F2 — Automaten und formale Sprachen

Berndt Farwer

Fachbereich Informatik

AB „Theoretische Grundlagen der Informatik“ (TGI)

Universität Hamburg

farwer@informatik.uni-hamburg.de

Themen der heutigen Vorlesung

Wir werden uns beschäftigen mit:

- Variante des endlichen Automaten

Stichwörter: DFA, NFA

Themen der heutigen Vorlesung

Wir werden uns beschäftigen mit:

- Varianten des endlichen Automaten
- Ein Beweis: Nicht jede formale Sprache ist von einem DFA akzeptierbar.

Stichwörter: Diagonalbeweis

Themen der heutigen Vorlesung

Wir werden uns beschäftigen mit:

- Varianten des endlichen Automaten
- Ein Beweis: Nicht jede formale Sprache ist von einem DFA akzeptierbar.
- Ordnungen auf Wörtern

Stichwörter: lexikalische Ordnung, Gödelisierung

Themen der heutigen Vorlesung

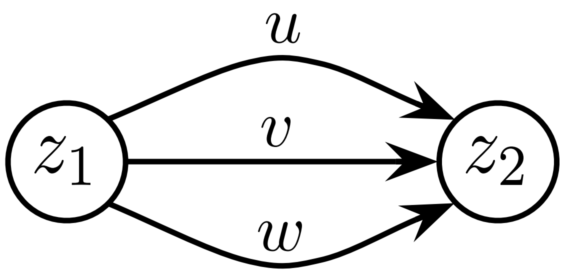
Wir werden uns beschäftigen mit:

- Varianten des endlichen Automaten
- Ein Beweis: Nicht jede formale Sprache ist von einem DFA akzeptierbar.
- Ordnungen auf Wörtern
- Eigenschaften endlicher Automaten

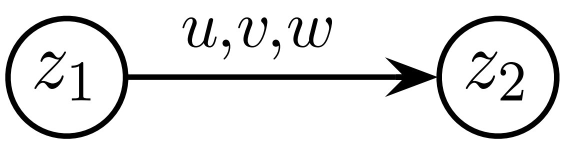
Stichwörter: vollständig, buchstabierend, λ-frei, Äquivalenz von Automaten

Kurznotation

Statt

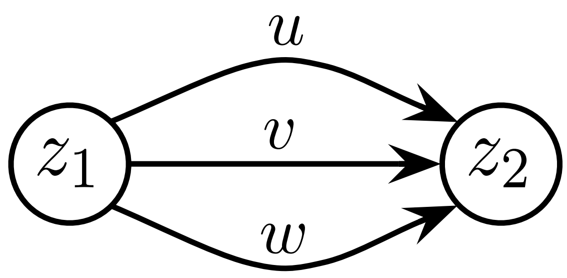


schreiben wir abkürzend:

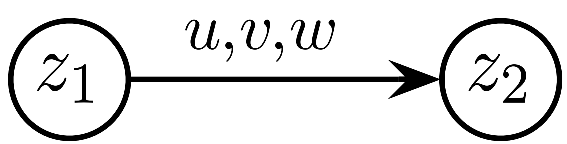


Kurznotation

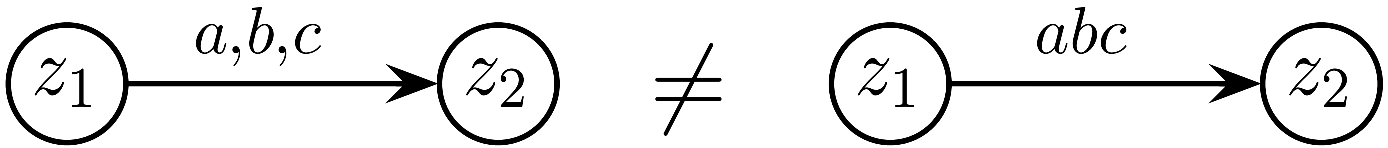
Statt



schreiben wir abkürzend:



Zu beachten:



Einsatz von DFAs

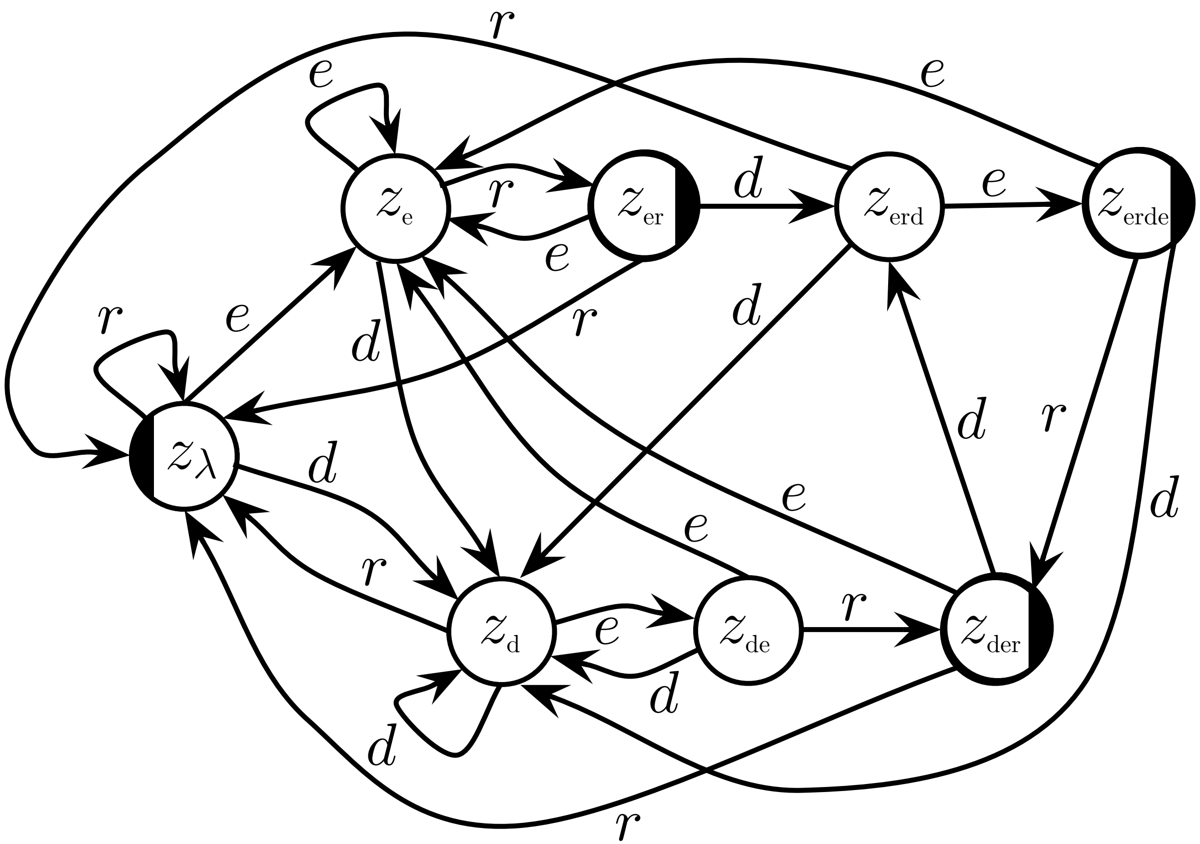
- Spezifikation/Modellierung von einfachen realen Maschinen (z.B. Fahrkartenautomaten)

Einsatz von DFAs

- Spezifikation/Modellierung von einfachen realen Maschinen (z.B. Fahrkartenautomaten)
- einfache Steuerungsaufgaben

Einsatz von DFAs

- Spezifikation/Modellierung von einfachen realen Maschinen (z.B. Fahrkartenautomaten)
- einfache Steuerungsaufgaben
- einfache Parser:



Eigenschaften

Ein DFA $A := (Z, \Sigma, \delta, z_0, Z_{\text{end}})$ heißt:

- **vollständig** (Abk.: vDFA) genau dann, wenn für jedes $(p, x) \in Z \times \Sigma$ ein $q \in Z$ existiert, so dass $q = \delta(p, x)$ ist.

