Theoretische Informatik und Logik
Übungsblatt 1 (2019W)
Lösungsvorschlag

Anmerkung: Zeichen mit reinem Symbolcharakter sind im Folgenden unterstrichen. Sie können, müssen das aber nicht in Ihrer Ausarbeitung beibehalten.

Aufgabe 1.1 Sei $L_n = \{w\underline{\heartsuit}w \mid w \in \{\underline{\mathsf{a}}_i \mid 1 \leq i \leq n\}^+\}$ für ein fixes n.

> (L_n enthält also Wörter, bei denen das Teilwort w nach dem Trennsymbol $\underline{\heartsuit}$ wiederholt wird. So gilt z.B.: $\underline{\mathsf{a}}_1\underline{\mathsf{a}}_2\underline{\mathsf{a}}_3\underline{\heartsuit}\underline{\mathsf{a}}_1\underline{\mathsf{a}}_2\underline{\mathsf{a}}_3 \in L_n$, aber $\underline{\mathsf{a}}_1\underline{\mathsf{a}}_2\underline{\mathsf{a}}_3\underline{\heartsuit}\underline{\mathsf{a}}_3\underline{\mathsf{a}}_2\underline{\mathsf{a}}_1 \notin L_n$ und $\underline{\mathsf{a}}_1\underline{\mathsf{a}}_2\underline{\heartsuit}\underline{\mathsf{a}}_3 \notin L_n$.)

Geben Sie eine deterministische Turingmaschine M_n an, welche die Sprache L_n akzeptiert, und erläutern Sie (jeweils) auch kurz verbal die Arbeitsweise Ihrer Maschine. Es steht Ihnen dabei frei, ob Sie das auf Folie 26 definierte Modell (mit einem Band) oder das auf Folie 72 definierte Modell (mit zwei Bändern, einem Eingabe- und einem Arbeitsband) verwenden (oder beide :-)).

Lösung

<u>**Variante 1:**</u>

Wir definieren eine (deterministische) Turingmaschine

$$M = (\{q_i \mid 0 \leq i \leq 2n+4\}, \{\underline{\mathsf{a}}_i \mid 1 \leq i \leq n\} \cup \{\underline{\heartsuit}\}, \{\underline{\mathsf{a}}_i \mid 1 \leq i \leq n\} \cup \{\underline{\heartsuit}, X, Y, B\}, \delta, q_0, B, \{q_{2n+4}\})$$

wobei

$$\begin{array}{rlr}
\delta = & \{(q_0, \underline{\mathsf{a}}_i; q_i, X, R) \mid 1 \leq i \leq n\} & 1 \\
\cup & \{(q_i, \underline{\mathsf{a}}_i; q_i, \underline{\mathsf{a}}_i, R) \mid 1 \leq i \leq n\} & 2 \\
\cup & \{(q_i, \underline{\heartsuit}; q_{n+i}, \underline{\heartsuit}, R) \mid 1 \leq i \leq n\} & 3 \\
\cup & \{(q_{n+i}, Y; q_{n+i}, Y, R) \mid 1 \leq i \leq n\} & 4 \\
\cup & \{(q_{n+i}, \underline{\mathsf{a}}_i; q_{2n+1}, Y, L) \mid 1 \leq i \leq n\} & 5 \\
\cup & \{(q_{2n+1}, z; q_{2n+1}, z, L) \mid z \in \{\underline{\mathsf{a}}_i \mid 1 \leq i \leq n\} \cup \{\underline{\heartsuit}, Y\}\} & 6 \\
\cup & \{(q_{2n+1}, X; q_{2n+2}, X, R)\} & 7 \\
\cup & \{(q_{2n+2}, \underline{\mathsf{a}}_i; q_i, X, R) \mid 1 \leq i \leq n\} & 8 \\
\cup & \{(q_{2n+2}, \underline{\heartsuit}; q_{2n+3}, \underline{\heartsuit}, R)\} & 9 \\
\cup & \{(q_{2n+3}, Y; q_{2n+3}, Y, R)\} & 10 \\
\cup & \{(q_{2n+3}, B; q_{2n+4}, B, S)\} & 11
\end{array}$$

Idee:

1: Das erste Symbol $\underline{\mathsf{a}}_i$ in der linken Worthälfte wird mit X markiert und davon abhängig in den Zustand q_i gewechselt.

