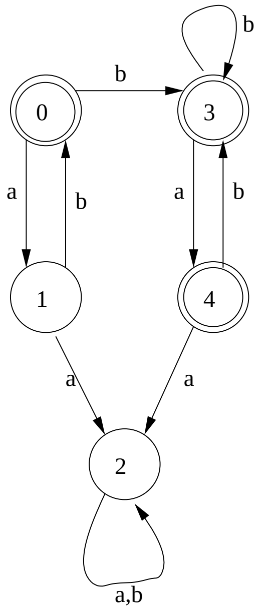
**Theoretische Informatik (WS04)**
Prof. Dr. Raimund Seidel
Wei Ding, Dierk Johannes
Lösungsvorschlag zu Aufgabenblatt 4

# Aufgabe 1

a) Die Sprache $L_a = \{w \in \{a,b\}^* | \#_a(w) \leq \#_b(w)$ und $aa$ ist kein Teilwort von $w\}$ ist regulär. Der folgende Automat erkennt sie:



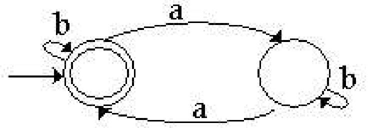

Aus diesem Automaten können wir folgenden regulären Ausdruck ableiten: $(ab)^* + bb^*a + b(b^*ab)^*$.

b) Die Sprache $L_b = \{w \in \{a,b\}^* | \#_a(w) \leq \#_b(w)$ und $aaa$ ist kein Teilwort von $w\}$ ist irregulär. Beweis über Myhill-Nerode (Argumentation über Anzahl der Fortsetzungssprachen).

Wir betrachten für $n \in \mathbb{N}$ Wörter $w_n = (aab)^n$. Es gilt: $b^0, b^1, \ldots, b^{n-1} \notin C_F(w_n)$ und $b^n, b^{n+1}, \ldots \in C_F(w_n)$. Es sei nun $m \neq n$, etwa $m < n$. Dann ist $b^m \in C_F(w_m)$ aber $b^m \in C_F(w_n)$, d. h. $C_F(w_m) \neq C_F(w_n)$. Die Fortsetzungssprachen $C_F(w_n)$ sind also paarweise verschieden, und es sind unendlich viele.

c) Die Sprache $L_c = \{w \in \{a,b\}^* | \#_a(w) \bmod 2 = 0\}$ ist regulär.

Beweis:




Regulärer Ausdruck (aus Transitionsgraphen abgeleitet): $(b^*(ab^*ab^*))$.

# Theoretische Informatik (WS04)

Prof. Dr. Raimund Seidel
Wei Ding, Dierk Johannes
Lösungsvorschlag zu Aufgabenblatt 4

## Aufgabe 2

In dieser Aufgabe wenden wir den Algorithmus zur Berechnung des Minimalautomatens an.

- **Schritt 1:** Wir erstellen eine Tabelle aller Zustandspaare $\{z, z'\}$ mit $z \neq z'$.

- **Schritt 2:** Wir markieren alle Paare $\{z, z'\}$ mit $z \in F$ und $z' \notin F$.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | x | x | x | | | | |
| 4 | | | | x | | | |
| 5 | x | x | x | | x | | |
| 6 | | | | x | | x | |
| 7 | | | | x | | x | |

- **Schritt 3:** Wir testen für jedes noch unmarkierte Paar $\{z, z'\}$ und jedes $a \in \Sigma$, ob das Paar $\{\delta(z, a), \delta(z', a)\}$ bereits markiert ist. Falls ja, dann markieren wir auch $\{z, z'\}$.

$\{\delta(1, a), \delta(0, a)\} = \{2, 1\}$
$\{\delta(1, b), \delta(0, b)\} = \{3, 5\}$

$\{\delta(2, a), \delta(0, a)\} = \{2, 1\}$
$\{\delta(2, b), \delta(0, b)\} = \{3, 5\}$

$\{\delta(4, a), \delta(0, a)\} = \{1, 1\}$
$\{\delta(4, b), \delta(0, b)\} = \{0, 5\}$
$\Rightarrow$ wir markieren $\{4, 0\}$
.

$\{\delta(6, a), \delta(0, a)\} = \{5, 1\}$
$\{\delta(6, b), \delta(0, b)\} = \{7, 5\}$
$\Rightarrow$ wir markieren $\{6, 0\}$.

$\{\delta(7, a), \delta(0, a)\} = \{2, 1\}$
$\{\delta(7, b), \delta(0, b)\} = \{0, 5\}$
$\Rightarrow$ wir markieren $\{7, 0\}$.

$\{\delta(2,a),\delta(1,a)\} = \{2,2\}$
$\{\delta(2,b),\delta(1,b)\} = \{3,3\}$

$\{\delta(4,a),\delta(1,a)\} = \{1,2\}$
$\{\delta(4,b),\delta(1,b)\} = \{0,3\}$
$\Rightarrow$ wir markieren $\{4,1\}$.