Jede Eingabe kann verarbeitet werden!

Eigenschaften

Ein DFA $A := (Z, \Sigma, \delta, z_0, Z_{\text{end}})$ heißt:

- **vollständig** (Abk.: vDFA) genau dann, wenn für jedes $(p, x) \in Z \times \Sigma$ ein $q \in Z$ existiert, so dass $q = \delta(p, x)$ ist.

Jede Eingabe kann verarbeitet werden!

- **initial zusammenhängend** (Abk.: izDFA) genau dann, wenn zu jedem Zustand $p \in Z$ ein Wort $w \in \Sigma^*$ existiert, mit $p = \widehat{\delta}(z_0, w)$.

Jeder Zustand kann von z_0 erreicht werden!

Eigenschaften

Ein DFA $A := (Z, \Sigma, \delta, z_0, Z_{\text{end}})$ heißt:

- **vollständig** (Abk.: vDFA) genau dann, wenn für jedes $(p, x) \in Z \times \Sigma$ ein $q \in Z$ existiert, so dass $q = \delta(p, x)$ ist.

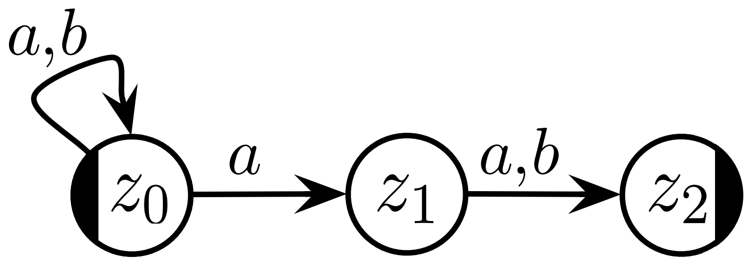
Jede Eingabe kann verarbeitet werden!

- **initial zusammenhängend** (Abk.: izDFA) genau dann, wenn zu jedem Zustand $p \in Z$ ein Wort $w \in \Sigma^*$ existiert, mit $p = \widehat{\delta}(z_0, w)$.

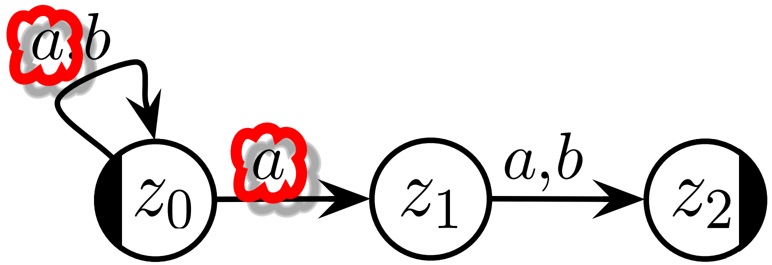
Jeder Zustand kann von z_0 erreicht werden!

- Zwei DFA, A und B, heißen genau dann **äquivalent**, wenn sie die gleiche Sprache akzeptieren, d.h. $L(A) = L(B)$ ist.

Ist dies ein DFA?

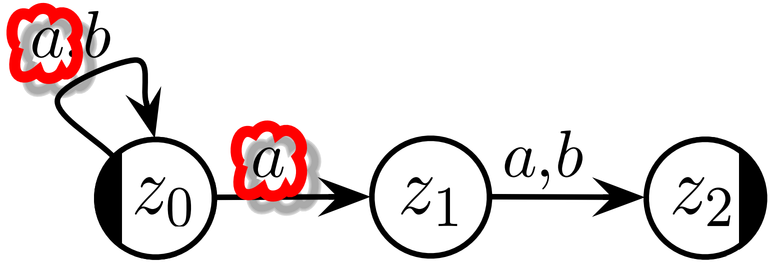


Ist dies ein DFA?



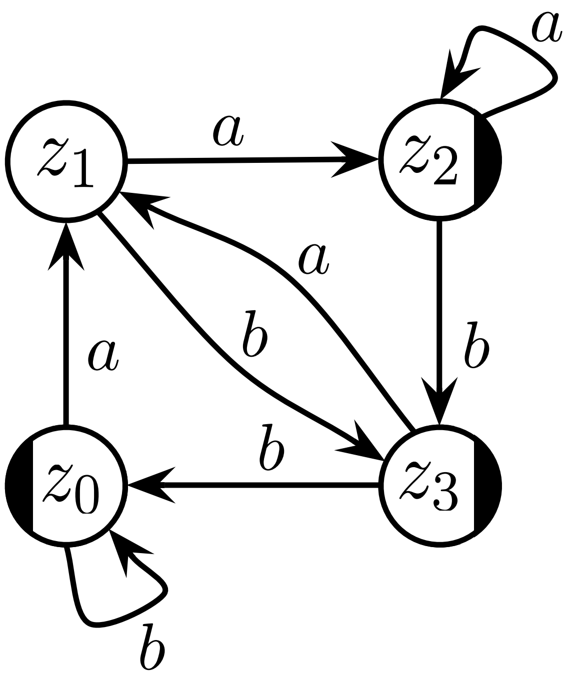
NEIN!

Ist dies ein DFA?

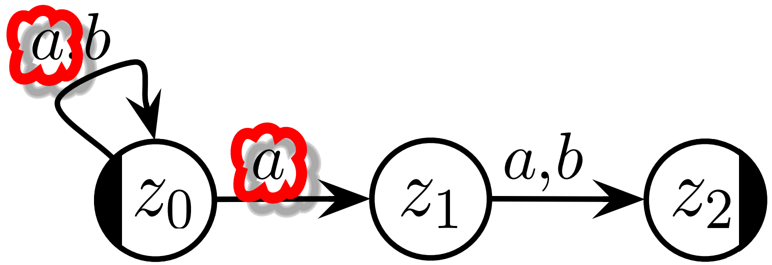


NEIN!

... und dieser Automat:

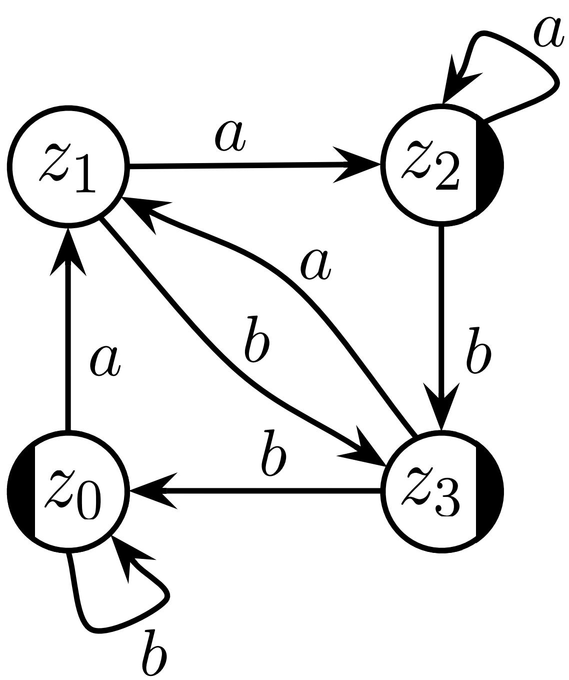


Ist dies ein DFA?



