Bemerkung zur Serie 2:

4b) Achtung! $2^{n^2} = 2^{(n^2)} \neq (2^n)^2 = 2^{2n}$

$|w_n| = 2^{n^2} \iff n = \sqrt{\log_2(|w_n|)}$

$|x_n| = 2^{2n} \iff n = \log_2(\sqrt{|x_n|})$

6a) Lest die Musterlösung durch!

3.2 Die Darstellung von endlichen Automaten

Def: Ein endlicher Automat (EA) ist ein Quintupel $M = (\Omega, \Sigma, \delta, q_0, F)$ wobei

(i) Q ist eine endliche Menge von Zuständen

(ii) Σ ist ein Alphabet, das sogenannte Eingabealphabet

(iii) $q_0 \in Q$ ist der Anfangszustand

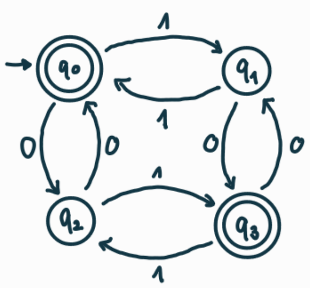
(iv) $F \subseteq Q$ ist die Menge der akzeptierenden Zustände

(v) $\delta : Q \times \Sigma \to Q$ ist eine Fkt., die sogenannte Übergangsfunktion

Bmk: δ ist nur für Buchstaben definiert. Für Wörter $w \in \Sigma^+$ mit $w = xa$, $x \in \Sigma^*$ haben wir

$\hat{\delta} : Q \times \Sigma^* \to Q$; $\quad \hat{\delta}(q, w) = \hat{\delta}(q, xa) = \delta(\underbrace{\hat{\delta}(q, w)}_{q' \in Q}, a)$

Bsp:



$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{0, 1\}$

$F = \{q_0, q_3\}$

Interpretation als Code:

```
0: if input = 1 then goto 1 else goto 2
1: if input = 1 then goto 0 else goto 3
2: if input = 1 then goto 3 else goto 0
3: if input = 1 then goto 2 else goto 1
```

Zeilen

$F = \{0, 3\}$

δ definiert durch Code

Bmk: Jeder Knoten (Zustand) braucht so viele ausgehende Kanten, wie es Buchstaben im Eingabealphabet gibt.

Def: Eine Konfiguration von M ist ein Element in $Q \times \Sigma^*$.
Die Startkonfiguration von M auf $x \in \Sigma^*$ ist $(q_0, x) \in \{q_0\} \times \Sigma^*$.
Eine Konfiguration in $Q \times \{\lambda\}$ heisst Endkonfiguration.

Bsp: Wir betrachten den oben abgebildeten EA und das Wort $x = 101 \in (\Sigma_{bool})^*$.
Konfigurationen: $\{(q_0, 101), (q_1, 01), (q_3, 1), (q_2, \lambda)\}$
Startkonfiguration: $(q_0, 101)$
Endkonfiguration: (q_2, λ)

Def: Ein Schritt $\vdash_M$ von M ist eine Relation auf Konfigurationen
$\vdash_M \subseteq (Q \times \Sigma^*) \times (Q \times \Sigma^*)$ definiert durch
$(q, w) \vdash_M (p, x) \iff w = ax,\ a \in \Sigma$ und $\delta(q, a) = p$

Bsp: Fortsetzung des obigen Beispiels
$(q_1, 01) \vdash_M (q_3, 1)$
$(q_0, 101) \not\vdash_M (q_2, 01)$

Bmk: $\vdash_M^*$ ist die reflexive und transitive Hülle der Schrittrelation

kein Schritt

mehrere Schritte

Def: • Eine Berechnung C von M ist eine endliche Folge $C = C_0, C_1, \ldots, C_n$ von Konfigurationen, s.d. $\forall\, 0 \le i \le n-1 : C_i \vdash_M C_{i+1}$

• C ist die Berechnung von M auf einer Eingabe $x \in \Sigma^*$, falls $C_0 = (q_0, x)$ und $C_n \in Q \times \{\lambda\}$ eine Endkonfiguration ist.

• C ist eine akzeptierende Berechnung von M auf x, falls $C_n \in F \times \{\lambda\}$.

• C ist eine verwerfende Berechnung von M auf x, falls $C_n \in (Q - F) \times \{\lambda\}$.

Def: Die <u>von M akzeptierte Sprache</u> $L(M)$ ist definiert als:

$L(M) := \{ w \in \Sigma^* \mid$ die Berechnung von M auf w endet in einer Endkonfiguration (q, λ) mit $q \in F\}$

Def: Eine Sprache L heisst <u>regulär</u>, falls es einen EA M gibt, der L akzeptiert.

Def: $\mathcal{L}_{EA} := \{ L(M) : M \text{ ist ein EA}\}$ ist die Klasse der regulären Sprachen.

Aufg 1) Konstruiere (deterministische) endliche Automaten, die folgende Sprachen erkennen und definiere ihre Zustandsklassen.

