Einführung in die theoretische Informatik – Aufgabenblatt 4

Beachten Sie: Soweit nicht explizit angegeben, sind Ergebnisse stets zu begründen!

Hausaufgaben: Abgabe bis zum 11.05.2016 (Mittwoch) um 12:00

Aufgabe 4.1 Umkehrung (Spiegelung) 1P+1P+3P

In den Folien finden Sie einen Algorithmus mit dem man aus einem DFA M einen ε-NFA N konstruieren kann, so dass $L(M)^R = L(N)$ gilt.

(a) Konstruieren Sie für den regulären Ausdruck $r = 0 \mid 1(0|1)^*0$ einen DFA und wenden Sie das Spiegelungsverfahren an.

(b) Geben Sie einen regulären Ausdruck r' mit $L(r) = L(r')^R$ an.

(c) Geben Sie nun, entsprechend TA 3.3(c) ein Verfahren an, dass rekursiv einen regulären Ausdruck r in einen regulären Ausdruck r' umschreibt, so dass $L(r)^R = L(r')$ gilt. Zeigen Sie die Korrektheit Ihres Verfahrens mittels struktureller Induktion und seien Sie besonders ausführlich in den Fällen $\alpha\beta$, $\alpha \mid \beta$ und α^*.

Aufgabe 4.2 Pumping-Lemma 2P+2P

Zeigen Sie mithilfe des Pumping-Lemmas, dass folgende Sprachen nicht regulär sind:

(a) $L_a = \{a^{6i}b^{6i} \mid i \geq 0\}$

(b) $L_b = \{a^{2^i} \mid i \geq 0\}$

Aufgabe 4.3 Homomorphismen auf regulären Sprachen 3P+1P

Sei Σ ein Alphabet und $L \subseteq \Sigma^*$ eine reguläre Sprache. Weiter sei $h\colon \Sigma \to \Sigma^*$ eine Abbildung, die jedem Zeichen $a \in \Sigma$ ein Wort $h(a) \in \Sigma^*$ zuordnet. h erweitert man kanonisch auf Σ^* mittels $h(\varepsilon) = \varepsilon$ und $h(xy) = h(x)h(y)$ für alle $x, y \in \Sigma^*$. Schließlich sei $h(L) = \{h(w) \mid w \in L\}$.

(a) Zeigen Sie: Ist L regulär, dann ist auch $h(L)$ regulär.

(b) Verwenden Sie das Resultat aus (a), um zu zeigen, dass $L' = \{ab^{3i}cd^{2i}e \mid i \in \mathbb{N}_0\}$ nicht regulär ist. Hierbei dürfen Sie das Ergebnis aus HA 4.2 verwenden.

Aufgabe 4.4 Produktkonstruktion 2P+1P

Wir definieren den *ITE*-Operator (*if-then-else*) für Mengen:

$$ITE(A, B, C) = \{w \in \Sigma^* \mid (w \in A \land w \in B) \lor (w \notin A \land w \in C)\}$$

Seien jetzt $A, B, C \subseteq \Sigma^*$ reguläre Sprachen, die von den DFAs M_A, M_B und M_C erkannt werden.

(a) Erweitern Sie die Produktkonstruktion aus der Vorlesung, so dass aus den Automaten M_A, M_B und M_C ein DFA $M_{ITE(A,B,C)}$ mit $L(M_{ITE(A,B,C)}) = ITE(A, B, C)$ konstruiert wird. Beweisen Sie auch die Korrektheit dieser Konstruktion!

(b) Sei nun $\Sigma = \{a, b, c\}$. Wenden Sie diese Konstruktion auf die Sprachen $A = \{w \in \Sigma^* \mid |w| \text{ ist gerade}\}$, $B = \{a^nb^m \mid n, m \geq 0\}$ und $C = \{c^nb^m \mid n, m \geq 0\}$.