2, 3: Die Maschine wandert über alle weiteren Symbole $\underline{\mathsf{a}}_i$ in der linken Worthälfte bis zum Trennzeichen $\underline{\heartsuit}$.

4: Etwaige in der rechten Worthälfte angetroffene Symbole Y werden überlesen, die Maschine wandert weiter nach rechts.

5: Findet sich in der rechten Worthälfte ein entsprechendes Symbol $\underline{\mathsf{a}}_i$, so wird dieses mit Y markiert.

6: Nun wandert die Maschine wieder zurück nach links.

7: Das am weitesten rechts stehende Symbol X in der linken Worthälfte wird gefunden.

8: Das nächste Symbol $\underline{\mathsf{a}}_i$ in der linken Worthälfte wird mit X markiert, die Maschine wechselt in den Zustand q_i, um erneut nach rechts zu wandern.

9: Das Trennzeichen wird gefunden.

10: Die Bandsymbole Y in der rechten Worthälfte werden überlesen.

11: Findet sich nach dem letzten Symbol Y ein Blanksymbol B, so begibt sich die Maschine in den Endzustand q_{2n+4} und akzeptiert somit die Eingabe. (Die Maschine erreicht diesen Zustand nur, wenn alle Symbole vor und nach dem Trennzeichen entsprechend markiert wurden.)

<u>Variante 2:</u>
Wir definieren eine (deterministische) Turingmaschine

$$M = (\{q_i \mid 0 \leq i \leq 4\}, \{\underline{\mathsf{a}}_i \mid 1 \leq i \leq n\} \cup \{\underline{\heartsuit}\}, \{X_i \mid 1 \leq i \leq n\} \cup \{B, Z_0\}, \delta, q_0, \{Z_0, Z_1, Z_2\}, B, \{q_4\})$$

die Übergangsfunktion δ kann z.B. folgendermaßen definiert werden:

$$\begin{array}{rll}
1: & \delta(q_0, \underline{\mathsf{a}}_i, B) = (q_1, X_i, R, R) & \text{für } 1 \leq i \leq n \\
2: & \delta(q_1, \underline{\mathsf{a}}_i, B) = (q_1, X_i, R, R) & \text{für } 1 \leq i \leq n \\
3: & \delta(q_1, \underline{\heartsuit}, B) = (q_2, B, S, L) & \\
4: & \delta(q_2, \underline{\heartsuit}, X_i) = (q_2, X_i, S, L) & \text{für } 1 \leq i \leq n \\
5: & \delta(q_2, \underline{\heartsuit}, Z_0) = (q_3, Z_0, R, R) & \\
6: & \delta(q_3, \underline{\mathsf{a}}_i, X_i) = (q_3, X_i, R, R) & \\
7: & \delta(q_3, \underline{\mathsf{Z}}_2, B) = (q_4, B, S, S) &
\end{array}$$

$1, 2$: Für jedes eingelesene Symbol $\underline{\mathsf{a}}_i$ wird ein Symbol X_i auf das Arbeitsband geschrieben. (Durch 1 wird sichergestellt, dass sich mindestens 1 Symbol vor dem Trennzeichen befindet.)

3 : Das Trennzeichen wird gefunden, der Lesekopf auf dem Arbeitsband bleibt hier vorübergehend stehen.

$4, 5$: Währenddessen bewegt sich der Lese-/Schreibkopf auf dem Arbeitsband zurück nach links, bis zum linken Begrenzungssymbol Z_0.

6 : Für jedes eingelesene Symbol a_i der rechten Worthälfte muss sich nun ein entsprechendes Symbol X_i auf dem Arbeitsband finden.

7 : Wird Z_2 auf dem Eingabeband (d.h., das Ende der Eingabe) erreicht, so sollte auch auf dem Arbeitsband ein Blanksymbol erreicht sein. M geht dann in den (einzigen) Endzustand q_4 über und akzeptiert somit die Eingabe.

