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

Sommersemester 2011
Lösungsblatt Mittelklausur
1. Juli 2011

Einführung in die Theoretische Informatik

Name	Vorname	Studiengang	Matrikelnummer
....	....	☐ Diplom ☐ Inform. ☐ Bachelor ☐ BioInf. ☐ Lehramt ☐ Mathe.	....

Hörsaal	Reihe	Sitzplatz	Unterschrift
....	....	....	....

Code:

Allgemeine Hinweise

- Bitte füllen Sie obige Felder in Druckbuchstaben aus und unterschreiben Sie!
- Bitte schreiben Sie nicht mit Bleistift oder in roter/grüner Farbe!
- Die Arbeitszeit beträgt 120 Minuten.
- Alle Antworten sind in die geheftete Angabe auf den jeweiligen Seiten (bzw. Rückseiten) der betreffenden Aufgaben einzutragen. Auf dem Schmierblattbogen können Sie Nebenrechnungen machen. Der Schmierblattbogen muss ebenfalls abgegeben werden, wird aber in der Regel nicht bewertet.
- Es sind keine Hilfsmittel außer einem DIN-A4-Blatt zugelassen.

Hörsaal verlassen von ...... bis ...... / von ...... bis ......
Vorzeitig abgegeben um ......
Besondere Bemerkungen:

	A1	A2	A3	A4	A5	Σ	Korrektor
Erstkorrektur							
Zweitkorrektur							

Aufgabe 1 (8 Punkte)

Wahr oder falsch? Begründen Sie im Folgenden Ihre Antworten möglichst knapp!

1. Jede Teilmenge einer kontextfreien Sprache ist wieder kontextfrei.
2. Sei L_1 regulär und L_2 kontextfrei. Dann ist $L_1 \cap L_2$ regulär.
3. Es gibt eine nicht-reguläre Sprache L, so dass L^* regulär ist.
4. Sei $\mathcal{L}$ eine abzählbare Menge regulärer Sprachen und sei L die Vereinigung der Sprachen in $\mathcal{L}$. Dann ist L regulär.
5. Seien A und B Sprachen mit $A = AB$. Dann gilt $A = B^*$.
6. Ein endlicher Automat, bei dem die Kanten mit Wörtern anstatt mit einzelnen Buchstaben beschriftet sind, beschreibt eine reguläre Sprache.
7. $A \equiv a^*$ ist eine Lösung der Gleichung $A \equiv aA$.
8. Sei $G = (\{S, T\}, \{a, b, c\}, P, S)$ eine Grammatik, wobei P aus den Produktionen $S \to aSb \mid Sb \mid T$ und $T \to aT \mid \epsilon$ besteht. Dann ist $L(G)$ regulär.

Lösungsvorschlag

1. Falsch. Für $\Sigma = \{a, b\}$ ist $\{ww \mid w \in \Sigma^*\} \subseteq \Sigma^*$ nicht kontextfrei. Σ^* ist jedoch kontextfrei.
2. Falsch. $L_1 = \{a^m b^n \mid m, n \in \mathbb{N}\}$ ist regulär und $L_2 = \{a^n b^n \mid n \in \mathbb{N}\}$ ist kontextfrei (aber nicht regulär). Damit ist auch $L_1 \cap L_2 = L_2$ nicht regulär.
3. Richtig. $L = \{1^n \mid n = 1 \text{ oder } n \text{ ist prim}\}$ ist nicht regulär, aber $L^* = \Sigma^*$ ist regulär.
4. Falsch. Setze $L_i = \{a^i b^i\}$. Für jedes $i \in \mathbb{N}$ ist L_i regulär und $\mathbb{N}$ ist abzählbar. Somit ist $\mathcal{L} = \{L_i \mid i \in \mathbb{N}\}$ eine abzählbare Menge regulärer Sprachen. Aber $\bigcup_{i \in \mathbb{N}} L_i = \{a^i b^i \mid i \in \mathbb{N}\}$, die Vereinigung der Sprachen in $\mathcal{L}$, ist nicht regulär.
5. Falsch. Sei $A = \varnothing$ und $B = \{\epsilon\}$. Dann ist $A = \varnothing = AB$, aber $A = \varnothing \neq \{\epsilon\} = B^*$.
6. Richtig. Ist eine Kante mit einem Wort aw, $|w| > 0$ beschriftet, so kann sie durch die Kanten a und w verbunden mit einem neuen Knoten ersetzt werden, ohne die Sprache des Automaten zu verändern. Dies lässt sich wiederholen, bis alle Beschriftungen die Länge 0 oder 1 haben. Ein solcher Automat lässt sich aber als $\boldsymbol{\epsilon}$-NFA auffassen.
7. Falsch. Durch Einsetzen von $A \equiv a^*$ in $A \equiv aA$ erhalten wir $a^* \equiv aa^*$. Dies gilt aber nicht, denn $\epsilon \in L(a^*)$ aber $\epsilon \notin L(aa^*)$.
8. Richtig. Es gilt $L(G) = L(a^* b^*)$.

