GBI Tutorium Nr. 41

Foliensatz 10

Vincent Hahn – vincent.hahn@student.kit.edu | 10. Januar 2013

Outline/Gliederung

Vincent Hahn – vincent.hahn@student.kit.edu

Überblick

Vincent Hahn – vincent.hahn@student.kit.edu

Wiederholung - Quiz

Vincent Hahn – vincent.hahn@student.kit.edu

- Aus $f \in \Omega(g) \wedge f \in \Theta(g) \Rightarrow f \in \mathcal{O}(g)$
- $n^5 \in \mathcal{O}(2^n)$
- $\frac{n^3+2n}{2n+1} \in \mathcal{O}(n)$
- Alle Algorithmen liegen in $\Omega(1)$

Wiederholung - Quiz

Vincent Hahn – vincent.hahn@student.kit.edu

- Aus $f \in \Omega(g) \wedge f \in \Theta(g) \Rightarrow f \in \mathcal{O}(g)$ $\checkmark$
- $n^5 \in \mathcal{O}(2^n)$
- $\frac{n^3+2n}{2n+1} \in \mathcal{O}(n)$
- Alle Algorithmen liegen in $\Omega(1)$

Wiederholung - Quiz

Vincent Hahn – vincent.hahn@student.kit.edu

- Aus $f \in \Omega(g) \wedge f \in \Theta(g) \Rightarrow f \in \mathcal{O}(g)$ $\surd$
- $n^5 \in \mathcal{O}(2^n)$ $\surd$
- $\frac{n^3+2n}{2n+1} \in \mathcal{O}(n)$
- Alle Algorithmen liegen in $\Omega(1)$

Wiederholung - Quiz

Vincent Hahn – vincent.hahn@student.kit.edu

- Aus $f \in \Omega(g) \wedge f \in \Theta(g) \Rightarrow f \in \mathcal{O}(g)$ $\surd$
- $n^5 \in \mathcal{O}(2^n)$ $\surd$
- $\frac{n^3+2n}{2n+1} \in \mathcal{O}(n)$ X
- Alle Algorithmen liegen in $\Omega(1)$

Wiederholung - Quiz

Vincent Hahn – vincent.hahn@student.kit.edu

- Aus $f \in \Omega(g) \land f \in \Theta(g) \Rightarrow f \in \mathcal{O}(g)$ $\surd$
- $n^5 \in \mathcal{O}(2^n)$ $\surd$
- $\frac{n^3+2n}{2n+1} \in \mathcal{O}(n)$ X
- Alle Algorithmen liegen in $\Omega(1)$ $\surd$

Überblick

Vincent Hahn – vincent.hahn@student.kit.edu

Master-Theorem

Karlsruher Institut für Technologie

Definition

Für einen **rekursiven** Algorithmus der Form

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

kann die Laufzeit für drei Fälle abgeschätzt werden:

1. Wenn $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$ für ein $\varepsilon > 0$, dann ist $T(n) \in \Theta\left(n^{\log_b a}\right)$
2. Wenn $f(n) \in \Theta\left(n^{\log_b a}\right)$, dann ist $T(n) \in \Theta\left(n^{\log_b a} \log n\right)$
3. Wenn $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$ für ein $\varepsilon > 0$,
   und wenn es eine Konstante d gibt mit $0 < d < 1$,
   sodass für alle hinreichend großen n gilt $af(n/b) \leq df(n)$,
   dann ist $T(n) \in \Theta(f(n))$

- Fall 2 wird etwa bei Quicksort benötigt
- Fall 3 ist eher die Ausnahme

Master-Theorem

Vincent Hahn – vincent.hahn@student.kit.edu

Definition

Für einen **rekursiven** Algorithmus der Form

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

kann die Laufzeit für drei Fälle abgeschätzt werden:

1. Wenn $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$ für ein $\varepsilon > 0$, dann ist $T(n) \in \Theta\left(n^{\log_b a}\right)$
2. Wenn $f(n) \in \Theta\left(n^{\log_b a}\right)$, dann ist $T(n) \in \Theta\left(n^{\log_b a} \log n\right)$
3. Wenn $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$ für ein $\varepsilon > 0$,
   und wenn es eine Konstante d gibt mit $0 < d < 1$,
   sodass für alle hinreichend großen n gilt $af\left(n/b\right) \leq df(n)$,
   dann ist $T(n) \in \Theta(f(n))$