Aufgabe 1.2 Sei A eine beliebige Sprache, L_u das Halteproblem (siehe z.B. Folie 49) und D eine entscheidbare Sprache. Welche der folgenden Aussagen sind wahr, welche falsch? Begründen Sie jeweils Ihre Antwort (z.B. durch Angabe eines Gegenbeispiels).

a) Falls $A \leq L_u$, dann ist A entscheidbar.

b) Falls $A \leq L_u$ und $\overline{A} \leq L_u$, dann ist A entscheidbar.

c) Falls $A \leq L_u$ und $L_u \leq A$, dann ist A rekursiv aufzählbar.

d) Falls $D \leq A$ und $A \leq D$, dann ist $A = D$.

e) Falls $D \leq A$, dann ist A entscheidbar.

f) Falls $D \leq A$, dann ist A rekursiv aufzählbar.

Lösung

a) **Falsch**. Setze $A = L_u$, dann gilt (wegen Reflexivität der $\leq$-Relation) $L_u \leq L_u$, aber L_u ist nicht entscheidbar.

b) **Richtig**. Aus $A \leq L_u$ folgt, dass A rekursiv aufzählbar ist, aus $\overline{A} \leq L_u$ folgt, dass $\overline{A}$ rekursiv aufzählbar ist. Damit ist A laut Definition (Folie 41) entscheidbar.

c) **Richtig**. Diese Aussage folgt schon aus $A \leq L_u$. (Folie 56)

d) **Falsch**. Betrachte $D = \{\underline{0}\}$ und $D' = \{\underline{1}\}$. Beide Sprachen sind entscheidbar und es gilt $D \leq D'$ mittels $f(0) = 1$ und $f(x) = 0$, für alle $x \in \{\underline{0}, \underline{1}\}^* - \{\underline{0}\}$. Analog gilt $D' \leq D$, aber $D \neq D'$.

e) **Falsch**. Sei $D = \{\}$, $\overline{D} = \{\underline{0}, \underline{1}\}^*$ und setze $A = \overline{L_u}$. Sei $x \in \overline{L_u}$ (z.B. die Kodierung der Turingmaschine, die nie hält und einfach immer nach rechts geht) und $y \notin \overline{L_u}$ (z.B. die Kodierung der Turingmaschine, die vom Initialzustand sofort in einen akzeptierenden Endzustand übergeht). Dann folgt $\{\} \leq \overline{L_u}$ mittels $f(z) = y$, für alle $z \in \{\underline{0}, \underline{1}\}^*$ und $\{\underline{0}, \underline{1}\}^* \leq \overline{L_u}$ mittels $f(z) = x$, für alle $z \in \{\underline{0}, \underline{1}\}^*$. A ist jedoch nicht rekursiv aufzählbar.

f) **Falsch**. Folgt aus der Begründung zu e).

Aufgabe 1.3 Geben Sie an, ob folgende Probleme (un)entscheidbar sind, und begründen Sie jeweils Ihre Antwort. Sofern möglich, verwenden Sie dafür den *Satz von Rice* (und geben Sie, im Falle einer nicht trivialen Eigenschaft, auch immer ein Beispiel und ein Gegenbeispiel an). (Das Alphabet ist dabei jeweils $\Sigma = \{\underline{0}, \underline{1}\}$.)

a) Hält die Turingmaschine in einem Endzustand, wenn sie mit leerem Band gestartet wird?

b) Wird die Sprache L von genau einer Turingmaschine akzeptiert?

c) Ist das Komplement der von einer Turingmaschine akzeptierten Sprache gleich $\{\underline{0}, \underline{1}\}^*$?

d) Enthält die von einer Turingmaschine M akzeptierte Sprache ein Wort, das in weniger als 1000 Schritten von M akzeptiert wird?

e) Enthält die von einer Turingmaschine akzeptierte Sprache nur Wörter w, nicht aber deren Spiegelbild w^r ?

f) Sind die von einer Turingmaschine akzeptierte Sprache und auch ihr Komplement rekursiv aufzählbar?