Richtige Antwort: 0,5 Punkte
Begründung auch richtig/sinnvoll: 0,5 Punkte

Aufgabe 2 (8 Punkte)

Sei $\Sigma = \{0, 1\}$ ein Alphabet.

1. Eliminieren Sie die Vorkommen von $\emptyset$ in $\alpha = a^*(b \mid a \mid \emptyset)((a\emptyset)^* \mid b\epsilon)$ soweit wie möglich. Das heißt, finden Sie einen regulären Ausdruck β mit $L(\alpha) = L(\beta)$ und $\beta = \emptyset$ oder $\emptyset$ kommt in β nicht vor.

2. Konstruieren Sie nun eine Funktion f, die aus einem regulären Ausdruck α über Σ die Vorkommen von $\emptyset$ soweit wie möglich eliminiert. Das heißt, $L(\alpha) = L(f(\alpha))$ und es gilt $f(\alpha) = \emptyset$ oder $\emptyset$ kommt in $f(\alpha)$ nicht vor.

   Hinweis: Definieren Sie f durch Rekursion über den Aufbau regulärer Ausdrücke.

3. Sei L eine Sprache über Σ.

   Für zwei gleichlange Wörter $u = a_1 \cdots a_n$ und $v = b_1 \cdots b_n$ definieren wir $d(u, v)$ als die Anzahl der Positionen $1 \leq i \leq n$, so dass $a_i \neq b_i$ ist. Dieser Wert ist die *Hamming-Distanz* von u und v, und wir nennen

   $$\mathcal{H}_1(L) = \{u \in \Sigma^* \mid \exists v \in L.\ |u| = |v| \wedge d(u, v) \leq 1\}$$

   die *Hamming-Sprache* von L.

   Zeigen Sie: Ist L regulär, so ist auch $\mathcal{H}_1(L)$ regulär. Konstruieren Sie dazu einen endlichen Automaten für $\mathcal{H}_1(L)$ und begründen Sie dessen Korrektheit.

Lösungsvorschlag

1. Eine Lösung ist $\alpha = a^*(b \mid a)(\epsilon \mid b)$.

2. Definiere f wie folgt:

   $$f(0) = 0 \qquad f(1) = 1 \qquad f(\epsilon) = \epsilon \qquad f(\emptyset) = \emptyset$$

   $$f(\alpha\beta) = \begin{cases} \emptyset & \text{falls } f(\alpha) = \emptyset \text{ oder } f(\beta) = \emptyset \\ f(\alpha)f(\beta) & \text{sonst} \end{cases}$$

   $$f(\alpha \mid \beta) = \begin{cases} \emptyset & \text{falls } f(\alpha) = \emptyset = f(\beta) \\ f(\alpha) & \text{falls } f(\alpha) \neq \emptyset = f(\beta) \\ f(\beta) & \text{falls } f(\alpha) = \emptyset \neq f(\beta) \\ f(\alpha) \mid f(\beta) & \text{sonst} \end{cases}$$

   $$f(\alpha^*) = \begin{cases} \epsilon & \text{falls } f(\alpha) = \emptyset \\ f(\alpha)^* & \text{sonst} \end{cases}$$

3. Da L regulär ist, gibt es einen NFA $M = (Q, \Sigma, \delta, q_0, F)$, der L akzeptiert. Wir zeigen die Regularität der Hammingsprache, indem wir aus M einen NFA M' für $\mathcal{H}_1(L)$ konstruieren. Setze

$$M' = (\{1,2\} \times Q, \Sigma, \delta', (1, q_0), \{1,2\} \times F)$$

wobei wir für alle $a \in \Sigma$, $q \in Q$ die Übergangsrelation δ' wie folgt definieren:

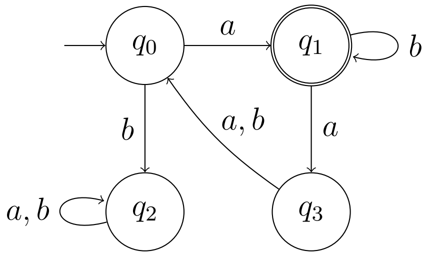
$$\begin{aligned}
\delta'((1,q),a) &= \{1\} \times \delta(q,a) \cup \bigcup_{b \in \Sigma} (\{2\} \times \delta(q,b)) \\
\delta'((2,q),a) &= \{2\} \times \delta(q,a)
\end{aligned}$$

Der Automat besteht also aus zwei Kopien des ursprünglichen Automaten und erlaubt es, nichtdeterministisch von der ersten in die zweite Kopie zu springen. Die Idee dahinter ist die folgende: Entweder ist ein Wort $w \in \mathcal{H}_1(L)$ bereits in L, und kann dann mit der ersten Kopie von M akzeptiert werden. Oder es gibt ein Wort w' in L, so dass sich w und w' an genau einer Stelle unterscheiden, also $w = uav$ und $w' = ubv$ mit $a \neq b$. Dann liest der Automat u in der ersten Kopie und springt bei a nichtdeterministisch in die zweite Kopie. Dabei wählt er den Zustand, den der Originalautomat beim Akzeptieren von w' nach dem Lesen von ub erreicht hätte. Die zweite Kopie simuliert exakt den Originalautomat M, so dass nach dem Lesen von uav ein Endzustand erreicht werden kann.

Aufgabe 3 (8 Punkte)

Sei $\Sigma = \{a, b\}$ ein Alphabet.

1. Konstruieren Sie durch schrittweises Lösen entsprechender Gleichungen einen regulären Ausdruck für die Sprache, die durch den folgenden DFA A_1 beschrieben wird:



   Verwenden Sie beim schrittweisen Lösen nur jeweils einen elementaren Schritt und machen Sie deutlich, aus welchen Gleichungen neue Gleichungen abgeleitet wurden.

2. Finden Sie einen DFA A_2, so dass $L(A_2) = L(X_2)$ mit folgenden Gleichungen gilt:

$$X_1 \equiv aX_1 \mid bX_2 \mid \boldsymbol{\epsilon} \qquad X_2 \equiv aa^*bX_1 \mid bX_2$$

   Geben Sie für den Automaten die Übergangsfunktion entweder graphisch oder als Tabelle an.

   Hinweis: Sie müssen zum Lösen der Aufgabe nicht die Gleichungen lösen.

Lösungsvorschlag

1. Die entsprechenden Gleichungen sind:

$$X_0 \equiv aX_1 \mid bX_2 \tag{1}$$
$$X_1 \equiv aX_3 \mid bX_1 \mid \boldsymbol{\epsilon} \tag{2}$$
$$X_2 \equiv (a \mid b)X_2 \tag{3}$$
$$X_3 \equiv (a \mid b)X_0 \tag{4}$$

   Aus (4) ergibt sich:

$$X_3 \equiv (a \mid b)X_0 \equiv (a \mid b)X_0 \mid \emptyset \overset{\text{Arden}}{\equiv} (a \mid b)^*\emptyset \equiv \emptyset \tag{5}$$

   Einsetzen von (4) in (2) ergibt:

$$X_1 \equiv a(a \mid b)X_0 \mid bX_1 \mid \boldsymbol{\epsilon} \overset{\text{Arden}}{\equiv} b^*(a(a \mid b)X_0 \mid \boldsymbol{\epsilon}) \tag{6}$$

   Einsetzen von (5) und (6) in (1) ergibt schließlich:

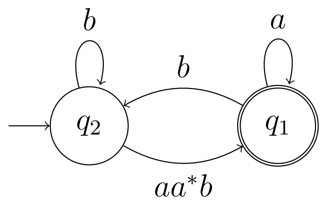
$$X_0 \equiv ab^*(a(a \mid b)X_0 \mid \boldsymbol{\epsilon}) \mid b\emptyset \equiv ab^*a(a \mid b)X_0 \mid ab^* \overset{\text{Arden}}{\equiv} (ab^*a(a \mid b))^*ab^*$$

   Es gilt nun: $L(A_1) = L(X_0) = L((ab^*a(a \mid b))^*ab^*)$.

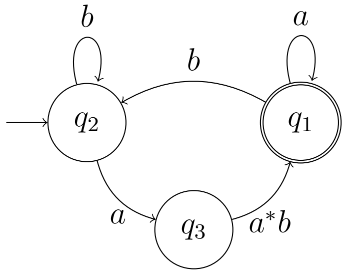
2. Wir kehren hier das Ablesen eines Gleichungssystems von einem DFA um.