Tutoraufgaben: Besprechung in KW19

Bitte beachten Sie die Regelung zu den Pfingstferien auf der Webseite.

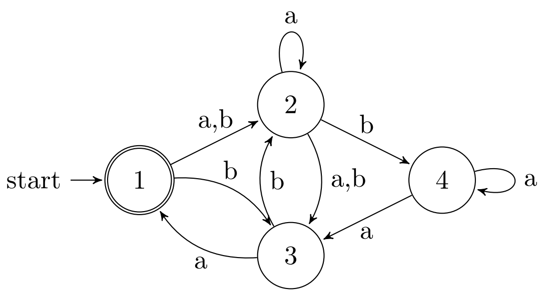
Aufgabe 4.1

Geben Sie einen möglichst effizienten Algorithmus an, der für einen gegebenen NFA $N = (Q, \Sigma, \delta, q_0, F)$ den Wert $\min\{|w| \mid w \in L(N)\}$ berechnet.

Aufgabe 4.2

(a) Passen Sie den Minimierungsalgorithmus so an, dass er für jedes Paar von inäquivalenten Zuständen q, q' eines DFAs ein Wort w mit $\delta(q, w) \in F \Leftrightarrow \delta(q', w) \notin F$ berechnet.

(b) Determinisieren und dann minimieren Sie folgenden NFA. Verwenden Sie für die Minimierung den Algorithmus aus (a).



Aufgabe 4.3

Bestimmen Sie die MyHill-Nerode-Relation für jede der folgenden Sprachen über dem Alphabet $\Sigma = \{a, b, c\}$ und skizzieren Sie jeweils den Quotientenautomaten $(\Sigma^*/\equiv_L, \Sigma, \delta_L, [\varepsilon]_{\equiv_L}, F_L)$.

(a) $L_1 = L((ab \mid ba)^*)$

(b) $L_2 = L((aa)^*)$

(c) $L_3 = \{a^i b^i c^i \mid i \in \mathbb{N}_0\}$

Knobelaufgabe

Freiwillige Abgabe per E-Mail an theo-uebungsleitung@in.tum.de. Der Preis diese Woche für die ersten drei richtigen Gesamtlösungen ist jeweils ein Chari Tea Red. Alle richtigen Einsendungen bis zum 11.05.2016 um 12:00 Uhr erhalten die Bonuspunkte. Wir bewerten nur die erste Einsendung und behalten uns das Recht vor unleserliche oder unverständliche Abgaben nicht zu korrigieren.

Aufgabe 4.1 **Bonus: 2P**

Eine Arena $(V, \Sigma, E, \sigma, v_0, Z)$ besteht aus einer endlichen Menge V von Knoten, einem endlichen Alphabet Σ, Σ-beschrifteten Kanten $E \subseteq V \times \Sigma \times V$, einer Abbildung $\sigma \colon V \to \{\top, \bot\}$, die jedem Knoten einen Spieler zuordnet, einem Startknoten v_0 und einer Menge Z von Zielknoten.

Gegeben ein Wort $w = a_0 a_1 \dots a_l \in \Sigma^*$ ($a_i \in \Sigma$) sieht ein *Spielverlauf* wie folgt aus: beginnend in v_0 wählt stets Spieler $\sigma(v_i)$ den nächsten Knoten v_{i+1} aus allen über eine mit a_i beschriftete Kante erreichbaren Knoten, d.h. es muss $(v_i, a_i, v_{i+1}) \in E$ gelten. Spieler $\top$ gewinnt den Spielverlauf $v_0 v_1 \dots v_l v_{l+1}$, falls am Ende $v_{l+1} \in Z$ gilt, ansonsten gewinnt Spieler $\bot$.

Spieler $\top$ gewinnt das Wort w, falls er seine Züge so wählen kann, dass er jeden dann noch möglichen Spielverlauf gewinnt.

Zeigen Sie: Die Sprache der Wörter, die $\top$ gewinnt, ist regulär.