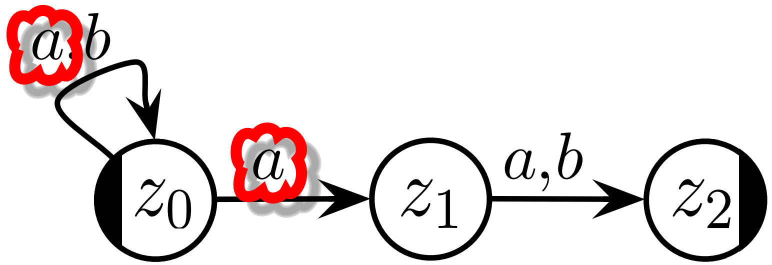
NEIN!

…und dieser Automat:



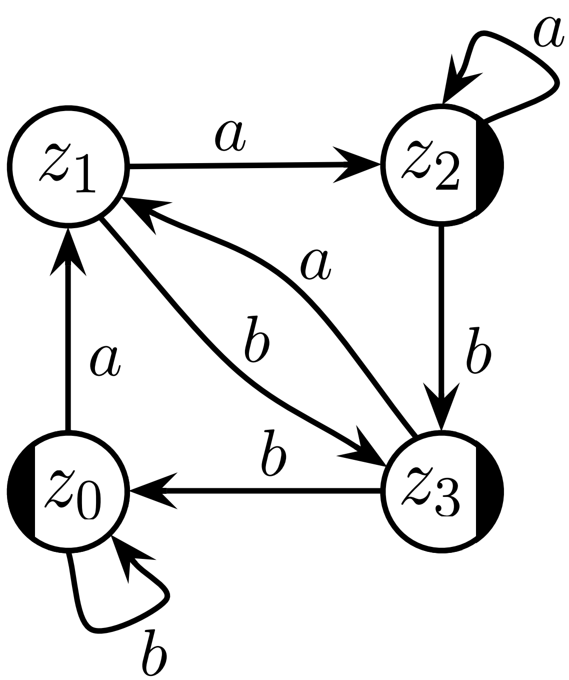
JA!

Ist dies ein DFA?



NEIN!

... und dieser Automat:



JA!

... aber was ist die akzeptierte Sprache?

Zustandsübergänge

- **Beim DFA:** $\delta :\subseteq Z \times \Sigma \longrightarrow Z$

Verwendung einer Funktion mit Nachbereich Z zur Beschreibung der erlaubten Zustandsübergänge.

$\Rightarrow$ Ein Zustand hat für jede Eingabe **höchstens einen** Folgezustand.

Zustandsübergänge

- **Beim DFA:** $\delta :\subseteq Z \times \Sigma \longrightarrow Z$
- Wie kann dies verallgemeinert werden?

Verwendung einer Funktion mit Nachbereich Z zur Beschreibung der erlaubten Zustandsübergänge.

$\Rightarrow$ Ein Zustand hat für jede Eingabe **höchstens einen** Folgezustand.

Zustandsübergänge

- **Beim DFA:** $\delta :\subseteq Z \times \Sigma \longrightarrow Z$
- Wie kann dies verallgemeinert werden?
- **Beim NFA:** $\delta : Z \times \Sigma \longrightarrow 2^Z$

Änderung des **Nachbereichs**: Verwendung der Potenzmenge von Z = Menge aller Teilmengen von Zuständen aus Z.

$\Rightarrow$ Ein Zustand hat eine Menge von Folgezuständen.

Hier wird üblicherweise eine *totale* Funktion vorausgesetzt.

Zustandsübergänge

- **Beim DFA:** $\delta :\subseteq Z \times \Sigma \longrightarrow Z$
- Wie kann dies verallgemeinert werden?
- **Beim NFA:** $\delta : Z \times \Sigma \longrightarrow 2^Z$
- **Beim NFA:** $K \subseteq Z \times \Sigma \times Z$

Änderung der **Überführungsfunktion** in eine **Relation**: Relationen müssen nicht eindeutig sein!

$\Rightarrow$ Ein Zustand kann mit mehreren (Folge-)Zuständen in Relation stehen.

NFA: 1. Definition

...mit 2^Z als Nachbereich der Übergangsfunktion:

NFA: 1. Definition

…mit 2^Z als Nachbereich der Übergangsfunktion:

Ein **nichtdeterministischer endlicher Automat** (**NFA**, engl. **n***ondeterministic* **f***inite* **a***utomaton*) wird spezifiziert werden durch ein Tupel $A := (Z, \Sigma, \delta, Z_{\text{start}}, Z_{\text{end}})$, mit:

- Z ist endliche Menge von **Zuständen**,
- Σ ist endliches Alphabet von **Eingabesymbolen**,
- $\delta : Z \times \Sigma \longrightarrow 2^Z$ ist **Überführungsfunktion**,
- $Z_{\text{start}} \subseteq Z$ ist **Menge der Startzustände** und
- $Z_{\text{end}} \subseteq Z$ ist **Menge der Endzustände**.

NFA: 2. Definition

. . . mit Kantenmenge, d.h. einer Übergangsrelation:

NFA: 2. Definition

…mit Kantenmenge, d.h. einer Übergangsrelation:

Ein **nichtdeterministischer, endlicher Automat** ist ein Tupel $A := (Z, \Sigma, K, Z_{\text{start}}, Z_{\text{end}})$, mit:

- Z ist endliche Menge von **Zuständen**,
- $Z_{\text{start}} \subseteq Z$ ist **Menge der Startzustände**,
- $Z_{\text{end}} \subseteq Z$ ist **Menge der Endzustände**,
- Σ ist ein endliches **Eingabealphabet** und
- $K \subset Z \times \Sigma^* \times Z$ ist endliche **Zustandsübergangsrelation**.

Achtung: Hier weitere Verallgemeinerung bei den gelesenen Symbolen des Eingabewortes!

…**mehrere aufeinanderfolgende** Symbole lesbar

NFA: 2. Definition

…mit Kantenmenge, d.h. einer Übergangsrelation:

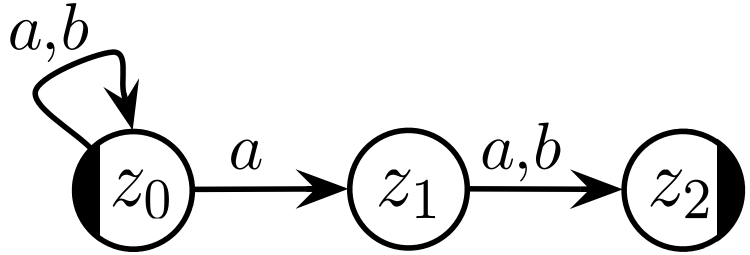
Ein **nichtdeterministischer, endlicher Automat** ist ein Tupel $A := (Z, \Sigma, K, Z_{\mathsf{start}}, Z_{\mathsf{end}})$, mit:

- Z ist endliche Menge von **Zuständen**,
- $Z_{\mathsf{start}} \subseteq Z$ ist **Menge der Startzustände**,
- $Z_{\mathsf{end}} \subseteq Z$ ist **Menge der Endzustände**,
- Σ ist ein endliches **Eingabealphabet** und
- $K \subset Z \times \Sigma^* \times Z$ ist endliche **Zustandsübergangsrelation**.

- Ein Element $(p, \lambda, q) \in K$ heißt λ-**Kante**.
- Ein NFA ohne λ-Kanten wird als λ-**freier NFA** bezeichnet.

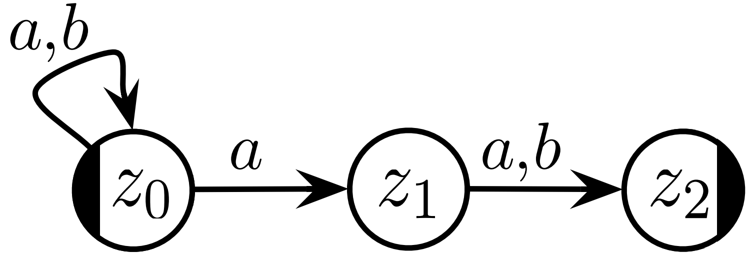
Ein Beispiel-NFA

- Welche Sprache akzeptiert folgender NFA?