   Aus der Struktur der Gleichungen erkennen wir, dass X_1 mit Endzustand des DFA A_2 korrespondieren muss. Da laut Aufgabenstellung $L(A_2) = L(X_2)$ gelten soll, muss daher X_2 der reguläre Ausdruck des Anfangszustand von A_2 sein.

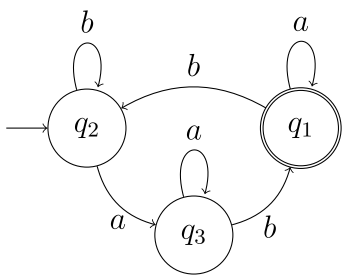
   Wir konstruieren nun A_2 schrittweise und verwenden dazu endliche Automaten, deren Übergänge durch reguläre Ausdrücke beschriftet sind. Zunächst können wir folgenden Automaten aus den Gleichungen ablesen:



   Den unteren Übergang kann man nun aufspalten. Dies entspricht der Umkehrung des Einsetzens der Gleichung $X_3 \equiv a^*bX_1$ in die Gleichung $X_2 \equiv aX_1 \mid bX_2$.



   Die Gleichung $X_3 \equiv a^*bX_1$ kann offensichtlich nur durch die Anwendung von Ardens Lemma erhalten worden sein, und zwar aus der Gleichung $X_3 \equiv aX_3 \mid bX_1$. Daher können wir den Automaten wie folgt anpassen und erhalten den gesuchten DFA A_2:

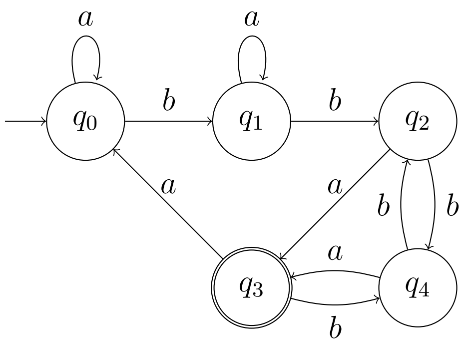


   Alternative Lösung: Man kann die Gleichungen auch nach X_2 auflösen und aus dem erhaltenen regulären Ausdruck einen ϵ-NFA konstruieren, den man dann über einen NFA in einen DFA umwandeln kann.

Aufgabe 4 (8 Punkte)

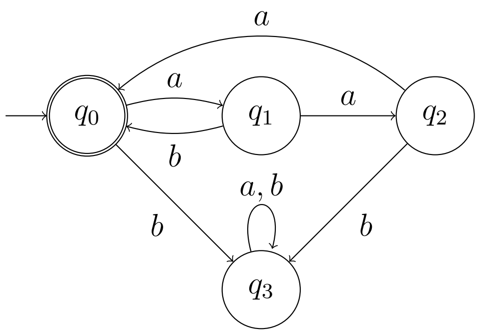
Sei $\Sigma = \{a, b\}$ ein Alphabet.

1. Verwenden Sie das in der Vorlesung vorgestellte Verfahren, um den folgenden deterministischen Automaten A_1 zu minimieren.



   Stellen Sie dazu die Tabelle aus der Vorlesung auf und geben Sie in der Tabelle zu jedem unterscheidbaren Paar von Zuständen ein kürzestes Wort an, mit dem diese unterschieden werden können. Verwenden Sie die aufgestellte Tabelle, um den Quotientenautomaten zu konstruieren.

2. Betrachten Sie den folgenden minimalen DFA A_2:



   (a) Geben Sie die Anzahl der Äquivalenzklassen von $\equiv_{L(A_2)}$ an.

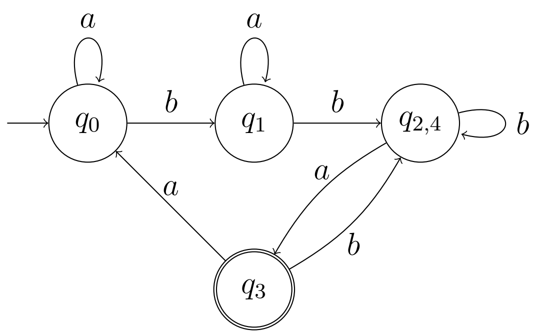
   (b) Geben Sie zu jeder Äquivalenzklasse von $\equiv_{L(A_2)}$ genau zwei Repräsentanten an.