Lösung

a) **Nicht entscheidbar**, Satz von Rice: Eine Turingmaschine hält in einem Endzustand, wenn sie mit leerem Band gestartet wird, wenn das Leerwort ε akzeptiert wird. Es geht hier also um die Eigenschaft $P = \{L \mid \varepsilon \in L\}$, welche nicht trivial ist, denn es gilt z.B.: $\{\varepsilon\} \in P$ aber $\{\underline{1}\} \notin P$. Daher ist P aufgrund des Satzes von Rice nicht entscheidbar.

b) **Entscheidbar**. Diese Eigenschaft trifft auf keine rekursiv aufzählbare Sprache zu, d.h., $P = \{\}$, denn zu jeder rekursiv aufzählbaren Sprache gibt es unendlich viele Turingmaschinen, die sie akzeptieren. Die Frage kann also immer mit "nein" beantwortet werden, und ist somit entscheidbar.

c) **Nicht entscheidbar**, Satz von Rice: Es handelt sich um die Eigenschaft $P = \{\{\}\}$. Diese Eigenschaft kommt einer Sprache L zu, nämlich $L = \{\}$. Keine andere rekursiv aufzählbare Sprache ist in P, dementsprechend ist P nicht trivial, und damit nach dem Satz von Rice nicht entscheidbar.

(*Anmerkung:* Beachten Sie den Unterschied zwischen $P = \{\{\}\}$, der Eigenschaft die Leersprache zu sein und der leeren Eigenschaft $P = \{\}$, welche keiner rekursiv aufzählbaren Sprache zukommt, siehe auch b).)

d) **Entscheidbar**. Der Satz von Rice kann hier nicht angewendet werden. Wir beobachten jedoch Folgendes: Um ein Wort in weniger als 1000 Schritten zu akzeptieren, muss die Maschine ihre Entscheidung nach dem Einlesen von höchstens 999 Symbolen treffen. Ein einfacher Entscheidungsalgorithmus ist demnach z.B. folgender: Liste alle Wörter über Σ auf, die bis zu 999 Zeichen lang sind. Lasse die Maschine für jedes Wort bis zu höchstens 999 Schritten laufen (sofern sie nicht bereits vorher hält). Hält sie auf mindestens einem solchen Wort, so ist die Antwort "ja", ansonsten "nein".

e) **Nicht entscheidbar**, Satz von Rice: Die Eigenschaft $P = \{L \mid w \in L, w^r \notin L\}$ ist nicht trivial, denn es gilt z.B.: $\{\underline{\texttt{0}}\underline{\texttt{1}}\} \in P$ aber $\{\underline{\texttt{1}}\} \notin P$. Daher ist P aufgrund des Satzes von Rice nicht entscheidbar.

f) **Nicht entscheidbar**, Satz von Rice: Die Eigenschaft $P = \{L \mid L, \overline{L} \text{ sind rekursiv aufzählbar}\}$ ist nicht trivial, denn es gilt z.B.: $\{\} \in P$ aber $L_u \notin P$. (L_u bezeichnet das Halteproblem, siehe z.B. Folie 49.) Daher ist P aufgrund des Satzes von Rice nicht entscheidbar.

Aufgabe 1.4 Sind folgende Aussagen korrekt? Begründen Sie jeweils Ihre Antwort.

a) Sei $L_1 = \{\underline{\texttt{a}}^n\underline{\texttt{b}}^n \mid n \geq 0\}$ und $L_2 = \{\underline{\texttt{b}}^{2n}\underline{\texttt{c}}^{2n} \mid n \geq 0\}$. Dann gilt: $L_1L_2 = \{\underline{\texttt{a}}^n\underline{\texttt{b}}^{3n}\underline{\texttt{c}}^{2n} \mid n \geq 0\}$.

b) Seien $L_1, ...L_k$ Sprachen über einem Alphabet Σ so, dass:
   - für alle $i \neq j$ gilt: $L_i \cap L_j = \{\}$
   - $L_1 \cup L_2 \cup ... \cup L_k = \Sigma^*$
   - jede dieser Sprachen $L_i, 1 \leq i \leq k$, rekursiv aufzählbar ist.

   Dann ist jede dieser Sprachen L_i rekursiv (entscheidbar).

c) Ist L rekursiv und F endlich, dann ist $L - F$ rekursiv.

d) Seien L_1 und L_2 entscheidbare Sprachen, und sei L eine Sprache so, dass $L_1 \subseteq L \subseteq L_2$. Dann ist auch L jedenfalls entscheidbar.

e) Es gibt unentscheidbare Sprachen über $\Sigma = \{\underline{\texttt{a}}\}$.

f) Ist L rekursiv aufzählbar, so ist jede Teilmenge von L rekursiv aufzählbar.