Ein Beispiel-NFA

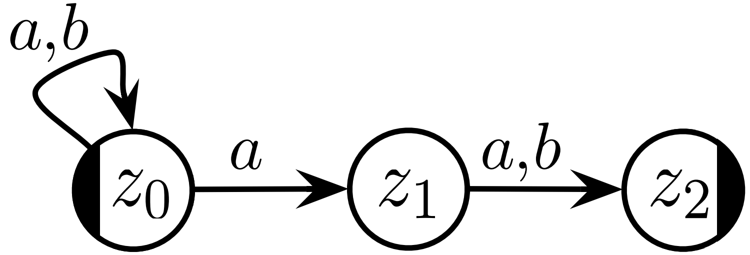
- Welche Sprache akzeptiert folgender NFA?



- Wir benötigen eine formale Definition der akzeptierten Sprache!

Ein Beispiel-NFA

- Welche Sprache akzeptiert folgender NFA?

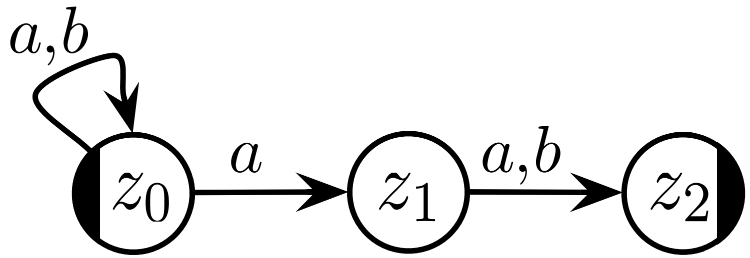


- Wir benötigen eine formale Definition der akzeptierten Sprache!
- Wie sah die Definition für DFA aus?

$$L(A) := \{w \in \Sigma^* \mid \widehat{\delta}(z_0, w) \in Z_{\mathsf{end}}\}.$$

Ein Beispiel-NFA

- Welche Sprache akzeptiert folgender NFA?



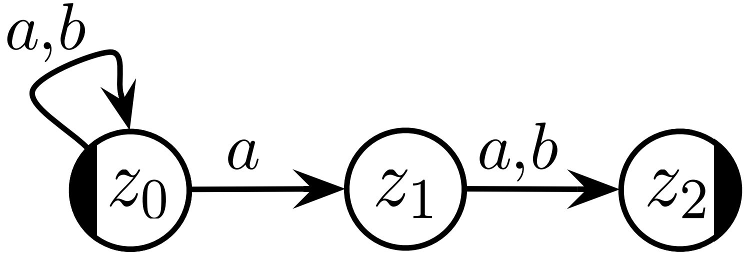
- Wir benötigen eine formale Definition der akzeptierten Sprache!
- Wie sah die Definition für DFA aus?

$$L(A) := \{w \in \Sigma^* \mid \widehat{\delta}(z_0, w) \in Z_{\mathsf{end}}\}.$$

- Wir hatten die Zustandsüberführungsfunktion für Wörter erweitert.

Ein Beispiel-NFA

- Welche Sprache akzeptiert folgender NFA?



- Wir benötigen eine formale Definition der akzeptierten Sprache!
- Wie sah die Definition für DFA aus?

$$L(A) := \{w \in \Sigma^* \mid \widehat{\delta}(z_0, w) \in Z_{\mathsf{end}}\}.$$

- Wir hatten die Zustandsüberführungsfunktion für Wörter erweitert.
- Ähnlich wollen wir für NFA vorgehen!

Rechnung

- Eine **Kantenfolge**

$$p := (z_0, x_1, z_1), (z_1, x_2, z_2), \ldots, (z_{n-1}, x_n, z_n)$$

mit $(z_{i-1}, x_i, z_i) \in K$, für $0 < i \leq n$, heißt
Rechnung von A für das Wort
$|p| := w := x_1 x_2 \ldots x_n \in \Sigma^*$.

Rechnung

- Eine **Kantenfolge**

$$p := (z_0, x_1, z_1), (z_1, x_2, z_2), \ldots, (z_{n-1}, x_n, z_n)$$

mit $(z_{i-1}, x_i, z_i) \in K$, für $0 < i \leq n$, heißt **Rechnung** von A für das Wort $|p| := w := x_1 x_2 \ldots x_n \in \Sigma^*$.

- **Notation:** $z_0 \xrightarrow[w]{*} z_n$ und für alle $w \in \Sigma^*$ ist $\xrightarrow[w]{*}$ eine Teilmenge von $Z \times Z$ (binäre Relation auf Z).

Rechnung

- Eine **Kantenfolge**

$$p := (z_0, x_1, z_1), (z_1, x_2, z_2), \ldots, (z_{n-1}, x_n, z_n)$$

  mit $(z_{i-1}, x_i, z_i) \in K$, für $0 < i \leq n$, heißt **Rechnung** von A für das Wort $|p| := w := x_1 x_2 \ldots x_n \in \Sigma^*$.

- **Notation:** $z_0 \xrightarrow[w]{*} z_n$ und für alle $w \in \Sigma^*$ ist $\xrightarrow[w]{*}$ eine Teilmenge von $Z \times Z$ (binäre Relation auf Z).
- Eine **Erfolgsrechnung** für das Wort w, ist eine Rechnung, bei der $z_0 \in Z_{\text{start}}$ und $z_n \in Z_{\text{end}}$ ist.

  Spezialfall: $w = \lambda$, $z_0 = z_n$ mit $z_n \in Z_{\text{start}} \cap Z_{\text{end}}$.

Akzeptierte Sprache eines NFA

- $L(A) :=$

  $\{w \in \Sigma^* \mid \exists z_0 \in Z_{\mathsf{start}} : \exists z_n \in Z_{\mathsf{end}} : z_0 \xrightarrow[w]{*} z_n\}$

  bezeichnet die vom NFA A **akzeptierte Sprache**.

Akzeptierte Sprache eines NFA

- $L(A) :=$

  $\{w \in \Sigma^* \mid \exists z_0 \in Z_{\mathsf{start}} : \exists z_n \in Z_{\mathsf{end}} : z_0 \xrightarrow[w]{*} z_n\}$

  bezeichnet die vom NFA A **akzeptierte Sprache**.
- Jeder DFA ist auch als NFA darstellbar!

Akzeptierte Sprache eines NFA

- $L(A) :=$

  $\{w \in \Sigma^* \mid \exists z_0 \in Z_{\mathsf{start}} : \exists z_n \in Z_{\mathsf{end}} : z_0 \xrightarrow[w]{*} z_n\}$

  bezeichnet die vom NFA A **akzeptierte Sprache**.
- Jeder DFA ist auch als NFA darstellbar!
- Eine Funktion ist ein Spezialfall einer Relation, so dass die Zustandsüberführungsfunktion als Kantenrelation interpretiert werden kann.

Akzeptierte Sprache eines NFA

- $L(A) :=$

  $\{w \in \Sigma^* \mid \exists z_0 \in Z_{\mathsf{start}} : \exists z_n \in Z_{\mathsf{end}} : z_0 \xrightarrow[w]{*} z_n\}$

  bezeichnet die vom NFA A **akzeptierte Sprache**.
- Jeder DFA ist auch als NFA darstellbar!
- Eine Funktion ist ein Spezialfall einer Relation, so dass die Zustandsüberführungsfunktion als Kantenrelation interpretiert werden kann.
- Aber nicht jeder NFA ist (direkt) auch ein DFA.

