Technische Universität München
Fakultät für Informatik
Prof. Tobias Nipkow, Ph.D.
Sascha Böhme, Lars Noschinski

Sommersemester 2011
Übungsblatt 2
16. Mai 2011

Einführung in die Theoretische Informatik

Abgabetermin: 23. Mai 2011 **vor der Vorlesung** *in die THEO-Briefkästen*

Hinweis: *Bitte beachten Sie unbedingt die Hinweise zum Übungsablauf und zu den Aufgabentypen auf der THEO-Website (`http://theo.in.tum.de/`).*

Hausaufgabe 1 (5 Punkte)

1. Beweisen oder widerlegen Sie für Sprachen $A, B \subseteq \Sigma^*$:

$$(AB)^* = (AB \cup B)^* \qquad\qquad (A^*B^*)^* = (A \cup B)^*$$

   Gilt eine Gleichung nicht, so geben Sie möglichst einfache Eigenschaften für A und B an, so dass alle Fälle erfasst sind, in denen diese Gleichung gilt.

2. Beweisen oder widerlegen Sie für reguläre Ausdrücke α und β:

$$(\alpha\beta|\alpha)^*\alpha \equiv \alpha(\beta\alpha|\alpha)^* \qquad\qquad (\alpha\beta|\alpha)^* \equiv (\alpha\beta)^*|\alpha^*$$

Hausaufgabe 2 (7 Punkte)

Wir betrachten das Alphabet $\Sigma = \{0, 1\}$.

1. Finden Sie einen regulären Ausdruck α, der folgende Sprache beschreibt: Jedes Wort w in $L(\alpha)$ endet in einer Sequenz ungerader Länge von Einsen und alle anderen Sequenzen von Einsen in w sind von gerader Länge. Wandeln Sie diesen regulären Ausdruck mit dem Standardverfahren aus der Vorlesung in einen ϵ-NFA M um, so dass $L(M) = L(\alpha)$, und geben sie dabei sowohl den Graphen als auch die Übergangsrelation δ an.

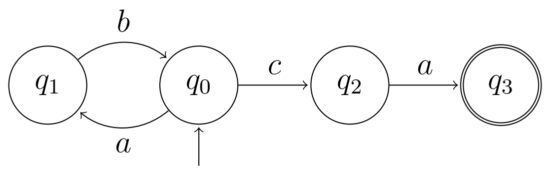
2. Geben Sie einen NFA M_1 an, der die Menge aller Wörter beschreibt, die entweder aus einer ungeraden Anzahl von Nullen bestehen oder aber mit einer Null anfangen, die von beliebig vielen Teilwörtern 01 gefolgt wird. Nutzen Sie den Nichtdeterminismus aus. Wandeln Sie diesen NFA in einen DFA M_2 um, so dass $L(M_2) = L(M_1)$. Geben Sie für beide Automaten sowohl den Graphen als auch die Übergangsrelation an.

Hausaufgabe 3 (4 Punkte)

Ein *Fragment* von $w \in \Sigma$ ist ein Wort, dass aus w durch Entfernen beliebiger Buchstaben entstanden ist. Zum Beispiel sind ϵ, a, b, aa, ab, ba, bb, aaa, aab, aba, baa. bab, bba, $baaa$, $baab$, $baba$ und $baaba$ alle Fragmente von $baaba$.

Für eine Sprache L bezeichen wir mit $\text{Fragment}(L)$ die Sprache aller Fragmente aller Wörter in L.

1. Sei L die von dem folgenden Automaten akzeptierte Sprache.



   Konstruieren Sie daraus einen endlichen Automaten, der $\text{Fragment}(L)$ akzeptiert.

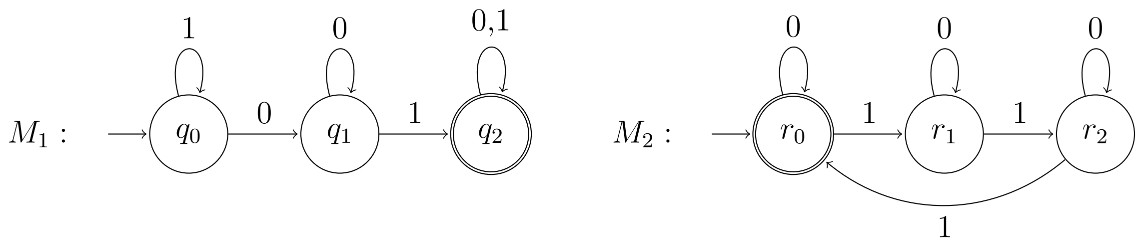
2. Zeigen Sie, dass für alle regulären Sprachen L auch $\text{Fragment}(L)$ regulär ist. Formulieren Sie dazu einen Algorithmus, der zu einem NFA M einen NFA M' mit $L(M') = \text{Fragment}(L(M))$ konstruiert, und begründen Sie seine Korrektheit.