Lösung

a) **Falsch.** Das ist sicher nicht korrekt, da z.B. $\underline{\texttt{a}}^2\underline{\texttt{b}}^2 \in L_1$ und $\underline{\texttt{b}}^2\underline{\texttt{c}}^2 \in L_2$, aber $\underline{\texttt{a}}^2\underline{\texttt{b}}^4\underline{\texttt{c}}^2 \notin \{\underline{\texttt{a}}^n\underline{\texttt{b}}^{3n}\underline{\texttt{c}}^{2n} \mid n \geq 0\}$.

   (Richtig wäre in diesem Fall: $\{\underline{\texttt{a}}^n\underline{\texttt{b}}^n \mid n \geq 0\}\{\underline{\texttt{b}}^{2n}\underline{\texttt{c}}^{2n} \mid n \geq 0\} = \{\underline{\texttt{a}}^n\underline{\texttt{b}}^{n+2m}\underline{\texttt{c}}^{2m} \mid n, m \geq 0\}$)

b) **Richtig.** Für alle i gilt: $\overline{L_i} = L_1 \cup L_2 \cup ... \cup L_{i-1} \cup L_{i+1} \cup ... \cup L_k$. Jede der Sprachen $L_j, 1 \leq j \leq k$, ist rekursiv aufzählbar, wie auch deren Vereinigung. Nachdem also die Sprache L_i wie auch ihr Komplement rekursiv aufzählbar sind, ist L_i jedenfalls rekursiv (entscheidbar).

   (*Anmerkung*: Den Abschluss unter Vereinigung kann man z.B. so begründen: A (bzw. B) werde durch die Turingmaschine M_1 (bzw. M_2) erkannt. Um $A \cup B$ zu erkennen, entwerfen wir eine Turingmaschine M, die zuerst M_1 simuliert und dann M_2 simuliert. M wird genau dann akzeptieren, wenn eine der beiden Turingmaschinen akzeptiert, und nicht akzeptieren, wenn keine der beiden Maschinen akzeptiert.)

c) **Richtig.** Es gilt: $L - F = L \cap \overline{F}$. Endliche Sprachen sind regulär, und reguläre Sprachen sind entscheidbar. Nachdem rekursive Sprachen unter Komplement abgeschlossen sind, ist auch $\overline{F}$ rekursiv. Rekursive Sprachen sind auch unter Durchschnitt abgeschlossen ($A \cap B = \overline{\overline{A} \cup \overline{B}}$), also muss $L \cap \overline{F}$ bzw. $L - F$ ebenfalls rekursiv sein.

d) **Falsch.** Sei z.B.: $L_1 = \{\}$ und $L_2 = \Sigma^*$. Beide Sprachen sind entscheidbar, und es gilt: $L_1 \subseteq L_u \subseteq L_2$, wobei aber L_u (das Halteproblem) nicht entscheidbar ist.

e) **Richtig.** Σ^* ist abzählbar (unendlich), die Menge aller Sprachen $L \subseteq \Sigma^*$ ist überabzählbar. Von diesen ist jedoch nur eine abzählbare Menge entscheidbar. (Es gibt ja nur abzählbar (unendlich) viele Turingmaschinen). Dementsprechend gibt es unentscheidbare Sprachen über Σ.

f) **Falsch.** Σ^* ist rekursiv aufzählbar. Es gilt aber z.B. $L_u \subseteq \Sigma^*$. Das Halteproblem L_u ist rekursiv aufzählbar, aber nicht entscheidbar. Dementsprechend ist das Komplement des Halteproblems $\overline{L_u}$ nicht rekursiv aufzählbar. Es gilt jedoch auch $\overline{L_u} \subseteq \Sigma^*$ (nachdem $\overline{L_u} = \Sigma^* - L_u$).