- Fall 2 wird etwa bei Quicksort benötigt
- Fall 3 ist eher die Ausnahme

Master-Theorem

Vincent Hahn – vincent.hahn@student.kit.edu

Definition

Für einen **rekursiven** Algorithmus der Form

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

kann die Laufzeit für drei Fälle abgeschätzt werden:

1. Wenn $f(n) \in \mathcal{O}\left(n^{\log_b a-\varepsilon}\right)$ für ein $\varepsilon > 0$, dann ist $T(n) \in \Theta\left(n^{\log_b a}\right)$
2. Wenn $f(n) \in \Theta\left(n^{\log_b a}\right)$, dann ist $T(n) \in \Theta\left(n^{\log_b a} \log n\right)$
3. Wenn $f(n) \in \Omega\left(n^{\log_b a+\varepsilon}\right)$ für ein $\varepsilon > 0$, und wenn es eine Konstante d gibt mit $0 < d < 1$, sodass für alle hinreichend großen n gilt $af\left({}^{n}/_{b}\right) \leq df(n)$, dann ist $T(n) \in \Theta(f(n))$

- Fall 2 wird etwa bei Quicksort benötigt
- Fall 3 ist eher die Ausnahme

Vincent Hahn – vincent.hahn@student.kit.edu

Master-Theorem

Definition

Für einen **rekursiven** Algorithmus der Form

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

kann die Laufzeit für drei Fälle abgeschätzt werden:

1. Wenn $f(n) \in \mathcal{O}\left(n^{\log_b a-\varepsilon}\right)$ für ein $\varepsilon > 0$, dann ist $T(n) \in \Theta\left(n^{\log_b a}\right)$
2. Wenn $f(n) \in \Theta\left(n^{\log_b a}\right)$, dann ist $T(n) \in \Theta\left(n^{\log_b a} \log n\right)$
3. Wenn $f(n) \in \Omega\left(n^{\log_b a+\varepsilon}\right)$ für ein $\varepsilon > 0$,
   und wenn es eine Konstante d gibt mit $0 < d < 1$,
   sodass für alle hinreichend großen n gilt $af\left(n/b\right) \leq df(n)$,
   dann ist $T(n) \in \Theta(f(n))$

