$$

Weiterhin gelte:

$$a^* := \begin{cases} -\infty & \text{falls } a \in [-\infty, 0) \\ 0 & \text{falls } a \in [0, \infty] \end{cases}$$

Beweisen Sie, dass $\langle \mathbb{R} \cup \{\pm\infty\}, \min, +, ^*, \infty, 0\rangle$ eine Kleene Algebra ist.

(c) Zeigen Sie, dass in jeder Kleene Algebra $\langle K, +, \cdot, ^*, 0, 1\rangle$ für beliebige $a, b, c, d, e, f \in K$ gilt:

(i) $\sqsubseteq$ ist eine partielle Ordnung auf K, die monoton bzgl. Addition, Multiplikation und Stern ist, d.h.:

$$a \sqsubseteq b \to (ca \sqsubseteq cb \wedge ac \sqsubseteq bc \wedge a + c \sqsubseteq b + c \wedge a^* \sqsubseteq b^*)$$

(ii) a^*b ist die bzgl. $\sqsubseteq$ kleinste Lösung der linearen Ungleichung $b + aX \sqsubseteq X$ in K (X Variable), genauer:

$$b + a(a^*b) \sqsubseteq a^*b \qquad \wedge \qquad \forall x \in K : b + ax \sqsubseteq x \to a^*b \sqsubseteq x$$

Entsprechend ist ba^* die kleinste Lösung in K von $b + Xa \sqsubseteq X$.

Man kann zeigen, dass jedes lineare Ungleichungssystem

$$\begin{array}{rcl} a_{1,1}X_1 + a_{1,2}X_2 + \ldots a_{1,n}X_n + b_1 & \sqsubseteq & X_1 \\ \vdots & & \\ a_{n,1}X_1 + a_{n,2}X_2 + \ldots a_{n,n}X_n + b_n & \sqsubseteq & X_n \end{array}$$

mit $a_{i,j}, b_i \in K$ und $X_1, \ldots, X_n$ Variablen stets eine eindeutige $\sqsubseteq$-kleinste Lösung in K hat, d.h. dass es konkrete Elemente $x_1, \ldots, x_n \in K$ gibt, so dass für $X_1 = x_1, \ldots, X_n = x_n$ alle obigen Ungleichungen erfüllt sind, und für jede weitere Lösung $X_1 = y_1, \ldots, X_n = y_n \in K$ stets $x_i \sqsubseteq y_i$ gilt. Insbesondere gilt

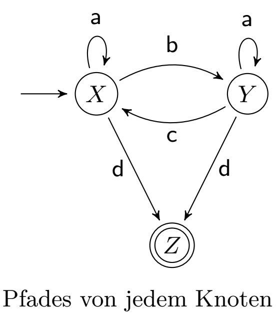
$$x_1 = a_{1,1}^*(a_{1,2}x_2 + \ldots + a_{1,n}x_n + b_1)$$

Somit kann das Gauß-Verfahren zur Bestimmung von $x_1, \ldots, x_n$ verwendet werden.

(d) Bestimmen Sie die kleinste Lösung x_1, x_2, x_3, x_4 für folgendes System:

$$\begin{array}{rcl} aX + bY + eZ & \sqsubseteq & X \\ cX + dY + fZ & \sqsubseteq & Y \\ 1 & \sqsubseteq & Z \end{array}$$

(e) Bestimmen Sie den regulären Ausdruck für den Zustand X. Durch welche Werte muss man a, b, c, d, e, f konkret ersetzen, damit sich die in (d) berechneten Terme in $\langle 2^{\Sigma^*}, \cup, \circ, ^*, \emptyset, \{\varepsilon\}\rangle$ zu dem gewünschten Ausdruck auswertet?



(f) Bestimmen Sie die Länge eines kürzesten Pfades von jedem Knoten zum Knoten Z in folgendem gewichteten Graphen. Durch welche Werte muss man a, b, c, d, e, f konkret ersetzen, damit sich die in (d) berechneten Terme in $\langle \mathbb{R} \cup \{\pm\infty\}, \min, +, \infty, 0\rangle$ zu den gesuchten Pfadlängen auswerten?