LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

Formale Sprachen und Komplexität
Theoretische Informatik für Studierende der Medieninformatik
Sommersemester 2024

Zentralübung 2

Prof. Dr. Jasmin Blanchette

Lehr- und Forschungseinheit für
Theoretische Informatik und Theorembeweisen

Stand: 8. Mai 2024
Basiert auf Folien von PD Dr. David Sabel

Plan für heute

1. Deterministische endliche Automaten (DFAs)
2. Nichtdeterministische endliche Automaten (NFAs)
3. Äquivalenz von DFAs, NFAs und regulären Grammatiken

1. Deterministische endliche Automaten (DFAs)

DFAs

5-Tupel $M = (Z, \Sigma, \delta, z_0, E)$ mit

	DFA
Zustandsmenge	Z
Alphabet	Σ
Übergänge	$\delta : Z \times \Sigma \to Z$
Startzustände	$z_0 \in Z$
Endzustände	$E \subseteq Z$
Übergänge für Wörter	$\widetilde{\delta} : Z \times \Sigma^* \to Z$
	$\widetilde{\delta}(z, \varepsilon) = z$
	$\widetilde{\delta}(z, aw) = \widetilde{\delta}(\delta(z, a), w)$
Akzeptierte Sprache	$L(M) = \{w \mid \widetilde{\delta}(z_0, w) \in E\}$

Darstellungsformen

- als 5-Tupel $(\{z_0, z_1, z_2\}, \{a, b\}, \delta, z_0, \{z_0, z_1\})$ mit

$$
\begin{aligned}
\delta(z_0, a) &= z_1 \qquad & \delta(z_0, b) &= z_2 \\
\delta(z_1, a) &= z_1 \qquad & \delta(z_1, b) &= z_2 \\
\delta(z_2, a) &= z_2 \qquad & \delta(z_2, b) &= z_2
\end{aligned}
$$

Darstellungsformen

- als 5-Tupel $(\{z_0, z_1, z_2\}, \{a, b\}, \delta, z_0, \{z_0, z_1\})$ mit

$$\begin{aligned} \delta(z_0, a) &= z_1 & \delta(z_0, b) &= z_2 \\ \delta(z_1, a) &= z_1 & \delta(z_1, b) &= z_2 \\ \delta(z_2, a) &= z_2 & \delta(z_2, b) &= z_2 \end{aligned}$$

- als 5-Tupel mit Übergangstabelle für δ:

δ	a	b
z_0	z_1	z_2
z_1	z_1	z_2
z_2	z_2	z_2

Darstellungsformen

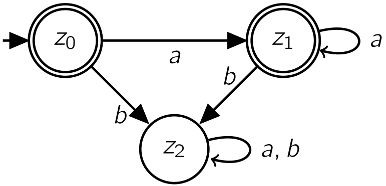
- als 5-Tupel $(\{z_0, z_1, z_2\}, \{a, b\}, \delta, z_0, \{z_0, z_1\})$ mit

$$\begin{aligned} \delta(z_0, a) &= z_1 & \delta(z_0, b) &= z_2 \\ \delta(z_1, a) &= z_1 & \delta(z_1, b) &= z_2 \\ \delta(z_2, a) &= z_2 & \delta(z_2, b) &= z_2 \end{aligned}$$

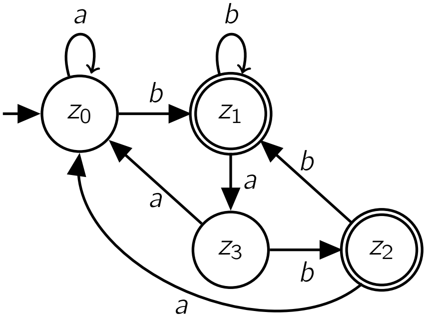
- als 5-Tupel mit Übergangstabelle für δ:

δ	a	b
z_0	z_1	z_2
z_1	z_1	z_2
z_2	z_2	z_2

- als Zustandsgraph:

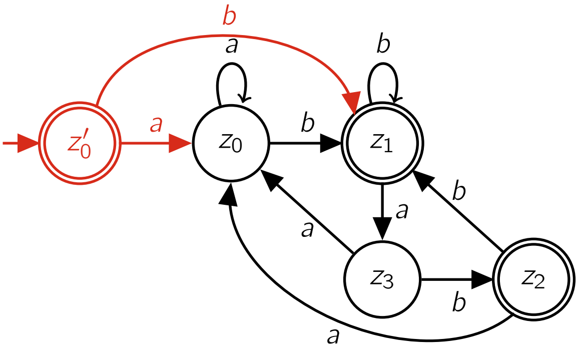


1. Quiz



- Welcher Zustand ist der Startzustand? Was sind die Endzustände?
- Wo läuft der Automat auf dem Wort *ababab* entlang?
- Erkennt der Automat ε? Wenn nein, was müsste man ändern, damit er ε erkennt? Wenn ja, was müsste man ändern, damit er ε nicht erkennt?

2. Quiz



- Welcher Zustand ist der Startzustand? Was sind die Endzustände?
- Wo entlang läuft der Automat auf dem Wort *ababab*?
- Erkennt der Automat ε? Wenn nein, was müsste man ändern, damit er ε erkennt? Wenn ja, was müsste man ändern, damit er ε nicht erkennt?

Aufgabe: DFA angeben

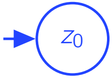
Geben Sie einen DFA an, der genau alle gültigen Uhrzeiten im 24-Stunden-Format von der Form Stunde:Minute mit Stunde $\in \{0, \ldots, 23\}$ und Minute $\in \{00, 01, \ldots, 59\}$ erkennt.

Aufgabe: DFA angeben

Antwort:

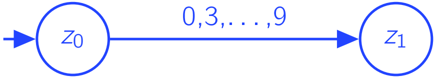
Aufgabe: DFA angeben

Antwort:



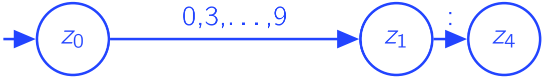
Aufgabe: DFA angeben

Antwort:



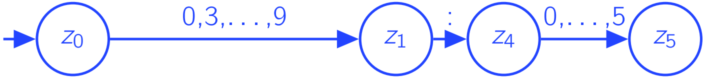
Aufgabe: DFA angeben

Antwort:



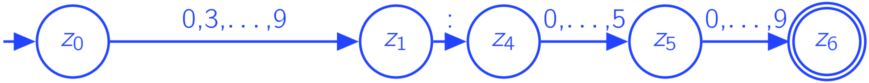
Aufgabe: DFA angeben

Antwort:



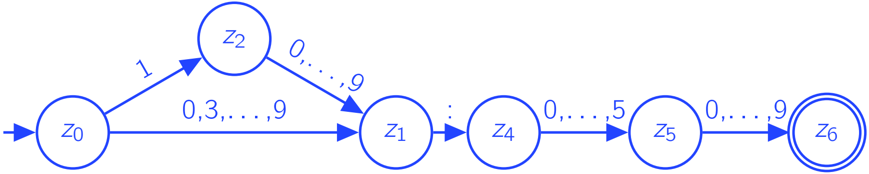
Aufgabe: DFA angeben

Antwort:



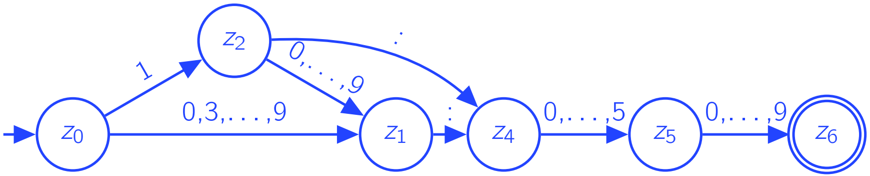
Aufgabe: DFA angeben

Antwort:



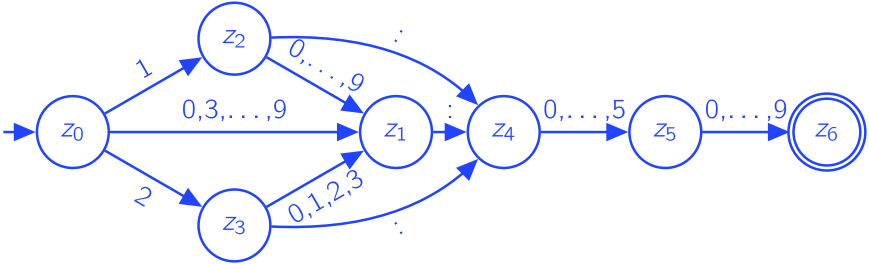
Aufgabe: DFA angeben

Antwort:



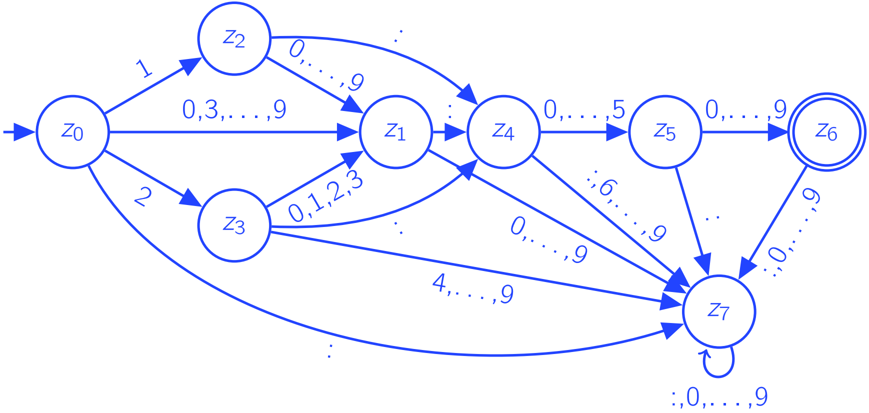
Aufgabe: DFA angeben

Antwort:



Aufgabe: DFA angeben

Antwort:

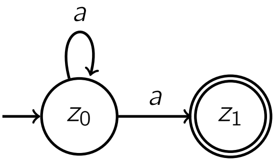


2. Nichtdeterministische endliche Automaten (NFAs)

DFAs vs. NFAs

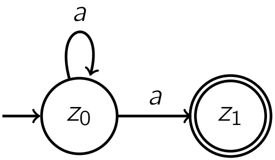
	DFA	NFA
Zustände	Z	Z
Alphabet	Σ	Σ
Übergänge	$\delta :: Z \times \Sigma \to Z$	$\delta : Z \times \Sigma \to \mathcal{P}(Z)$
Startzustände	$z_0 \in Z$	$S \subseteq Z$
Endzustände	$E \subseteq Z$	$E \subseteq Z$
Übergänge für Wörter	$\widetilde{\delta} : Z \times \Sigma^* \to Z$	$\widetilde{\delta} : \mathcal{P}(Z) \times \Sigma^* \to \mathcal{P}(Z)$
	$\widetilde{\delta}(z, \varepsilon) = z$	$\widetilde{\delta}(X, \varepsilon) = X$
	$\widetilde{\delta}(z, aw) = \widetilde{\delta}(\delta(z, a), w)$	$\widetilde{\delta}(X, aw) = \widetilde{\delta}(\bigcup_{z \in X} \delta(z, a), w)$
Akzeptierte Sprache	$L(M) = \{w \mid \widetilde{\delta}(z_0, w) \in E\}$	$L(M) = \{w \mid \widetilde{\delta}(S, w) \cap E \neq \emptyset\}$

1. Quiz



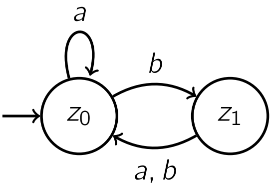
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

1. Quiz



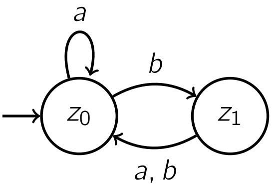
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

2. Quiz



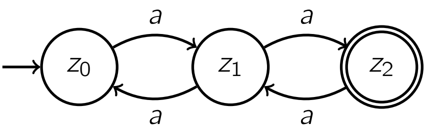
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

2. Quiz



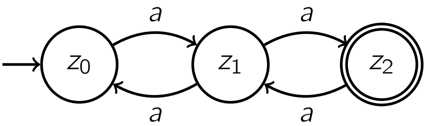
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