- Fall 2 wird etwa bei Quicksort benötigt
- Fall 3 ist eher die Ausnahme

Master Theorem

Vincent Hahn – vincent.hahn@student.kit.edu

Beispiele

$$49 \cdot T(\frac{n}{7}) + 3n + 5$$

$$49 \cdot T(\frac{n}{7}) + 3n^3 + 5$$

Überblick

Vincent Hahn – vincent.hahn@student.kit.edu

Mealy-Automat

Vincent Hahn – vincent.hahn@student.kit.edu

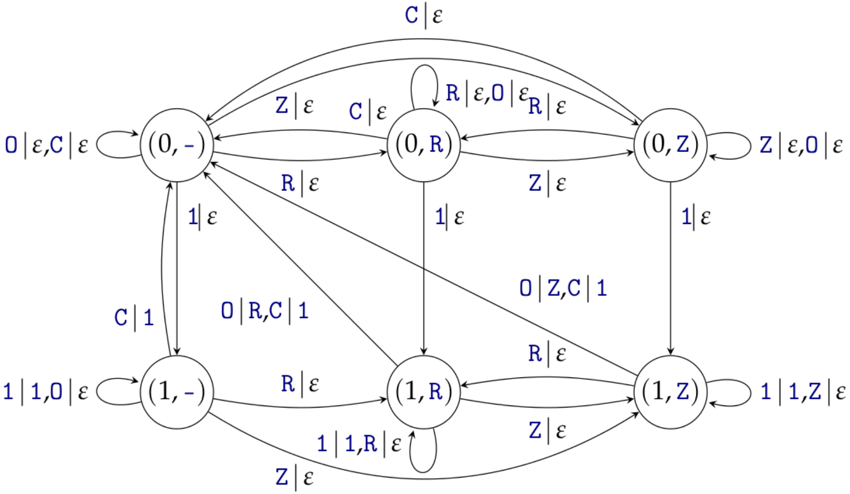
Definition: Mealy-Automat

Der Mealy-Automat $A = (Z, z_0, X, f, Y, g)$ besteht aus

1. der endlichen Zustandsmenge $\mathbf{Z}$,
2. dem Startzustand $\mathbf{z_0}$,
3. dem Eingabealphabet $\mathbf{X}$,
4. der Zustandsübergangsfunktion $\mathbf{f} : \mathbf{Z} \times \mathbf{X} \to \mathbf{Z}$,
5. einem Ausgabealphabet $\mathbf{Y}$ und
6. der Ausgabefunktion $\mathbf{g} : \mathbf{Z} \times \mathbf{X} \to \mathbf{Y}^*$.

Getränkeautomat

Vincent Hahn – vincent.hahn@student.kit.edu



Getränkeautomat

Was ist was?

- Zustandsmenge Z:
- Eingabealphabet X:
- Zustandsübergangsfunktion f:
- Ausgabealphabet Y:
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Getränkeautomat

Vincent Hahn – vincent.hahn@student.kit.edu

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X:
- Zustandsübergangsfunktion f:
- Ausgabealphabet Y:
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Getränkeautomat

Vincent Hahn – vincent.hahn@student.kit.edu

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X: $\{1,R,Z,C,0\}$
- Zustandsübergangsfunktion f:
- Ausgabealphabet Y:
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Getränkeautomat

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X: $\{1, R, Z, C, 0\}$
- Zustandsübergangsfunktion f: die Pfeile
- Ausgabealphabet Y:
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Getränkeautomat

Karlsruher Institut für Technologie

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X: $\{1,R,Z,C,0\}$
- Zustandsübergangsfunktion f: die Pfeile
- Ausgabealphabet Y: $\{1,R,Z\}$
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Vincent Hahn – vincent.hahn@student.kit.edu

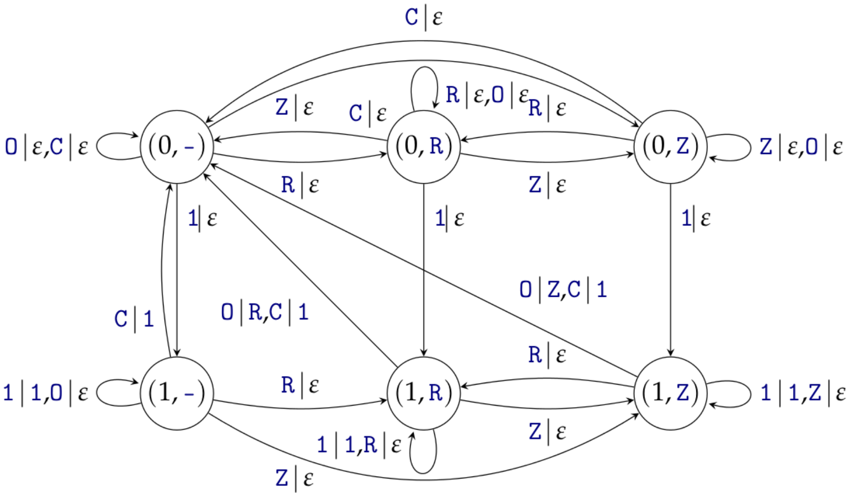
Getränkeautomat

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X: $\{1,R,Z,C,0\}$
- Zustandsübergangsfunktion f: die Pfeile
- Ausgabealphabet Y: $\{1,R,Z\}$
- Ausgabefunktion g: bisher noch nicht eingezeichnet, siehe nächste Folie

Getränkeautomat (mit Ausgabe)

Vincent Hahn – vincent.hahn@student.kit.edu



Vincent Hahn – vincent.hahn@student.kit.edu

Getränkeautomat

Was ist was?