3. Sei $\Sigma = \{a, b\}$. Die Sprache $L = \{a^n b^n \mid n \in \mathbb{N}\}$ ist nicht regulär. Ist $\text{Fragment}(L)$ dennoch regulär? Begründen Sie.

Hausaufgabe 4 (4 Punkte)

1. Gegeben seien zwei DFAs M_1 und M_2 über dem gleichen Eingabealphabet. Geben Sie ein direktes Verfahren an (ohne Umweg über reguläre Ausdrücke oder NFAs), um einen DFA M mit $L(M) = L(M_1) \cup L(M_2)$ zu konstruieren.

2. Wenden Sie dieses Verfahren auf die beiden folgenden Automaten an.



Quiz 1

Beantworten Sie kurz die folgenden Fragen:

1. Geben sie eine Sprache A über dem Alphabet $\Sigma = \{0, 1\}$ an, so dass $01 \in A$ und $A^*A = A$.
2. Wann genau ist die von einem endlichen Automaten erzeugte Sprache endlich?
3. Finden Sie einen regulären Ausdruck über dem Alphabet $\Sigma = \{0, 1\}$, der die Menge aller Wörter beschreibt, die mit 00 beginnen und in denen 1 genau dreimal vorkommt.

Tutoraufgabe 1

Wir betrachten den regulären Ausdruck $\alpha = (1(0|1)^*)|0$.

1. Konstruieren Sie für α mit dem Standardverfahren aus der Vorlesung einen ϵ-NFA A, so dass $L(\alpha) = L(A)$ gilt.
2. Wandeln Sie den erhaltenen Automaten in einen äquivalenten NFA ohne ϵ-Übergänge um.
3. Konstruieren Sie durch Anwendung des Potenzmengenverfahrens einen DFA, der die Sprache des Ausdrucks α akzeptiert.

Tutoraufgabe 2

Das sogenannte *Shuffle-Produkt* spielt in der Theorie der nebenläufigen Systeme eine wichtige Rolle. Für zwei Sprachen L_1 und L_2 bezeichnet $L_1 \parallel L_2$ die Menge der Wörter, die man erhält, inde man zwei Wörter $v \in L_1$ und $u \in L_2$ beliebig miteinander verschränkt. Dabei können sich Teile aus v und w beliebig abwechseln, wobei die Reihenfolge der Zeichen aus v und w jedoch erhalten bleibt. Das kann man sich gut als das Ineinanderschieben zweier Kartenstapel veranschaulichen.
Formal definieren wir $L_1 \parallel L_2$ wie folgt:

$$L_1 \parallel L_2 = \{v_1 w_1 v 2_w 2 \cdots v_n w_n \mid v_1, \ldots, v_n, w_1, \ldots, w_n \in \Sigma^*, \\ v_1 v_2 \cdots v_n \in L_1 \text{ und } w_1 w_2 \cdots w_n \in L_2\}$$

1. Versuchen Sie, eine einfache Beschreibung von $L((01)^*) \parallel L((10)^*)$ zu finden.
2. Begründen Sie: Wenn L_1 und L_2 regulär sind, dann ist auch $L_1 \| L_2$ regulär. *Hinweis: Konstruieren Sie einen NFA für $L_1 \parallel L_2$.*
3. Führen Sie die Konstruktion konkret für die Sprachen $L((01)^*)$ und $L((10)^*)$ durch und geben Sie einen regulären Ausdruck an, der die Sprache beschreibt.

Tutoraufgabe 3

Beweisen oder widerlegen Sie für reguläre Ausdrücke α und β:

1. $(\alpha\beta)^*\alpha \equiv \alpha(\beta\alpha)^*$
2. $\alpha|\alpha^* \equiv \alpha^*$

Falls möglich, verwenden Sie für einen Beweis die Regeln zum Rechnen mit regulären Ausdrücken aus der Vorlesung.