3. Quiz



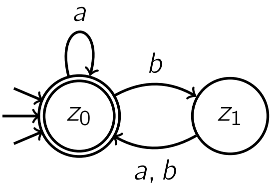
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

3. Quiz



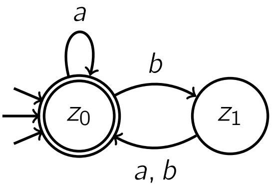
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

4. Quiz



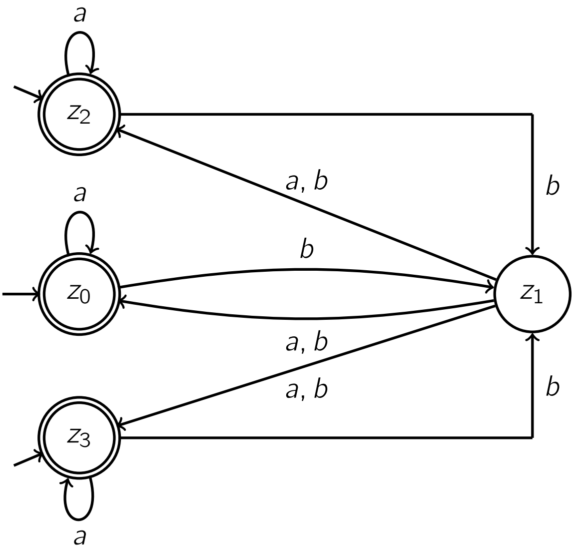
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

4. Quiz



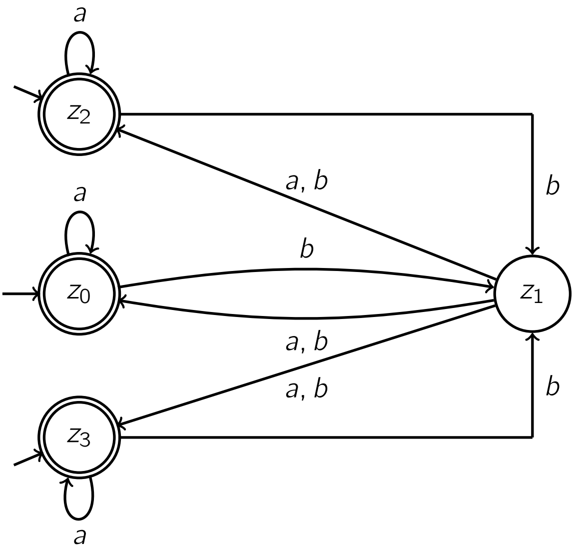
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

5. Quiz



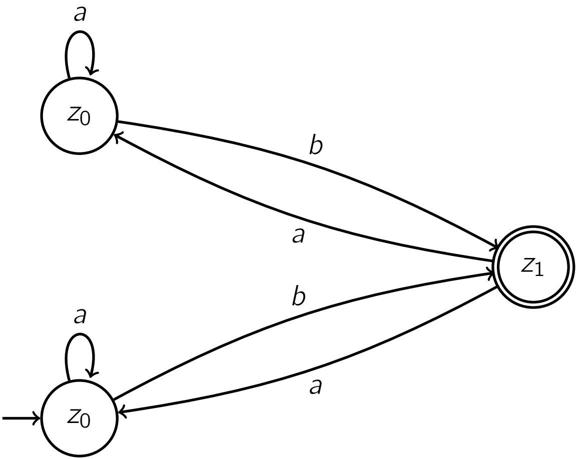
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

5. Quiz



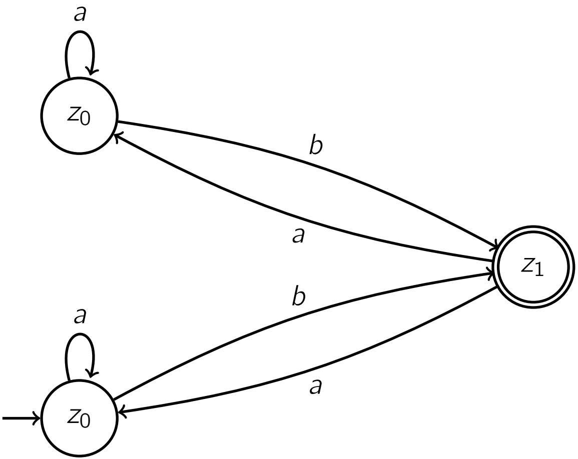
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