- Zustandsmenge Z: $\{(0,-),(0,R),(0,Z),(1,-),(1,R),(1,Z)\}$
- Eingabealphabet X: $\{1,R,Z,C,0\}$
- Zustandsübergangsfunktion f: die Pfeile, was vor einem senkrechten Strich $|$ steht
- Ausgabealphabet Y: $\{1,R,Z\}$
- Ausgabefunktion g: die Pfeile, was hinter einem senkrechten Strich $|$ steht

$f*$ und $f**$

Definition: $f*$ und $f**$

$f^* = f^*(z, w)$ kann im Gegensatz zu f ein ganzes Wort w als zweites Funktionsargument nehmen:

$$f^* : Z \times X^* \to Z$$
$$f^*(z, \varepsilon) = z$$
$$f^*(z, wx) = f(f^*(z, w), x)$$

f^{**} kann im Gegensatz zu f^* ganze Wörter anstatt einem Symbol ausgeben:

$$f^* : Z \times X^* \to Z^*$$
$$f^{**}(z, \varepsilon) = z$$

$f*$ und $f**$

Definition: $f*$ und $f**$

$f^* = f^*(z, w)$ kann im Gegensatz zu f ein ganzes Wort w als zweites Funktionsargument nehmen:

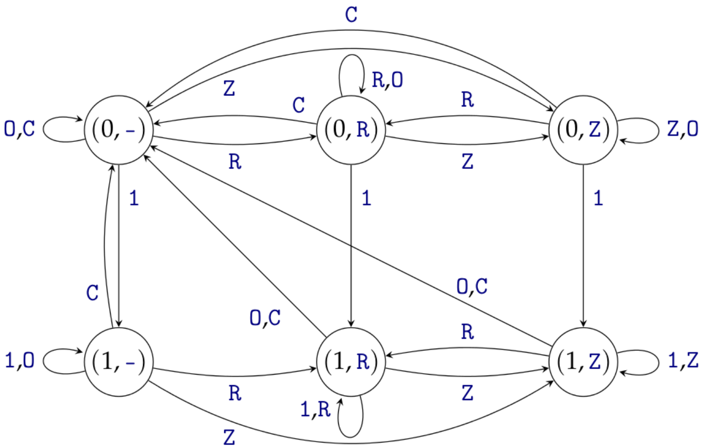
$$f^* : Z \times X^* \to Z$$
$$f^*(z, \varepsilon) = z$$
$$f^*(z, wx) = f(f^*(z, w), x)$$

f^{**} kann im Gegensatz zu f^* ganze Wörter anstatt einem Symbol ausgeben:

$$f^* : Z \times X^* \to Z^*$$
$$f^{**}(z, \varepsilon) = z$$
$$f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

Getränkeautomat (ohne Ausgabe)

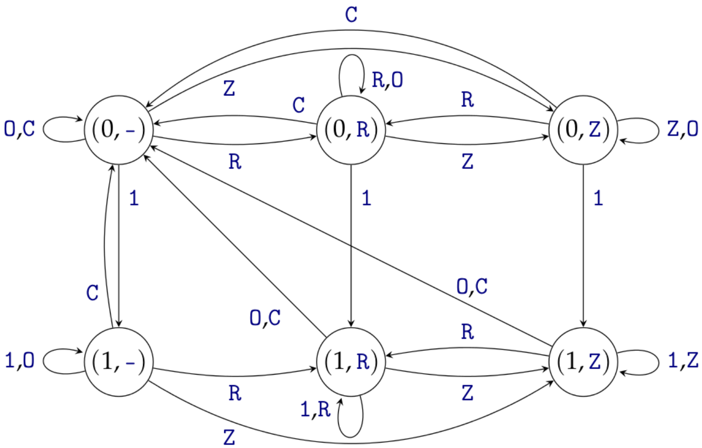
Vincent Hahn – vincent.hahn@student.kit.edu



Was macht $f^*\left((0,-), R10\right)$?