Aufgabe 1.5 Sind folgende Sprachen regulär? Falls ja, so geben Sie einen entsprechenden deterministischen endlichen Automaten (oder regulären Ausdruck) an; falls nein, so beweisen Sie dies mit Hilfe des Pumping Lemmas für reguläre Sprachen.
(*Hinweis*: 2 der folgenden 4 Sprachen sind regulär.)

a) $\{v = xyzy \mid x, y, z \in \{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+\}$

b) $\{uww^r v \mid u, v, w \in \{\underline{\mathtt{a}}, \underline{\mathtt{b}}\}^+\}$

c) $\{opqp^r o \mid o, p, q \in \{\underline{\mathtt{a}}, \underline{\mathtt{b}}\}^+\}$

d) $\{\underline{\mathtt{a}}^n \underline{\mathtt{b}}^m \mid n, m \geq 0\} \cup \{w \in \{\underline{\mathtt{c}}, \underline{\mathtt{d}}\}^* \mid w = w^r\}$

Lösung

a) **Regulär**.

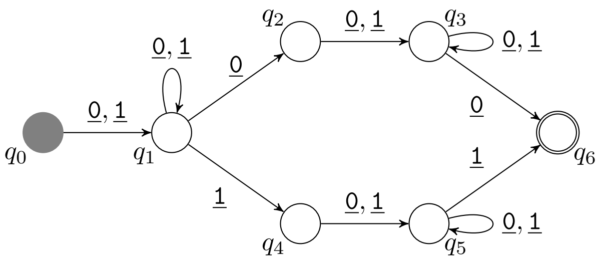
Wir überlegen zunächst: Jedes Wort $v \in \{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+$ ist in L wenn Folgendes gilt:

- v besteht aus mindestens 4 Symbolen.
- Das letzte Symbol von v kommt mindestens noch an einer anderen Stelle vor;
- diese ist nicht genau vor dem letzten Zeichen,
- und auch nicht das erste Symbol.

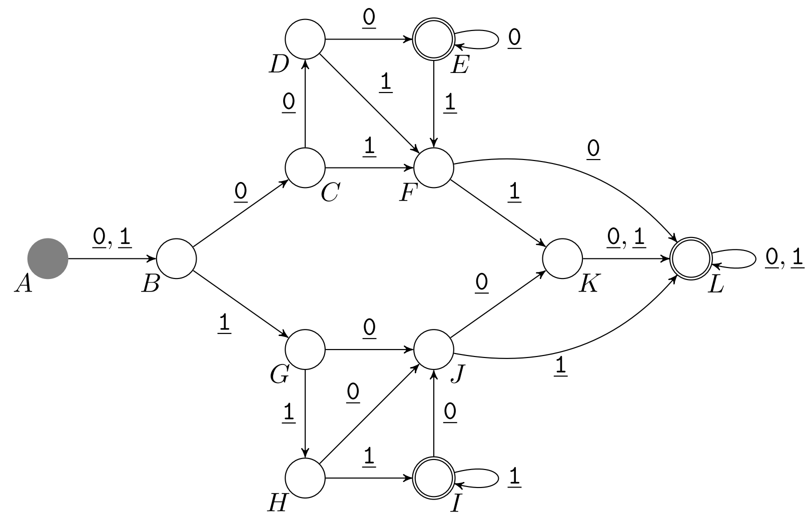
Das letzte Symbol kann entweder $\underline{\mathtt{0}}$ oder $\underline{\mathtt{1}}$ sein. Dementsprechend ergibt sich folgende reguläre Menge:

$L = \{\{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+ \{\underline{\mathtt{0}}\} \{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+ \{\underline{\mathtt{0}}\}\} \cup \{\{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+ \{\underline{\mathtt{1}}\} \{\underline{\mathtt{0}}, \underline{\mathtt{1}}\}^+ \{\underline{\mathtt{1}}\}\}$.

Hier zunächst ein NEA für L:



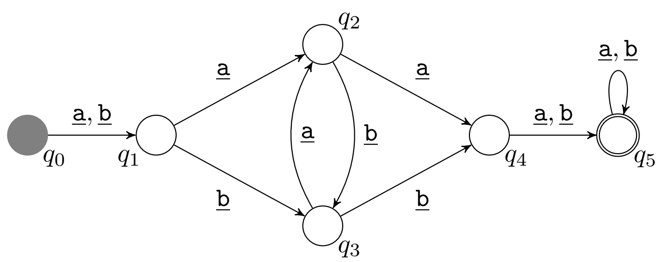
Determinisieren und Minimieren liefert folgenden DEA für L:



b) **Regulär**. Jedes Wort in L besteht aus mindestens 4 Symbolen. Sei $|w| = 1$, dann besteht das Teilwort ww^r einfach aus zwei gleichen Symbolen hintereinander. L enthält also Wörter, die aus mindestens 4 Symbolen bestehen, wobei ein Symbol zweimal hintereinander vorkommt, allerdings muss davor und danach mindestens je ein Zeichen sein.