Akzeptierte Sprache eines NFA

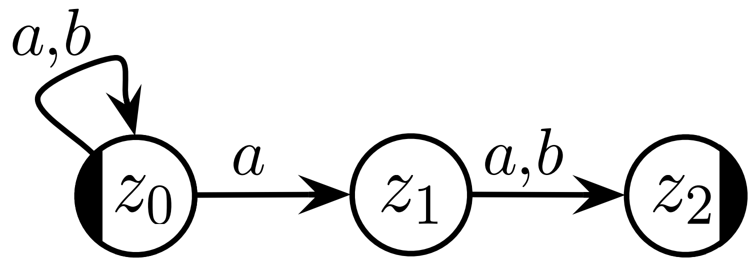
- $L(A) :=$

  $\{w \in \Sigma^* \mid \exists z_0 \in Z_{\mathsf{start}} : \exists z_n \in Z_{\mathsf{end}} : z_0 \xrightarrow[w]{*} z_n\}$

  bezeichnet die vom NFA A **akzeptierte Sprache**.
- Jeder DFA ist auch als NFA darstellbar!
- Eine Funktion ist ein Spezialfall einer Relation, so dass die Zustandsüberführungsfunktion als Kantenrelation interpretiert werden kann.
- Aber nicht jeder NFA ist (direkt) auch ein DFA.
- **Frage:** Können NFA mehr Sprachen akzeptieren, als DFA?

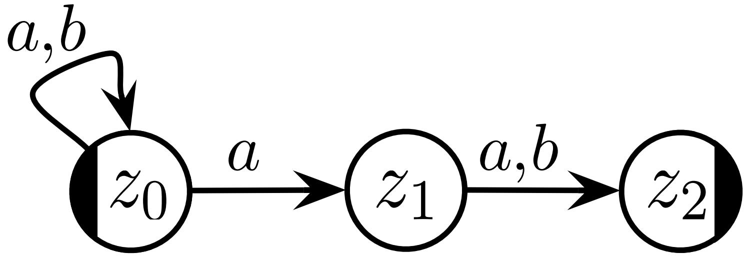
. . . zurück zum Beispielautomaten

$A :$



...zurück zum Beispielautomaten

A :

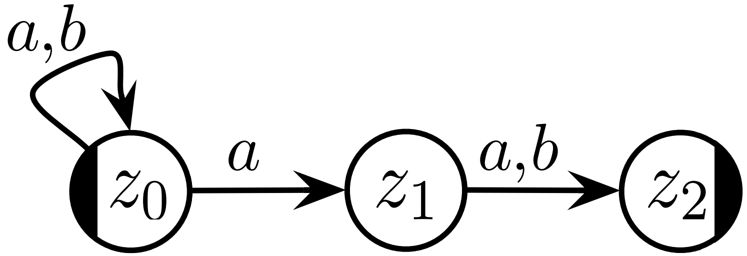


Beispielrechnungen:

$p := (z_0, a, z_0), (z_0, b, z_0), (z_0, b, z_0)$ ist einzige Rechnung von A auf dem Wort abb.

…zurück zum Beispielautomaten

A :



Beispielrechnungen:

$p := (z_0, a, z_0), (z_0, b, z_0), (z_0, b, z_0)$ ist einzige Rechnung von A auf dem Wort abb.

Sowohl $q := (z_0, b, z_0), (z_0, b, z_0), (z_0, a, z_1), (z_1, a, z_2)$ als auch $q' := (z_0, b, z_0), (z_0, b, z_0), (z_0, a, z_0), (z_0, a, z_1)$ sind Rechnungen von A auf $bbaa$.

Allein q ist eine Erfolgsrechnung für $bbaa$!

Eigenschaften

Ein NFA $A := (Z, \Sigma, K, Z_{\mathsf{start}}, Z_{\mathsf{end}})$ heißt

- **buchstabierend** genau dann, wenn $K \subseteq Z \times \Sigma \times Z$ ist,

Eigenschaften

Ein NFA $A := (Z, \Sigma, K, Z_{\mathsf{start}}, Z_{\mathsf{end}})$ heißt

- **buchstabierend** genau dann, wenn $K \subseteq Z \times \Sigma \times Z$ ist,
- λ-**frei** genau dann, wenn $K \subseteq Z \times \Sigma^{+} \times Z$ ist.

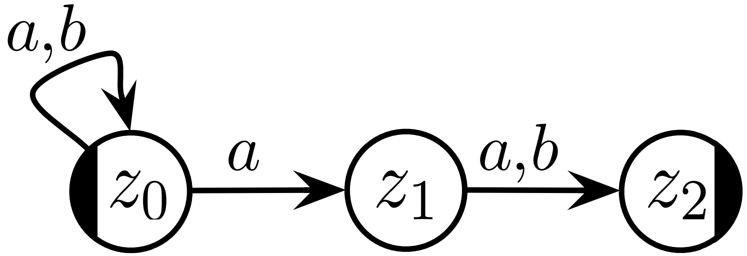
Eigenschaften

Ein NFA $A := (Z, \Sigma, K, Z_{\textsf{start}}, Z_{\textsf{end}})$ heißt

- **buchstabierend** genau dann, wenn $K \subseteq Z \times \Sigma \times Z$ ist,
- λ-**frei** genau dann, wenn $K \subseteq Z \times \Sigma^{+} \times Z$ ist.
- Zwei NFA, A und B, heißen genau dann **äquivalent**, wenn sie die gleiche Sprache akzeptieren, d.h. $L(A) = L(B)$ ist.

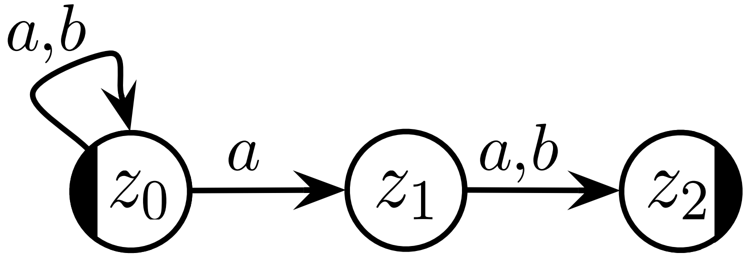
. . . nochmal der Beispielautomat

$A:$



. . . nochmal der Beispielautomat

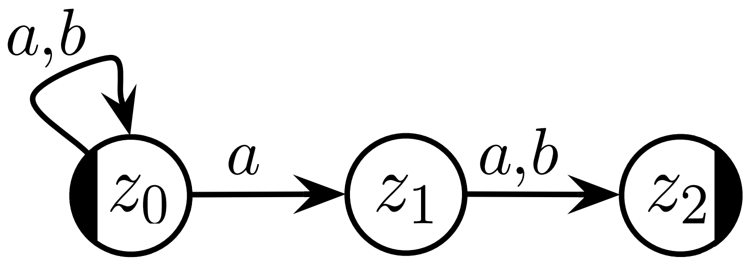
A :



A ist buchstabierend und λ-frei.