Getränkeautomat (ohne Ausgabe)

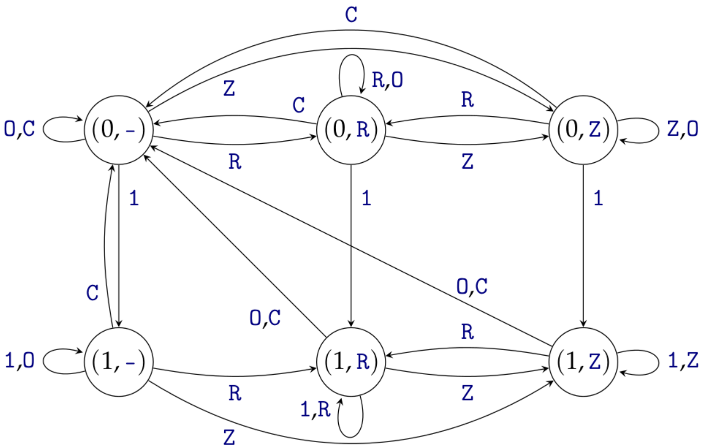
Vincent Hahn – vincent.hahn@student.kit.edu



Was macht $f^*((0,-), R10)$? Berechnet $f^*((0,-), R10)$.

Getränkeautomat (ohne Ausgabe)

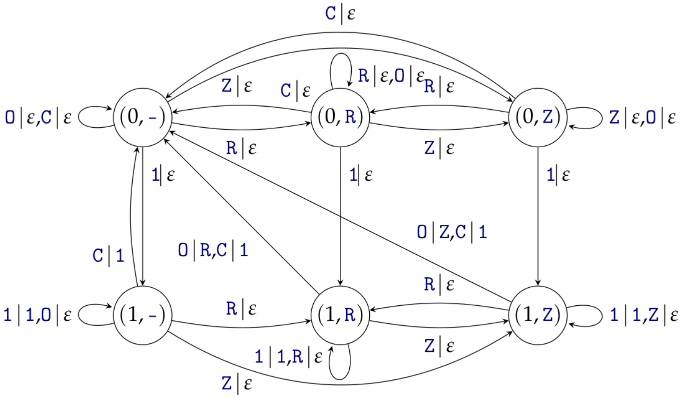
Vincent Hahn – vincent.hahn@student.kit.edu



Was macht $f^*\left((0,-), R10\right)$? Berechnet $f^*\left((0,-), R10\right)$. Was käme bei $f^{**}\left((0,-), R10\right)$ raus?

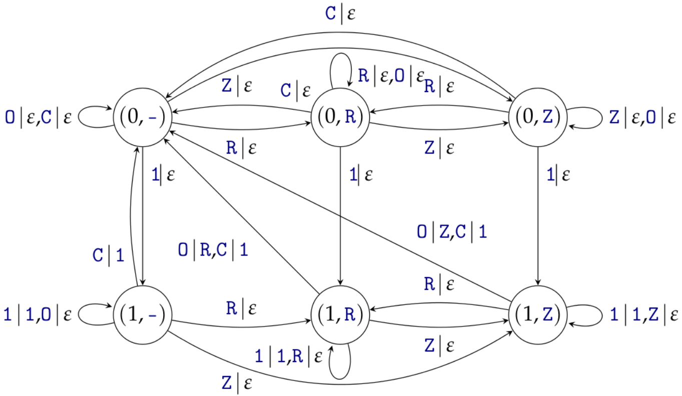
Getränkeautomat (mit Ausgabe)

Vincent Hahn – vincent.hahn@student.kit.edu



Was macht $g^* \left((0, -) \, , R10 \right)$?

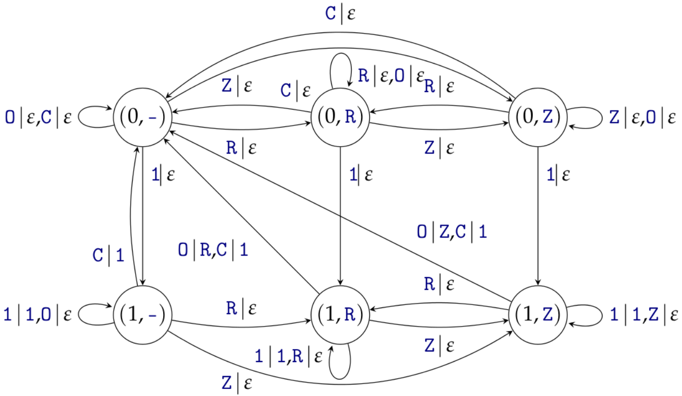
Getränkeautomat (mit Ausgabe)



Was macht $g^*\left((0,-), R10\right)$? Berechnet $g^*\left((0,-), R10\right)$.