Lösungsvorschlag

1. Die Tabelle sieht folgendermaßen aus, wobei Zellen mit X beschriftet sind, wenn die entsprechenden Zustände direkt durch Endzustände unterschieden werden können:

	0	1	2	3
1	ba			
2	a	a		
3	X	X	X	
4	a	a		X

Der dazugehörige Quotientenautomat ist dann wie folgt:



2. (a) Die Relation $\equiv_{L(A_2)}$ besteht aus genau vier Äquivalenzklassen.
   (b) Die kürzesten Repräsentanten der vier Äquivalenzklassen sind:

$$\epsilon, ab \qquad a, aba \qquad aa, abaa \qquad b, ba \text{ (oder } bb\text{)}$$

Aufgabe 5 (8 Punkte)

Sei $\Sigma = \{a, b\}$ ein Alphabet.

1. Betrachten Sie die kontextfreie Grammatik $G = (\{S\}, \Sigma, \{S \to aaS \mid baS \mid \epsilon\}, S)$. Zeigen Sie für alle Wörter $w = a_1 \cdots a_n \in \Sigma^*$ (mit $a_1, \ldots, a_n \in \Sigma$): Ist $|w|$ gerade und gilt $a_i = a$ für alle geraden i mit $1 \leq i \leq n$, so gilt $w \in L(G)$.

2. Zeigen Sie mit dem Pumping-Lemma, dass die Sprache $L = \{w\$w \mid w \in \Sigma^*\}$ nicht kontextfrei ist.

Lösungsvorschlag

1. Für die Eigenschaft „$w \in \Sigma^*$ hat gerade Länge und ist von der Form $w = a_1 \cdots a_n$ mit $a_i = a$ für alle geraden $1 \leq i \leq n$“ schreiben wir $P(w)$. Wir beweisen die Aussage per vollständiger Induktion über die Länge des Wortes.

   - $|w| = 0$: Dann ist $w = \epsilon$ und $w \in L(G)$, denn $S \to_G \epsilon$.
   - $|w| = 1$: Kein Wort w der Länge 1 erfüllt $P(w)$. Also sind alle Wörter der Länge 1, die $P(w)$ erfüllen, auch in $L(G)$ enthalten.
   - $|w| > 1$: Da $|w|$ gerade ist und jeder zweite Buchstabe ein a ist, hat w notwendigerweise die Form $w = xau$ für ein $x \in \Sigma$ und $u \in \Sigma^*$. Da xa ein Wort gerader Länge ist, gilt auch $P(u)$. Da $|u| = |w| - 2$ ist, können wir die Induktionshypothese anwenden und damit gilt $u \in L(G)$, also $S \to_G^* u$. Dann ist aber auch $S \to_G xaS \to_G^* xau$, also $w \in L(G)$.

2. Wir nehmen an, L wäre kontextfrei. Sei n eine Pumping-Lemma-Zahl und wähle $z = a^n b^n \$ a^n b^n \in L$. Nach dem Pumping-Lemma für kontextfreie Sprachen existiert jetzt eine Zerlegung $z = uvwxy$, so dass $vx \neq \epsilon$, $|vwx| \leq n$ und $\forall i \in \mathbb{N}.\ uv^i wx^i y \in L$ gelten.

   Wir untersuchen die möglichen Zerlegungen von z.

   - \$ kommt in v oder x vor: Dann enthält uv^0wx^0y das Zeichen \$ nicht, ist also nicht in L (denn jedes Wort in L enthält genau ein \$).
   - \$ kommt in w vor: Wegen $|vwx| \leq n$ gibt es dann $k, l \in \mathbb{N}$ mit $k + l > 0$ und $v = b^k$, $x = a^l$. Dann ist aber $uv^0wx^0y = a^n b^{n-k} \$ a^{n-l} b^n$ und $b^{n-k} \neq b^n$ oder $a^{n-l} \neq a^n$ und damit $uv^0wx^0y \notin L$.
   - \$ kommt in u vor: Dann ist (wegen $vx \neq \epsilon$) in uv^0wx^0y das Wort vor dem \$ echt länger als das Wort hinter dem \$, also $uv^0wx^0y \notin L$.
   - \$ kommt in y vor: Dann ist (wegen $vx \neq \epsilon$) in uv^0wx^0y das Wort vor dem \$ echt kürzer als das Wort hinter dem \$, also $uv^0wx^0y \notin L$.

   Es gibt also keine Zerlegung von z, die alle drei Bedingungen erfüllt. Also muss die Annahme falsch sein und damit ist L nicht kontextfrei.