$\{\delta(6,a),\delta(1,a)\} = \{5,2\}$
$\{\delta(6,b),\delta(1,b)\} = \{7,3\}$
$\Rightarrow$ wir markieren $\{6,1\}$.

$\{\delta(7,a),\delta(1,a)\} = \{2,2\}$
$\{\delta(7,b),\delta(1,b)\} = \{0,3\}$
$\Rightarrow$ wir markieren $\{7,1\}$.

$\{\delta(4,a),\delta(2,a)\} = \{1,2\}$
$\{\delta(4,b),\delta(2,b)\} = \{0,3\}$
$\Rightarrow$ wir markieren $\{4,2\}$.

$\{\delta(6,a),\delta(2,a)\} = \{5,2\}$
$\{\delta(6,b),\delta(2,b)\} = \{7,3\}$
$\Rightarrow$ wir markieren $\{6,2\}$.

$\{\delta(7,a),\delta(2,a)\} = \{2,2\}$
$\{\delta(7,b),\delta(2,b)\} = \{0,3\}$
$\Rightarrow$ wir markieren $\{7,2\}$
.

$\{\delta(5,a),\delta(3,a)\} = \{3,4\}$
$\{\delta(5,b),\delta(3,b)\} = \{6,3\}$
$\Rightarrow$ wir markieren $\{5,3\}$.

$\{\delta(6,a),\delta(4,a)\} = \{5,1\}$
$\{\delta(6,b),\delta(4,b)\} = \{7,0\}$
$\Rightarrow$ wir markieren $\{6,4\}$.

$\{\delta(7,a),\delta(4,b)\} = \{2,1\}$
$\{\delta(7,b),\delta(4,b)\} = \{0,0\}$

$\{\delta(7,a),\delta(6,a)\} = \{2,5\}$
$\{\delta(7,b),\delta(6,b)\} = \{0,7\}$

$\Rightarrow$ wir markieren $\{6, 7\}$.

Wir bekommen folgende Tabelle:

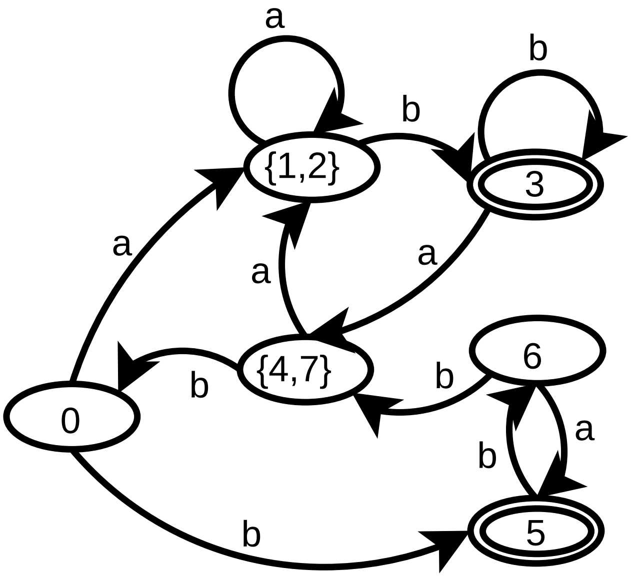
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | x | x | x | | | | |
| 4 | x | x | x | x | | | |
| 5 | x | x | x | x | x | | |
| 6 | x | x | x | x | x | x | |
| 7 | x | x | x | x | | x | x |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |

- **Schritt 4:** Wir wiederholen Schritt 3, bis sich keine Änderung in der Tabelle mehr ergibt.

  $\{\delta(1, a), \delta(0, a)\} = \{2, 1\}$
  $\{\delta(1, b), \delta(0, b)\} = \{3, 5\}$
  $\Rightarrow$ wir markieren $\{1, 0\}$.

  $\{\delta(2, a), \delta(0, a)\} = \{2, 1\}$
  $\{\delta(2, b), \delta(0, b)\} = \{3, 5\}$
  $\Rightarrow$ wir markieren $\{2, 0\}$.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | x | | | | | | |
| 2 | x | | | | | | |
| 3 | x | x | x | | | | |
| 4 | x | x | x | x | | | |
| 5 | x | x | x | x | x | | |
| 6 | x | x | x | x | x | x | |
| 7 | x | x | x | x | | x | x |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |




## Aufgabe 3

Es gab diverse Ansätze zur Sprachentheorie und aus den verschiedenen Ansätzen entwickelten sich unterschiedliche Sprachbeschreibungen, von denen sich viele jedoch als äquivalent erwiesen. Auch GKB-Automaten sind nur eine andere Beschreibungsform regulärer Sprachen, wie im folgenden in einem konstruktiven Beweis gezeigt werden soll.
Voraussetzung: $DEA = NEA$
Behauptung: $GKB = DEA = NEA$
z.Z.: $DEA \subseteq GKB \wedge GKB \subseteq NEA$
Beweis:
$DEA \subseteq GKB$:
Zu einem $DEA$ $M = (\Sigma, Q, s, F, \Delta)$ lässt sich wie folgt ein $GKB$ $N = (\Sigma', V, E, x, y, B)$ konstruieren, der die gleiche Sprache entscheidet:

1. Das Alphabet bleibt unverändert: $\Sigma' := \Sigma$
2. Für alle Knoten, sowie für alle Kanten des $DEA$ gibt es einen Knoten im $GKB$:
   $Q_v := Q$,
   $Q_e := \{Q_{AsB} | (A, s, B) \in \Delta\}$,
   Außerdem braucht man einen seperaten Endknoten f ($f \notin Q_v \wedge f \notin Q_e$), da der $GKB$ im Gegensatz zum $DEA$ genau einen Endzustand hat.
   $V := Q_v \cup Q_e \cup \{f\}$
3. Implizit ergibt sich $y := f$.
4. Der Startzustand bleibt erhalten: $x := s$

5. Die Kanten des $GKB$ ergeben sich aus denen des $DEA$ wie folgt:
$V := \{(a,b) | \exists (A,s,B) \in \Delta : (a = A \wedge b = Q_{AsB}) \vee (a = Q_{AsB} \wedge b = B) \vee (a \in F \wedge b = y)\}$
d.h. für eine Kante von $A$ nach $B$ im $DEA$ gibt es einen Knoten $Q_{AsB}$ im $GKB$, zusammen mit 2 Kanten, eine von $A$ nach $Q_{AsB}$ und eine von $Q_{AsB}$ nach $B$. Außerdem sind alle Knoten, die im $DEA$ Endknoten waren, im GKB mit dem neu eingeführten Endknoten verbunden.

6. Die Beschriftung basiert nun auf folgender Idee: Wenn es im $DEA$ eine Kante gibt, o.B.d.A. $(A,s,B) \in \Delta$, dann bedeutet dies, daß auf dem Pfad von $A$ nach $B$ das Zeichen $s$ konsumiert wird. Dies muß nun auch im $GKB$ geschehen, dort allerdings in einem Knoten, nicht auf einer Kante. Um dies zu erreichen wurden alle ursprünglichen Kanten aufgespalten und jeweils ein neuer Knoten eingefügt, sodaß man in diesem neuen Knoten nun das Zeichen $s$ konsumieren kann und somit immer noch auf dem Pfad von $A$ nach $B$ genau das Zeichen $s$ konsumiert. In allen Knoten, die bereits im $DEA$ vorhanden waren, wird nur $\epsilon$ gelesen, ebenso im neuen Endzustand:
$\forall q \in (Q_v \cup \{y\}) : B(q) := \epsilon$
$\forall q \in (V \setminus (Q_v \cup \{y\})) \exists A, B, s : (A,q) \in E \wedge (q,B) \in E \wedge (A,s,B) \in \Delta$ (da so definiert)
Dann definiert man: $B(q) := s$. Man beachte, daß ein einzelnes Zeichen aus $\Sigma$ bereits ein Wort ist.

Der Automat ist weder deterministisch, noch minimal, es ist jedoch ein $GKB$-Automat, der offensichtlich die gleiche Sprache entscheidet und mehr war nicht verlangt.
Dieses war der leichte Teil: In der Chomsky-Hierarchie stehen die regulären Sprachen ganz unten, es ist also nicht verwunderlich, daß $GKB$-Automaten diese unterste Sprachklasse entscheiden. Nicht so selbstverständlich ist, daß sie nicht auch mehr können:
$GKB \subseteq NEA$:
Zu einem $GKB$ $N = (\Sigma', V, E, x, y, B)$ lässt sich wie folgt ein $NEA$ $M = (\Sigma, Q, s, F, \Delta)$ konstruieren, der die gleiche Sprache entscheidet:

Für ein einzelnes Wort $w$, welches als Knotenbeschriftung im $GKB$ vorkommt, lässt sich ein $DEA$ konstruieren, der genau die Sprache $L_w = \{w\}$ entscheidet. Dies ist iterativ möglich, durch Induktion über die Wortlänge:
Fall $n := |w| = 0$: $w = \epsilon$ trivial
$n \rightarrow n+1$: Sei $a \in \Sigma$, $u \in \Sigma^*$, $w = au$ die Konkatenation der beiden und $M = (\Sigma, G, s, F, \Delta)$ ein $DEA$, der die Sprache $\{u\}$ entscheidet. Sei ferner o.B.d.A. $g \notin G$. Dann ist $M' := (\Sigma, G \cup \{g\}, g, F, \Delta \cup \{(g,a,s)\})$ offensichtlich ein Automat, der $\{w\}$ entscheidet.
Somit kann man zu jedem Zustand des $GKB$ einen $DEA$ konstruieren, der das gleiche Wort liest wie dieser Zustand, man kann die Zustände also sozusagen durch $DEAs$ ersetzen. Im $GKB$ gibt es aber auch Kanten, diese werden als $\epsilon$-Kanten übernommen, außerdem müssen Start- und Endzustände noch passend gewählt werden: Sei $|V| = n$ und sind $q_1, q_2, \ldots, g_n \in V$ die Knoten des $GKB$-Automaten und seien ferner $M_{q_1}, M_{q_2}, \ldots, M_{q_n}$, $M_{q_i} = (\Sigma, Q_{q_i}, s_{q_i}, F_{q_i}, \Delta_{q_i}) \forall i \in \{1, \ldots, n\}$ die dazu korrespondierenden $DEAs$ (o.B.d.A $\bigcap_{i=1}^{n} Q_{q_i} = \emptyset$), dann setzt sich der Gesamt-$NEA$ $M = (\Sigma, Q, s, F, \Delta)$ wie folgt zusammen:

Die Zustandmenge ist die disjunkte Vereinigung der Zustände aller $DEAs$
$G := \bigcup_{i=1}^{n} Q_{q_i}$
Die Menge der Endzustände enthält die Endzustände des Automaten, der zum Endzustand des $GKB$-Automaten korrespondiert.
$F := \{f \in G | f \in F_{q_i} \wedge q_i = y\}$
Analog ist der Startzustand des $NEA$ gleich dem Startzustand des $DEA$, der dem Startzustand des $GKB$-Atomaten entspricht.
$s := s_{q_i}$, $q_i = x$
Bleiben noch die Übergänge:
$\Delta := \left(\bigcup_{i=1}^{n} \Delta_{q_i}\right) \cup \{(A, \epsilon, B) | \exists q_i, q_j \in V, \exists (q_i, q_j) \in E \wedge A \in F_{q_i} \wedge B = s_{q_j}\}$
Beachte: Die $DEAs$ haben alle jeweils genau einen Endzustand, wodurch in der Praxis manches leichter würde. Obiges Beispiel lässt sich aber ohne Probleme so erweitern, daß es immer noch funktioniert, wenn die Knoten des $GKB$ nicht mit Wörtern, sondern mit regulären Ausdrücken beschriftet wären. In dem Fall könnte man daraus $NEAs$ generieren und diese zu minimalen $DEAs$ umformen, die dann aber nicht mehr zwingend jeweils genau einen Endzustand haben müssen. Aber insbesondere würde auch diese Erweiterung die Mächtigkeit der $GKB$-Automaten nicht erhöhen!

# Aufgabe 4

Die Aussage stimmt und kann folgendermassen bewiesen werden:

„$\Longrightarrow$“

Die Aussage $L(M) = \Sigma^*$ bedeutet, daß $M$ alle Wörter akzeptiert. Dann akzeptiert $M$ auch alle Wörter der Länge höchstens $n$.

„$\Longleftarrow$“

Voraussetzung: Der DEA $M$ erkennt alle Wörter mit Länge höchstens $n$. Es sei $w$ ein Wort kleinster Länge (mit Länge $|w| > n$), das nicht erkannt wird.

Fall 1: Das Wort wird deshalb nicht erkannt, weil es in einem Zustand keine anwendbare Regel gibt. Es sei etwa $w = w_1 a w_2$ und wir befinden uns nach Abarbeitung von $w_1$ im Zustand $q$ und es existiert keine Regel für $a$. (Es gilt $|w_1 a| < |w|$.) Dann ist $w_1 a$ ebenfalls ein Wort, das nicht akzeptiert wird. Seine Länge ist kürzer als die von $w$. Dies steht im Widerspruch zur Minimalität von $w$. Also tritt dieser Fall nicht ein.

Fall 2: Nach Abarbeiten von $w$ befinden wir uns *nicht* in einem Endzustand. (Wir präsentieren nur die Idee, dies sollte formaler ausgeführt werden.) Der Automat hat $n$ Zustände. Bei der Abarbeitung von $w$, $|w| > n$, muß ein Zustand $q$ doppelt durchlaufen werden, wir haben also eine Schleife. Das Wort zerlege sich etwa in $w = w_1 w_2 w_3$ wobei $w_2$ den Teil bezeichne, der der Schleife entspricht. Wir setzen $w' = w_1 w_3$. Dieses Wort durchläuft bei der Abarbeitung bis auf die Schleife die gleichen Zustände wie $w$. Somit wird es auch nicht akzeptiert. Allerdings ist $|w'| < |w|$, was der Minimalität von $|w|$ widerspricht.