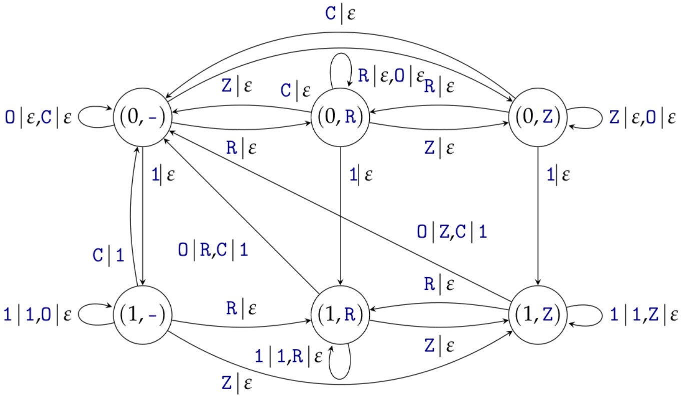
Getränkeautomat (mit Ausgabe)

Vincent Hahn – vincent.hahn@student.kit.edu



Was macht $g^*((0,-), R10)$? Berechnet $g^*((0,-), R10)$. Was käme bei $g^{**}((0,-), R10)$ raus?

Getränkeautomat (mit Ausgabe)



Was macht $g^*((0,-),R10)$? Berechnet $g^*((0,-),R10)$. Was käme bei $g^{**}((0,-),R10)$ raus? Was passiert bei $g^{**}((0,-),R110) = 1R$

Alternativer Automat

Vincent Hahn – vincent.hahn@student.kit.edu

Gegeben sei der Automat mit

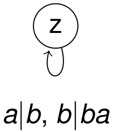
- $Z = \{z\}$,
- $X = Y = \{a, b\}$,
- $g(z, a) = b$,
- $g(z, b) = ba$.

1. Zeichnet den Automaten,
2. gebt $w_1 = g^{**}(z, a)$ an und
3. gebt $w_2 = g^{**}(z, w_1)$ an.

Wie sieht w_3 vermutlich aus? Allgemein, wie sieht w_i aus?

Alternativer Automat

Vincent Hahn – vincent.hahn@student.kit.edu



- $w_1 = g^{**}(z, a) = g^{**}(z, \varepsilon) \cdot g^{*}(z, wx) = g(f^{*}(z, \varepsilon), a) = g(z, a) = b$
- $w_2 = \cdots = ba$

Überblick

Vincent Hahn – vincent.hahn@student.kit.edu

Moore-Automaten

Vincent Hahn – vincent.hahn@student.kit.edu

Definition: Mealy-Automat

Der Mealy-Automat $A = (Z, z_0, X, f, Y, h)$ besteht aus

Bis auf die Asgabefunktion sind Mealy- und Moore-Automat identisch. Der Moore-Automat hat seine Ausgabe in einem Zustand.

Moore-Automaten

Vincent Hahn – vincent.hahn@student.kit.edu

Definition: Mealy-Automat

Der Mealy-Automat $A = (Z, z_0, X, f, Y, h)$ besteht aus

1. der endlichen Zustandsmenge $\mathbf{Z}$,
2. dem Startzustand $\mathbf{z_0}$,
3. dem Eingabealphabet $\mathbf{X}$,
4. der Zustandsübergangsfunktion $\mathbf{f} : \mathbf{Z} \times \mathbf{X} \to \mathbf{Z}$,
5. einem Ausgabealphabet $\mathbf{Y}$ und
6. der Ausgabefunktion $\mathbf{h} : \mathbf{Z} \to \mathbf{Y}^*$.