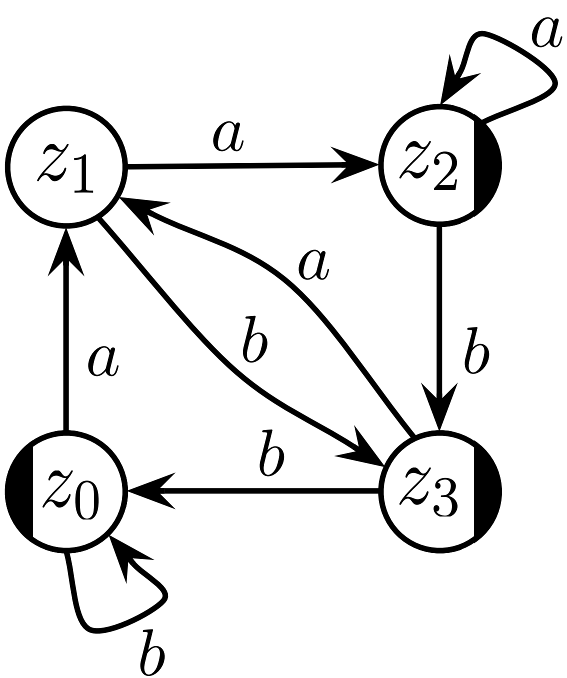
...nochmal der Beispielautomat

A :



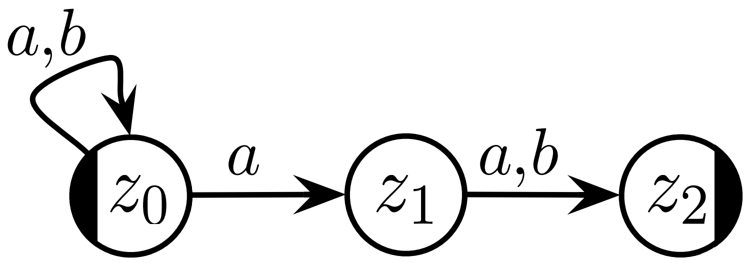
A ist buchstabierend und λ-frei.

Ein (altbekannter) äquivalenter DFA:



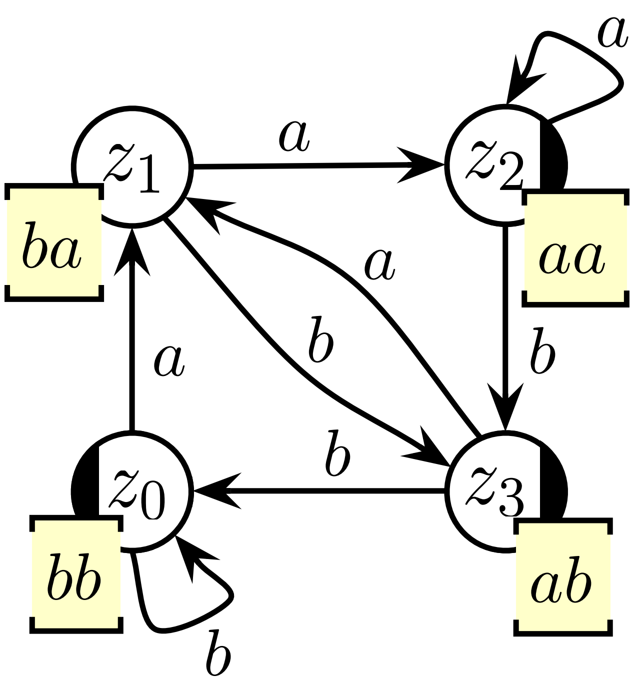
...nochmal der Beispielautomat

A :



A ist buchstabierend und λ-frei.

Ein (altbekannter) äquivalenter DFA:



Nicht reguläre Sprachen

- **Frage:** Gibt es formale Sprachen, die von **keinem** DFA akzeptiert werden können?

Nicht reguläre Sprachen

- **Frage:** Gibt es formale Sprachen, die von **keinem** DFA akzeptiert werden können?
- Können diese Sprachen evtl. von einem NFA akzeptiert werden?

Nicht reguläre Sprachen

- **Frage:** Gibt es formale Sprachen, die von **keinem** DFA akzeptiert werden können?
- Können diese Sprachen evtl. von einem NFA akzeptiert werden?
- Wir zeigen, dass es eine Sprache $L \subseteq \{0,1\}^*$ gibt, die von keinem FA akzeptiert werden kann!

Nicht reguläre Sprachen

- **Frage:** Gibt es formale Sprachen, die von **keinem** DFA akzeptiert werden können?
- Können diese Sprachen evtl. von einem NFA akzeptiert werden?
- Wir zeigen, dass es eine Sprache $L \subseteq \{0,1\}^*$ gibt, die von keinem FA akzeptiert werden kann!
- Wir benötigen: **Codierung von Automaten als Wörter über einem endlichen Alphabet**, d.h. eine *injektive* Funktion ν_{FA}

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.
  - $\forall\, x, y \in \Sigma\ \forall\, u, v \in \Sigma^+$:

$$(xu \prec^{\text{lex}} yv) \leftrightarrow (x \prec y) \vee \big[(x = y) \wedge (u \prec^{\text{lex}} v)\big]$$

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.
  - $\forall\, x, y \in \Sigma\ \forall\, u, v \in \Sigma^+$:

$$(xu \prec^{\text{lex}} yv) \leftrightarrow (x \prec y) \vee \left[(x = y) \wedge (u \prec^{\text{lex}} v)\right]$$

- $\preceq^{\text{lex}} := \prec^{\text{lex}} \cup Id_{\Sigma^*}$ heißt **lexikographische Ordnung** auf Σ^*.

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.
  - $\forall\, x, y \in \Sigma\ \forall\, u, v \in \Sigma^+$:

$$(xu \prec^{\text{lex}} yv) \leftrightarrow (x \prec y) \vee \left[(x = y) \wedge (u \prec^{\text{lex}} v)\right]$$

- $\preceq^{\text{lex}} := \prec^{\text{lex}} \cup Id_{\Sigma^*}$ heißt **lexikographische Ordnung** auf Σ^*.
- **Beispiel:** ALLE $\prec^{\text{lex}}$ FINDEN $\prec^{\text{lex}}$ THEORIE $\prec^{\text{lex}}$ WUNDERBAR

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.
  - $\forall\, x, y \in \Sigma\ \forall\, u, v \in \Sigma^+ :$

$$(xu \prec^{\text{lex}} yv) \leftrightarrow (x \prec y) \vee \left[(x = y) \wedge (u \prec^{\text{lex}} v)\right]$$

- $\preceq^{\text{lex}} := \prec^{\text{lex}} \cup Id_{\Sigma^*}$ heißt **lexikographische Ordnung** auf Σ^*.
- **Beispiel:** ALLE $\prec^{\text{lex}}$ FINDEN $\prec^{\text{lex}}$ THEORIE $\prec^{\text{lex}}$ WUNDERBAR
- Wenn wir systematisch alle Wörter aus $\{a, b\}^*$ mit $a \prec b$ aufzählen wollen, eignet sich $\prec^{\text{lex}}$ nicht!

Lexikographische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikographische Erweiterung** $\prec^{\text{lex}} \subseteq \Sigma^* \times \Sigma^*$ von $\prec$ definiert durch:
  - $\lambda \prec^{\text{lex}} w$ für alle $w \in \Sigma^*$.
  - $\forall\, x, y \in \Sigma\ \forall\, u, v \in \Sigma^+$:

$$(xu \prec^{\text{lex}} yv) \leftrightarrow (x \prec y) \vee \left[(x = y) \wedge (u \prec^{\text{lex}} v)\right]$$

- $\preceq^{\text{lex}} := \prec^{\text{lex}} \cup Id_{\Sigma^*}$ heißt **lexikographische Ordnung** auf Σ^*.
- **Beispiel:** ALLE $\prec^{\text{lex}}$ FINDEN $\prec^{\text{lex}}$ THEORIE $\prec^{\text{lex}}$ WUNDERBAR
- Wenn wir systematisch alle Wörter aus $\{a, b\}^*$ mit $a \prec b$ aufzählen wollen, eignet sich $\prec^{\text{lex}}$ nicht!
- $\lambda, a, aa, aaa, aaaa, aaaaa, \ldots$

Lexikalische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikalische** Erweiterung $\prec^{\mathrm{lg-lex}}$ von $\prec$ definiert durch:

Lexikalische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikalische** Erweiterung $\prec^{\mathrm{lg-lex}}$ von $\prec$ definiert durch:
  - $w_1 \prec^{\mathrm{lg-lex}} w_2$ gelte für beliebige Wörter $w_1, w_2 \in \Sigma^*$ gdw.

$$|w_1| < |w_2| \text{ oder } |w_1| = |w_2| \ \wedge \ w_1 \prec^{\mathrm{lex}} w_2.$$

Lexikalische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikalische** Erweiterung $\prec^{\text{lg}-\text{lex}}$ von $\prec$ definiert durch:
  - $w_1 \prec^{\text{lg}-\text{lex}} w_2$ gelte für beliebige Wörter $w_1, w_2 \in \Sigma^*$ gdw.

    $$|w_1| < |w_2| \text{ oder } |w_1| = |w_2| \wedge w_1 \prec^{\text{lex}} w_2.$$

  - $\preceq^{\text{lg}-\text{lex}} := \prec^{\text{lg}-\text{lex}} \cup Id_{\Sigma^*}$ heißt **lexikalische Ordnung** auf Σ^*.