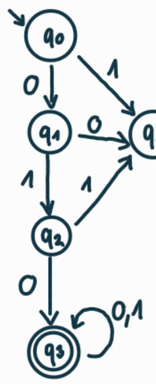
a) $L_1 = \{ w \in \{0,1\}^* : w = 010x,\ x \in \{0,1\}^* \}$

b) $L_2 = \{ w \in \{0,1\}^* : w = x010,\ x \in \{0,1\}^* \}$

c) $L_3 = \{ w \in \{0,1\}^* : w = xy,\ x \in \{0,1\}^+,\ 2 \mid \text{Nummer}(x),\ y \in \{0,1\}^* \}$

Bonus: d) $L_4 = \{ w \in \{0,1\}^* : w = 010x,\ x \in \{0,1\}^*,\ (|w|_0 - |w|_1) \bmod 3 = 2\}$

Lösung: a)



$Kl[q_0] = \{\lambda\}$

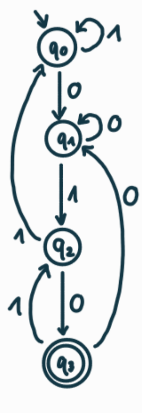
$Kl[q_1] = \{0\}$

$Kl[q_2] = \{01\}$

$Kl[q_3] = L_1 = \{ w \in \{0,1\}^* : w = 010x,\ x \in \{0,1\}^* \}$

$Kl[q_4] = \{0,1\}^* - \bigcup_{i=0}^{3} Kl[q_i]$

b)



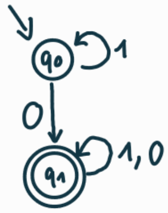
$Kl[q_3] = L_2 = \{ w \in \{0,1\}^* : w = x010,\ x \in \{0,1\}^* \}$

$Kl[q_2] = \{ w \in \{0,1\}^* : w = x01,\ x \in \{0,1\}^* \}$

$Kl[q_1] = \{ w \in \{0,1\}^* : w \text{ endet mit } 0\} - Kl[q_3]$

$Kl[q_0] = \{\lambda\} \cup \{ w \in \{0,1\}^* : w \text{ endet mit } 1\} - Kl[q_2]$

c) Interpretation: $L_3 = \{ w \in \{0,1\}^* : |w|_0 \geq 1 \}$

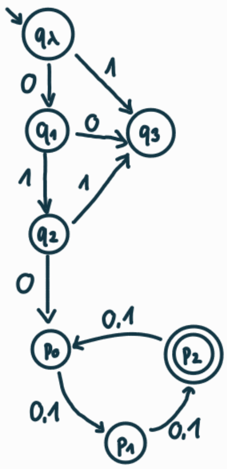


Dann endet x mit der ersten 0 in w, wodurch $x \neq \{\lambda\}$ und $2 \mid \text{Nummer}(x)$ erfüllt sind.

$Kl[q_0] = \{ 1^n : n \in \mathbb{N}_0 \}$

$Kl[q_1] = L_3$

d)



$Kl[q_0] = \{\lambda\}$

$Kl[q_1] = \{0\}$

$Kl[q_2] = \{01\}$

$Kl[q_3] = L_1^c = \{ w \in \{0,1\}^* : w = 010x \quad x \in \{0,1\}^* \}^C$

$Kl[p_0] = \{ w \in L_1 : (|w|_0 - 2|w|_1) \bmod 3 = 0 \}$

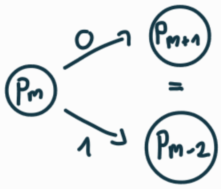
$Kl[p_1] = \{ w \in L_1 : (|w|_0 - 2|w|_1) \bmod 3 = 1 \}$

$Kl[p_2] = \{ w \in L_1 : (|w|_0 - 2|w|_1) \bmod 3 = 2 \} = L_4$

Bmk: $w = 010x$.

$(|w|_0 - 2|w|_1) \bmod 3 = (|x|_0 - 2|x|_1) \bmod 3$

Sei $m = (|w|_0 - 2|w|_1) \bmod 3$.



da $-2 \pmod 3 = 1$

3.3 Simulation

Lem 3.2: Sei Σ ein Alphabet.
Seien $M_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ und $M_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$ zwei EA.
Für jede Mengenoperation $\odot \in \{ \cup, \cap, - \}$ existiert ein EA M, s.d. $L(M) = L(M_1) \odot L(M_2)$

Bmk: Die Methode des modularen Entwurfs (Konstruktion eines Produktautomaten) dient dazu, ein EA M mit $L(M) = L(M_1) \odot L(M_2)$ aus M_1 und M_2 zu konstruieren.

Bmk: Die Art der Operation verändert lediglich die Menge der akzeptierenden Zustände F von $M = (Q, \Sigma, \delta, q_0, F)$.

