Prof. Dr. R. Schrader
Birgit Engels [engels@zpr.uni-koeln.de]

WS 2006/2007

Theoretische Informatik

11. Übung, Abgabe Mittwoch, 17.01.2007

Aktuelle Informationen bezüglich der Vorlesung und der Übungen finden sich unter :
http://www.zaik.uni-koeln.de/AFS/teachings/courses/ThInf/index.html und
http://www.zaik.uni-koeln.de/AFS/teachings/courses/ThInf/uebungen.html

Aufgabe 39 (Deterministische endliche Automaten — DEAs — 4 Punkte):
Entwerfen Sie einen deterministischen endlichen Automaten[1] $A_{cola} = (Q, q_0, F, \Sigma, \delta)$ mit folgender Funktionalität:

- Der Cola-Automat verkauft Cola und Fanta zu je 2 Euro und Wasser zu 1 Euro.
- Der Automat akzeptiert 50-Cent-Stücke sowie 1- und 2-Euro-Stücke.
- Ein Einwurf der den gesamten bisher eingeworfenen Geldbetrag über 2 Euro ansteigen läßt wird nicht akzeptiert (Münze fällt durch, Zustand des Automaten ändert sich nicht.)
- Der Automat hat 4 Tasten : **Cola**, **Fanta**, **Wasser**, **Abbruch**.
- Wird eine der Getränketasten gedrückt, so wird—falls genügend Geld eingeworfen wurde—das entsprechende Getränk und Restgeld ausgegeben.
- Wird die Taste **Abbruch** betätigt, so gibt der Automat sofort das bisher eingeworfene Geld aus und kehrt in den Ausgangszustand zurück.

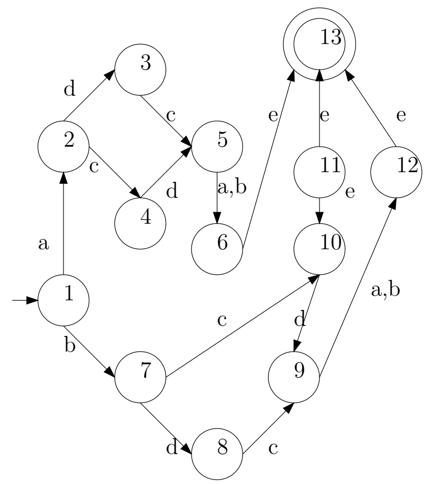
Hinweis: Geben Sie die Menge der Zustände Q, den Startzustand q_0, die Menge der (akzeptierenden) Endzustände F und das (relevante) Alphabet Σ separat an und stellen Sie δ als gerichteten Graphen mit Knoten aus Q und Kantenbeschriftungen aus Σ dar (Zustandsübergangsdiagramm).

Aufgabe 40 (DEA-Minimierung — Unerreichbare Zustände — 3 Punkte):
Geben Sie einen Algorithmus (in Pseudo-Code) an, der alle nicht erreichbaren Zustände eines Automaten entfernt. Der Automat M kann in seiner Darstellung als Zustandübergangsdiagramm als Graph $G(M) = (V, E)$ betrachtet werden.

Bitte wenden!

[1] Die englische Bezeichnung „Finite State Machine“ führt zu der gebräuchlichsten Abkürzung FSM.

Aufgabe 41 (DEA-Minimierung — Äquivalente Zustände — 6 Punkte):
Wenden Sie den Algorithmus zur Markierung äquivalenter Zustände (aus der Vorlesung) auf den untenstehenden Automaten M an und erzeugen Sie so den minimierten Automaten M' mit $L(M) = L(M')$! (Beachten Sie auch Aufgabe 40.)



Aufgabe 42 (Reguläre Mengen, reguläre Ausdrücke — 7 Punkte):
In der Vorlesung habe Sie reguläre Sprachen kennengelernt. Diese entsprechen im Prinzip einer regulären Menge von Worten über einem Alphabet Σ. Reguläre Mengen können wie folgt induktiv definiert und durch reguläre Ausdrücke beschrieben werden:

Reguläre Menge	regulärer Ausdruck
$\emptyset$ ist eine reguläre Menge über Σ.	$\emptyset$
$\{\epsilon\}$ ist eine reguläre Menge über Σ.	ϵ
$\forall a \in \Sigma : \{a\}$ reguläre Menge über Σ.	$a \in \Sigma$
Für 2 reguläre Mengen P und Q über Σ gilt:	
$P \cup Q$ ist eine reguläre Menge über Σ.	$(p\|q)$
$PQ = \{pq\|p \in P, q \in Q\}$ ist eine reguläre Menge über Σ.	(pq)
$P^* = \cup_{n \geq 0} P^n$ ist eine reguläre Menge über Σ.	(p^*)
Es gibt keine weiteren regulären Mengen über Σ.	

Beispielsweise lässt sich die Menge der Fliesskommazahlen über dem Alphabet Σ mittels des regulären Ausdrucks

$$(0|(1|\ldots|9)(0|1|\ldots|9)^*),(0|1|\ldots|9)^*, \Sigma_f = \{0,\ldots,9\} \cup \{,\}$$

darstellen. (Zur eindeutigen Lesbarkeit werden Klammern verwendet.) Wiederkehrende Teilausdrücke können dabei auch durch eigene Benennungen ($\notin \Sigma$!) und geeignete Abkürzungen (...) verkürzt werden:

$$(0|(1|\ldots|9)zif^*), zif^* \ mit \ zif = (0|1|\ldots|9).$$

Weiterhin liesse sich z.B. die Anzahl n der Nachkommastellen begrenzen mittels

$$(0|(1|\ldots|9)(0|1|\ldots|9)^*),(0|1|\ldots|9)^n.$$

Zeigen Sie, dass die Sprache L_{Kfz} der zulässigen Kfz-Kennzeichen von Köln regulär ist, indem Sie:

a) L_{Kfz} durch die Beschreibung mit regulären Ausdrücken als reguläre Menge ausweisen.

b) Einen (minimalen) DEA konstruieren, der L_{Kfz} akzeptiert!