Lexikalische Ordnung

- Sei $(\Sigma, \prec)$ ein mit $\prec$ linear geordnetes Alphabet, dann ist die **lexikalische** Erweiterung $\prec^{\mathrm{lg-lex}}$ von $\prec$ definiert durch:
  - $w_1 \prec^{\mathrm{lg-lex}} w_2$ gelte für beliebige Wörter $w_1, w_2 \in \Sigma^*$ gdw.

$$|w_1| < |w_2| \text{ oder } |w_1| = |w_2| \wedge w_1 \prec^{\mathrm{lex}} w_2.$$

  - $\preceq^{\mathrm{lg-lex}} := \prec^{\mathrm{lg-lex}} \cup Id_{\Sigma^*}$ heißt **lexikalische Ordnung** auf Σ^*.
- ... und hiermit ist das Problem gelöst:

$\lambda, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots$

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.
- Zur Codierung verwendetes Alphabet: $\Omega := \{0; 1; \, , \, ; (;)\}$

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.
- Zur Codierung verwendetes Alphabet:
  $\Omega := \{0; 1; ,; (;)\}$
- **Die Codierung:** $\nu_{FA} : \Omega^* \longrightarrow \mathcal{A}_{\{0,1\}}$ sei surjektiv und ...

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.
- Zur Codierung verwendetes Alphabet: $\Omega := \{0; 1; ,; (;)\}$
- **Die Codierung:** $\nu_{FA} : \Omega^* \longrightarrow \mathcal{A}_{\{0,1\}}$ sei surjektiv und . . .
- $(0^n, x, 0^m)$ bezeichne eine Kante des NFA.

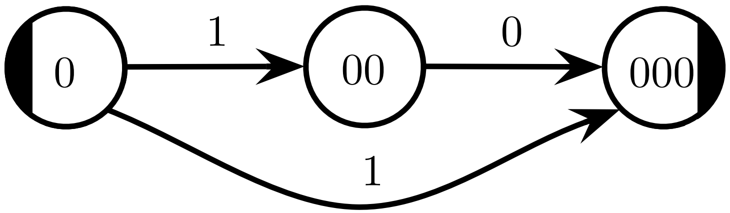
Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.
- Zur Codierung verwendetes Alphabet: $\Omega := \{0; 1; ,; (;)\}$
- **Die Codierung:** $\nu_{FA} : \Omega^* \longrightarrow \mathcal{A}_{\{0,1\}}$ sei surjektiv und ...
- $(0^n, x, 0^m)$ bezeichne eine Kante des NFA.
- (0^n) bezeichne einen Endzustand.

Notation von Automaten

- Einschränkung der Automaten auf genau einen Startzustand und das Alphabet $\Sigma = \{0, 1\}$.
- Zustände haben Namen aus $\{0, 00, 000, 0000, \ldots\}$. Der Startzustand sei der Zustand 0.
- $\mathcal{A}_{\{0,1\}}$ bezeichne diese Klasse von Automaten.
- Zur Codierung verwendetes Alphabet:
  $\Omega := \{0; 1; , ; (;)\}$
- **Die Codierung:** $\nu_{FA} : \Omega^* \longrightarrow \mathcal{A}_{\{0,1\}}$ sei surjektiv und ...
- $(0^n, x, 0^m)$ bezeichne eine Kante des NFA.
- (0^n) bezeichne einen Endzustand.
- $\nu_{FA}(w) = A$ gdw. $w \in \prod_{k \in K} k \cdot \prod_{z \in Z_{\text{end}}} (z)$.

Beispiel: Automatennotation



$$(0, 1, 00)(00, 0, 000)(0, 1, 000)(000)$$

und

$$(00, 0, 000)(0, 1, 000)(0, 1, 00)(000)$$

sind Codierungen des obigen NFA.

Diagonalbeweis

Nummerierung aller Automaten $A_i \in \mathcal{A}_{\{0,1\}}$ mit charakteristischer Funktion für $L(A_i)$:

Diagonalbeweis

Nummerierung aller Automaten $A_i \in \mathcal{A}_{\{0,1\}}$ mit charakteristischer Funktion für $L(A_i)$:

i	A_i	w_0	w_1	w_2	$\cdots$	w_n	$\cdots$
0	A_0	$[w_0 \in L(A_0)]$	$[w_1 \in L(A_0)]$	$[w_2 \in L(A_0)]$	$\cdots$	$[w_n \in L(A_0)]$	$\cdots$
1	A_1	$[w_0 \in L(A_1)]$	$[w_1 \in L(A_1)]$	$[w_2 \in L(A_1)]$	$\cdots$	$[w_n \in L(A_1)]$	$\cdots$
2	A_2	$[w_0 \in L(A_2)]$	$[w_1 \in L(A_2)]$	$[w_2 \in L(A_2)]$	$\cdots$	$[w_n \in L(A_2)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\cdots$
n	A_n	$[w_0 \in L(A_n)]$	$[w_1 \in L(A_n)]$	$[w_2 \in L(A_n)]$	$\cdots$	$[w_n \in L(A_n)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\ddots$

Diagonalbeweis

Nummerierung aller Automaten $A_i \in \mathcal{A}_{\{0,1\}}$ mit charakteristischer Funktion für $L(A_i)$:

i	A_i	w_0	w_1	w_2	$\cdots$	w_n	$\cdots$
0	A_0	$[w_0 \in L(A_0)]$	$[w_1 \in L(A_0)]$	$[w_2 \in L(A_0)]$	$\cdots$	$[w_n \in L(A_0)]$	$\cdots$
1	A_1	$[w_0 \in L(A_1)]$	$[w_1 \in L(A_1)]$	$[w_2 \in L(A_1)]$	$\cdots$	$[w_n \in L(A_1)]$	$\cdots$
2	A_2	$[w_0 \in L(A_2)]$	$[w_1 \in L(A_2)]$	$[w_2 \in L(A_2)]$	$\cdots$	$[w_n \in L(A_2)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\cdots$
n	A_n	$[w_0 \in L(A_n)]$	$[w_1 \in L(A_n)]$	$[w_2 \in L(A_n)]$	$\cdots$	$[w_n \in L(A_n)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\ddots$
	L	$[w_0 \notin L(A_0)]$	$[w_1 \notin L(A_1)]$	$[w_2 \notin L(A_2)]$	$\cdots$	$[w_n \notin L(A_n)]$	$\cdots$