$\odot = \cup \Rightarrow F = F_1 \times Q_2 \cup Q_1 \times F_2$

$\odot = \cap \Rightarrow F = F_1 \times F_2$

$\odot = - \Rightarrow F = F_1 \times (Q_2 - F_2)$

Bmk: Verwendet bei dieser Konstruktion (bitte auch in der Serie) verschiedene Farben.

Aufg 2: Wir definieren folgende Sprachen über $\Sigma = \{a,b\}$.

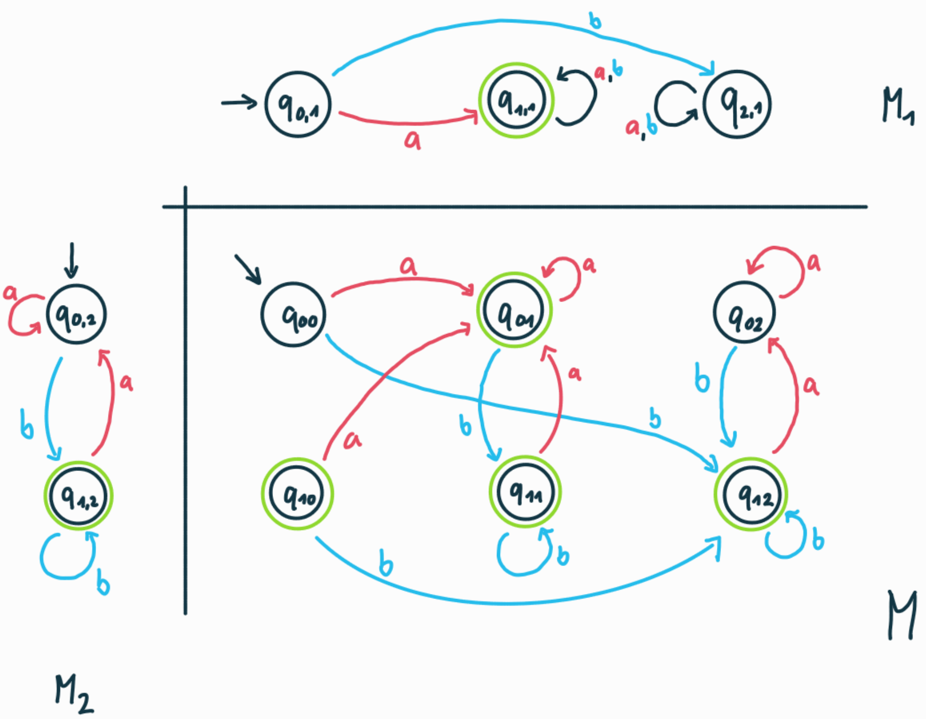
$L_1 := \{ w \in \{a,b\}^* : w \text{ beginnt mit } a \}$

$L_2 := \{ w \in \{a,b\}^* : w \text{ endet mit } b \}$

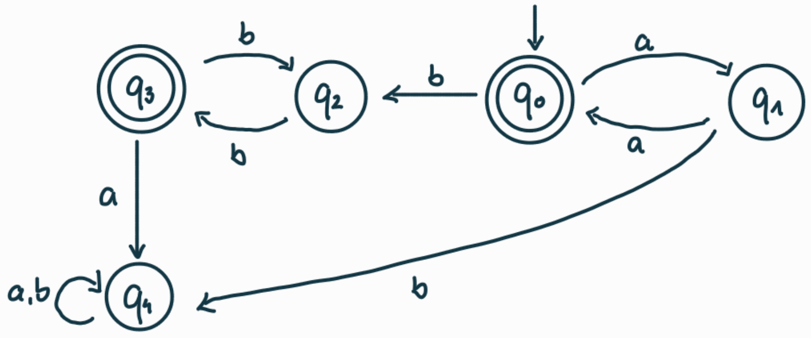
$L := L_1 \cup L_2 = \{ w \in \{a,b\}^* : w \text{ beginnt mit } a \text{ oder endet mit } b \}$.

Konstuiere einen EA M s.d. $L = L(M)$ mit der Methode des modularen Entwurfs.

Lösung:



Aufg 3: Bestimme $L(M)$ für folgenden endlichen Automaten M.



Lösung: $L(M) = \{ w \in \{a,b\} : w = xy \, , \, x \in \{aa\}^* \wedge y \in \{bb\}^* \}$

Begründung:
- $\lambda \in L(M)$ da $q_0 \in F$.
- Die akzeptierenden Knoten können nur über eine gerade Anzahl von aufeinanderfolgenden a's und b's erreicht werden.
- q_4 ist ein "Trash"-Zustand, da $q_4 \notin F$ und kein Pfeil von q_4 weg führt. Wir erreichen q_4, falls auf ein b ein a folgt oder falls auf eine ungerade Anzahl von a's ein b folgt.
- Falls die Berechnung bei q_1 oder q_2 hält, ist $|w|_a$ oder $|w|_b$ ungerade.

Im Allgemeinen sind das Definieren der Zustandsklassen eine gute Begründung für solche Aufgaben.