Bis auf die Asgabefunktion sind Mealy- und Moore-Automat identisch. Der Moore-Automat hat seine Ausgabe in einem Zustand.

Überblick

Vincent Hahn – vincent.hahn@student.kit.edu

Endlicher Akzeptor

Vincent Hahn – vincent.hahn@student.kit.edu

Definition: Endlicher Akzeptor

Ein endlicher Akzepter ist ein spezieller Moore-Automat, der

- 1 ausgibt, wenn ein Wort einer Wortbildungsregel (Syntax) entspricht und
- 0 ansonsten ausgibt.

Im Gegensatz zum gewöhnlichen Moore Automat besitzt er

- **keine Ausgabefunktion** h,
- dafür eine **eine Menge** $F \subseteq Z$ **akzeptierender Zustände**.

$$A = (Z, z_0, X, f, F)$$

Die akzeptierenden Zustände werden doppelt umrahmt.

Endlicher Akzeptor

Vincent Hahn – vincent.hahn@student.kit.edu

Definition: Endlicher Akzeptor

Ein endlicher Akzepter ist ein spezieller Moore-Automat, der

- 1 ausgibt, wenn ein Wort einer Wortbildungsregel (Syntax) entspricht und
- 0 ansonsten ausgibt.

Im Gegensatz zum gewöhnlichen Moore Automat besitzt er

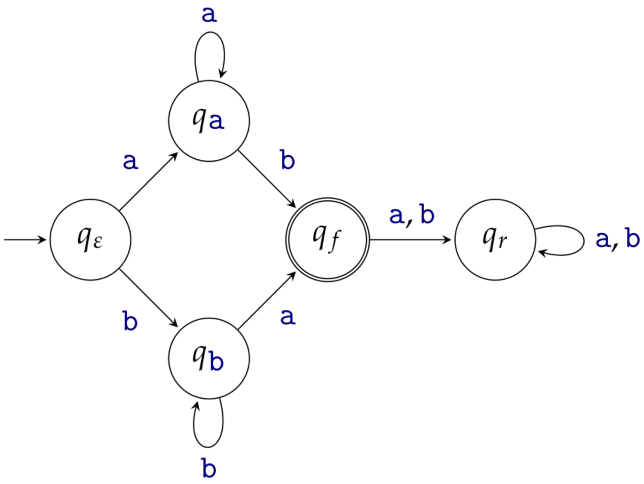
- **keine Ausgabefunktion** h,
- dafür eine **eine Menge** $F \subseteq Z$ **akzeptierender Zustände**.

$$A = (Z, z_0, X, f, F)$$

Die akzeptierenden Zustände werden doppelt umrahmt.

Endlicher Akzeptor: Beispiel

Vincent Hahn – vincent.hahn@student.kit.edu



Nennt Wörter und sagt, ob diese akzeptiert werden oder nicht.

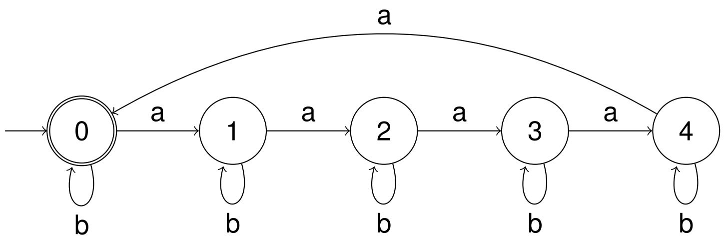
Endlicher Akzeptor: Aufgabe

Vincent Hahn – vincent.hahn@student.kit.edu

- Gesucht ist ein kleiner endlicher Akzeptor, der alle Wörter akzeptiert, bei denen die Anzahl der *a* durch 5 teilbar ist.
  Gegeben: $X = \{a, b\}$.
- Gesucht ist ein endlicher Akzeptor, in dem nirgends hintereinander zwei *b* vorkommen.

Lösung 1

Vincent Hahn – vincent.hahn@student.kit.edu



Lösung 2

Vincent Hahn – vincent.hahn@student.kit.edu