Diagonalbeweis

Nummerierung aller Automaten $A_i \in \mathcal{A}_{\{0,1\}}$ mit charakteristischer Funktion für $L(A_i)$:

i	A_i	w_0	w_1	w_2	$\cdots$	w_n	$\cdots$
0	A_0	$[w_0 \in L(A_0)]$	$[w_1 \in L(A_0)]$	$[w_2 \in L(A_0)]$	$\cdots$	$[w_n \in L(A_0)]$	$\cdots$
1	A_1	$[w_0 \in L(A_1)]$	$[w_1 \in L(A_1)]$	$[w_2 \in L(A_1)]$	$\cdots$	$[w_n \in L(A_1)]$	$\cdots$
2	A_2	$[w_0 \in L(A_2)]$	$[w_1 \in L(A_2)]$	$[w_2 \in L(A_2)]$	$\cdots$	$[w_n \in L(A_2)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\cdots$
n	A_n	$[w_0 \in L(A_n)]$	$[w_1 \in L(A_n)]$	$[w_2 \in L(A_n)]$	$\cdots$	$[w_n \in L(A_n)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\ddots$
	L	$[w_0 \notin L(A_0)]$	$[w_1 \notin L(A_1)]$	$[w_2 \notin L(A_2)]$	$\cdots$	$[w_n \notin L(A_n)]$	$\cdots$

$$\forall i \in \mathbb{N} : w_i \in L \text{ gdw. } w_i \notin L(A_i)$$

Diagonalbeweis

Nummerierung aller Automaten $A_i \in \mathcal{A}_{\{0,1\}}$ mit charakteristischer Funktion für $L(A_i)$:

i	A_i	w_0	w_1	w_2	$\cdots$	w_n	$\cdots$
0	A_0	$[w_0 \in L(A_0)]$	$[w_1 \in L(A_0)]$	$[w_2 \in L(A_0)]$	$\cdots$	$[w_n \in L(A_0)]$	$\cdots$
1	A_1	$[w_0 \in L(A_1)]$	$[w_1 \in L(A_1)]$	$[w_2 \in L(A_1)]$	$\cdots$	$[w_n \in L(A_1)]$	$\cdots$
2	A_2	$[w_0 \in L(A_2)]$	$[w_1 \in L(A_2)]$	$[w_2 \in L(A_2)]$	$\cdots$	$[w_n \in L(A_2)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\cdots$
n	A_n	$[w_0 \in L(A_n)]$	$[w_1 \in L(A_n)]$	$[w_2 \in L(A_n)]$	$\cdots$	$[w_n \in L(A_n)]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\ddots$
	L	$[w_0 \notin L(A_0)]$	$[w_1 \notin L(A_1)]$	$[w_2 \notin L(A_2)]$	$\cdots$	$[w_n \notin L(A_n)]$	$\cdots$

$$\forall i \in \mathbb{N} : L \neq L(A_i)$$

Diagonalisierung (Zusammenf.)

- Nummeriere alle NFAs

Diagonalisierung (Zusammenf.)

- Nummeriere alle NFAs
- Nummeriere alle Wörter

Diagonalisierung (Zusammenf.)

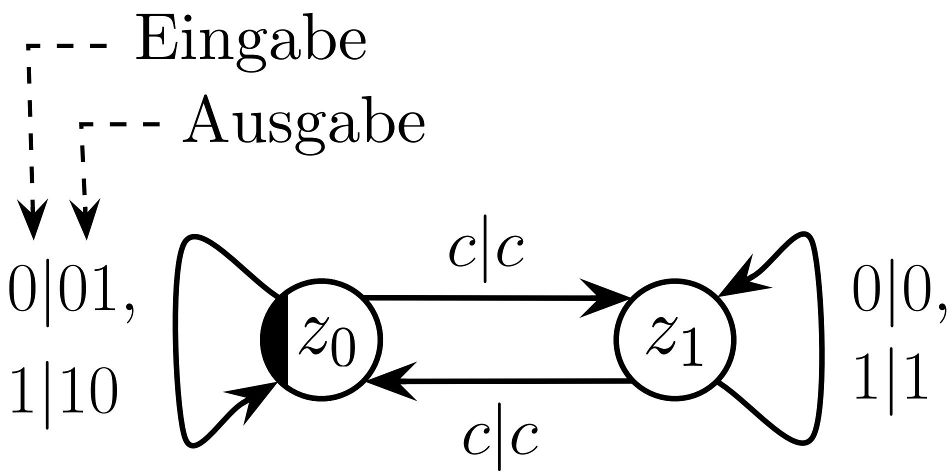
- Nummeriere alle NFAs
- Nummeriere alle Wörter
- Betrachte tabellarische Darstellung der charakteristischen Funktionen der akzeptierten Sprachen

Diagonalisierung (Zusammenf.)

- Nummeriere alle NFAs
- Nummeriere alle Wörter
- Betrachte tabellarische Darstellung der charakteristischen Funktionen der akzeptierten Sprachen
- Bilde neue Sprache, die sich von jeder in der Liste unterscheidet (jeweils in mindestens einem Wort)

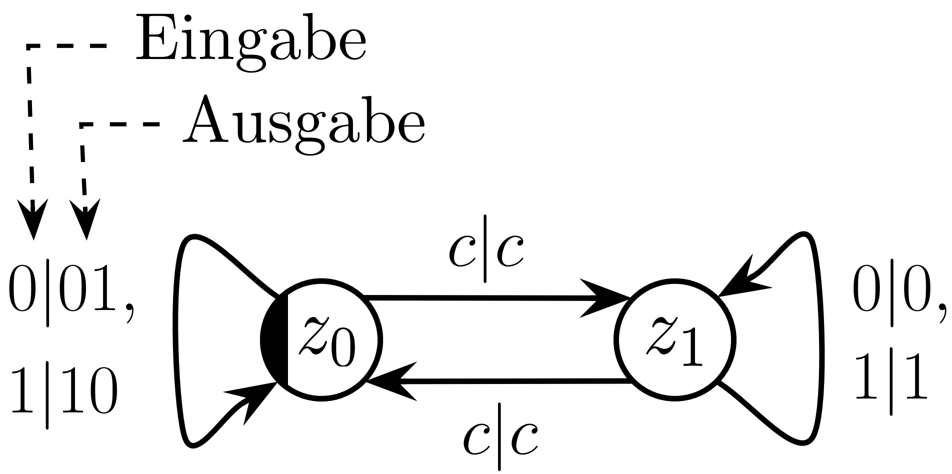
DFAs mit Ausgabe

- **Transduktor**



DFAs mit Ausgabe

- **Transduktor**

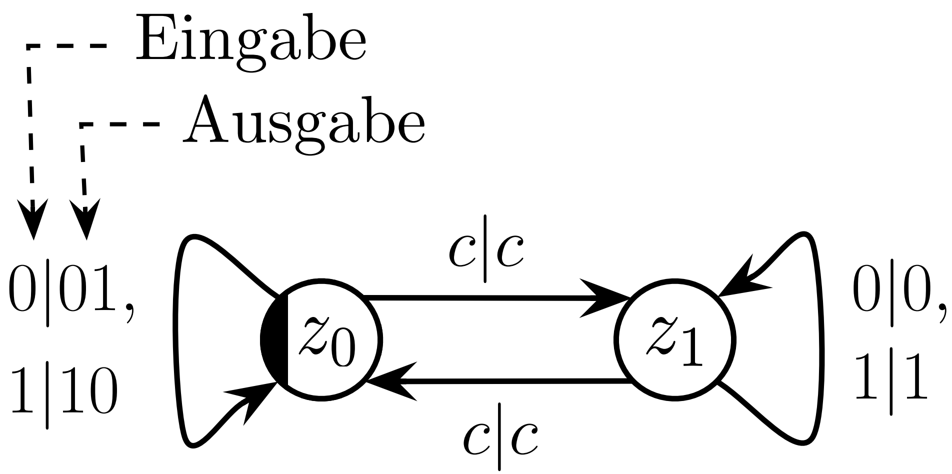


- **Moore-Automat** (Ausgabe im Zustand)

  Beispiel: Fahrkarten- und Getränkeautomaten

DFAs mit Ausgabe

- **Transduktor**



- **Moore-Automat** (Ausgabe im Zustand)

  Beispiel: Fahrkarten- und Getränkeautomaten
- **Mealy-Automat** (Ausgabe an der Kante)

  Beispiel: Kugelautomat