Also ergibt sich: $L = \{\underline{\mathtt{a}}, \underline{\mathtt{b}}\}^+ \{\underline{\mathtt{a}}\underline{\mathtt{a}}, \underline{\mathtt{b}}\underline{\mathtt{b}}\}\{\underline{\mathtt{a}}, \underline{\mathtt{b}}\}^+$.

DEA für L:



c) **Nicht regulär**.

Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und das gewählte Wort z.B.

$$w = \underline{\mathtt{a}}^m \underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{a}}^m \underline{\mathtt{b}}$$

Dann gilt $w \in L$ und $|w| = 2m + 5 > m$.

Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = \underline{\mathtt{a}}^m \underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{a}}^m \underline{\mathtt{b}}$, kann xy nur aus Symbolen $\underline{\mathtt{a}}$ bestehen.

Nach dem Pumping Lemma muss aber $xy^iz \in L$ für alle $i \geq 0$ gelten.

Wenn wir nun z.B. $i = 2$ wählen, müsste auch $xy^2z = \underline{\mathtt{a}}^{m+|y|}\underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{b}}\underline{\mathtt{a}}\underline{\mathtt{a}}^m \underline{\mathtt{b}}$ aus L sein, was aber nicht der Fall ist, da nun der erste und der letzte Wortteil nicht mehr gleich sind. Wir haben also einen Widerspruch gefunden, L kann somit keine reguläre Sprache sein.

(*Anmerkung:* Beachten Sie, dass hier eine Wortwahl wie etwa $w = \underline{\mathtt{a}}^m\underline{\mathtt{b}}^m\underline{\mathtt{a}}^m\underline{\mathtt{b}}^m\underline{\mathtt{a}}^m$ nicht zielführend ist. Denn wählen Sie z.B. $i = 2$ so erhalten Sie die aufgepumpte Version $w_2 = \underline{\mathtt{a}}^{m+|y|}\underline{\mathtt{b}}^m\underline{\mathtt{a}}^m\underline{\mathtt{b}}^m\underline{\mathtt{a}}^m$. Diese Wörter sind jedoch wiederum in L, denn sie können nach der Sprachbeschreibung z.B. so zerlegt werden: $o = \underline{\mathtt{a}}, p = \underline{\mathtt{a}}^{m-1}, q = \underline{\mathtt{a}}^{|y|}\underline{\mathtt{b}}^m\underline{\mathtt{a}}^m\underline{\mathtt{b}}^m$. Damit wird also kein Widerspruch erreicht.)

d) **Nicht regulär**.

Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und das gewählte Wort z.B.

$$w = \underline{\mathtt{c}}^m \underline{\mathtt{d}}\underline{\mathtt{c}}^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 1 > m$.

Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = \underline{\mathtt{c}}^m\underline{\mathtt{d}}\underline{\mathtt{c}}^m$, kann xy nur aus Symbolen $\underline{\mathtt{c}}$ bestehen.

Nach dem Pumping Lemma muss aber $xy^iz \in L$ für alle $i \geq 0$ gelten.

Wenn wir nun z.B. $i = 3$ wählen, müsste auch $xy^3z = \underline{\mathtt{c}}^{m+2*|y|}\underline{\mathtt{d}}\underline{\mathtt{c}}^m$ aus L sein, was aber nicht der Fall ist! Denn da y aus mindestens einem Symbol besteht, sind nun in jeder Zerlegung mehr Symbole $\underline{\mathtt{c}}$ am Anfang des Wortes als am Ende; demnach gilt $\underline{\mathtt{c}}^{m+2*|y|}\underline{\mathtt{d}}\underline{\mathtt{c}}^m \notin L$. Wir haben also einen Widerspruch gefunden, L kann somit keine reguläre Sprache sein.

(*Anmerkung*: Hier ein Wort aus der Menge $\{\underline{\mathtt{a}}^n\underline{\mathtt{b}}^m \mid n, m \geq 0\}$ zu wählen ist nicht zielführend, da damit kein Widerspruch erzielt werden kann.)