6. Quiz



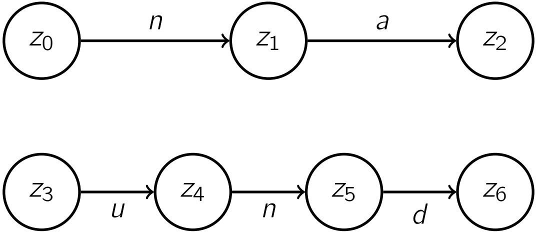
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

6. Quiz



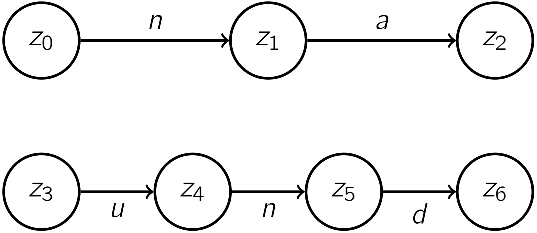
1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

7. Quiz



1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

7. Quiz



1. Ist ein DFA
2. Ist ein NFA, aber kein DFA
3. Ist weder DFA noch NFA

Aufgabe: NFAs angeben

Geben Sie einen NFA an, der Ziffernfolgen aus $\Sigma = \{0, \dots, 9\}$ verarbeitet und Wörter genau dann akzeptiert, wenn

a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam
b) die letzte Ziffer vorher nicht schon einmal vorkam.

Nutzen Sie den Nichtdeterminismus möglichst gut aus.

Aufgabe: NFAs angeben

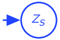
a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam

Antwort:

Aufgabe: NFAs angeben

a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam

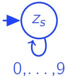
Antwort:



Aufgabe: NFAs angeben

a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam

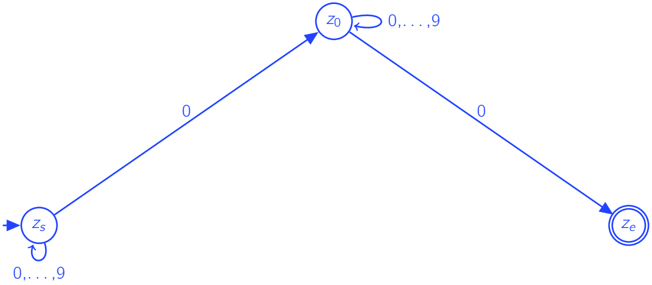
Antwort:



Aufgabe: NFAs angeben

a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam

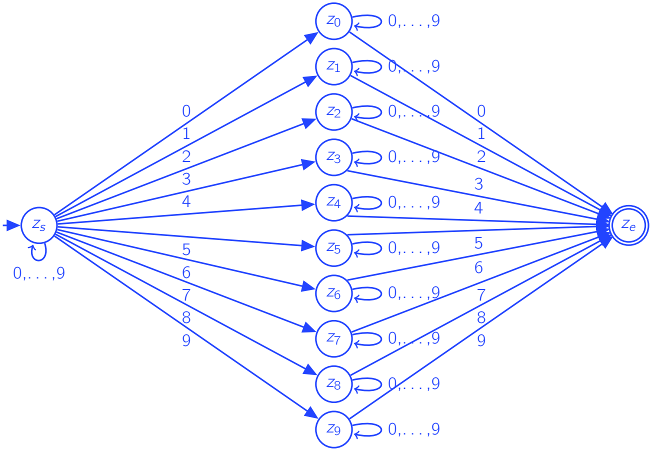
Antwort:



Aufgabe: NFAs angeben

a) die letzte Ziffer vorher schon mindestens einmal in der Folge vorkam

Antwort:



Aufgabe: NFAs angeben

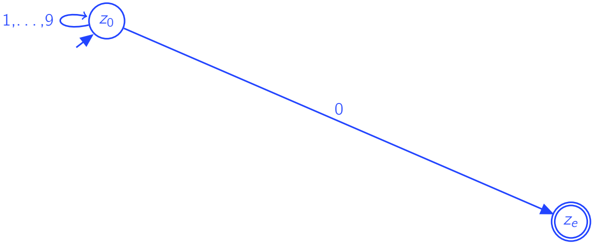
b) die letzte Ziffer vorher nicht schon einmal vorkam

Antwort:

Aufgabe: NFAs angeben

b) die letzte Ziffer vorher nicht schon einmal vorkam

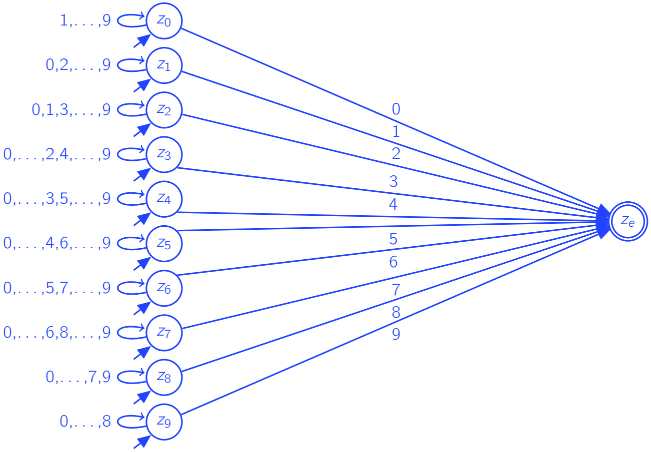
Antwort:



Aufgabe: NFAs angeben

b) die letzte Ziffer vorher nicht schon einmal vorkam

Antwort:

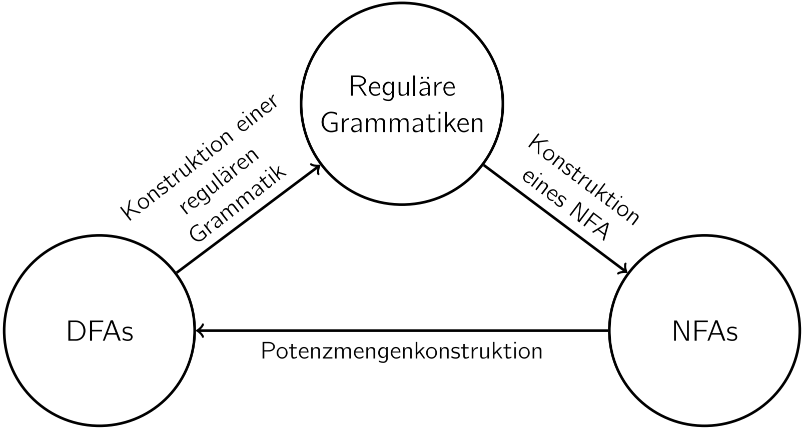


NFAs vs. NFAs mit ε-Übergängen

	NFA	NFA mit ε-Übergängen
Zustände	Z	Z
Alphabet	Σ	Σ
Übergänge	$\delta : Z \times \Sigma \to \mathcal{P}(Z)$	$\delta : Z \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Z)$
Startzustände	$S \subseteq Z$	$S \subseteq Z$
Endzustände	$E \subseteq Z$	$E \subseteq Z$
Übergänge für Wörter	$\widetilde{\delta} : \mathcal{P}(Z) \times \Sigma^* \to \mathcal{P}(Z)$	$\widetilde{\delta} : \mathcal{P}(Z) \times \Sigma^* \to \mathcal{P}(Z)$
	$\widetilde{\delta}(X, \varepsilon) = X$	$\widetilde{\delta}(X, \varepsilon) = X$
	$\widetilde{\delta}(X, aw) = \widetilde{\delta}(\bigcup_{z \in X} \delta(z, a), w)$	$\widetilde{\delta}(X, aw) = \widetilde{\delta}(\bigcup_{z \in X} \varepsilon\text{-}H\ddot{u}lle(\delta(z, a)), w)$
Akzeptierte Sprache	$L(M) = \{w \mid \widetilde{\delta}(S, w) \cap E \neq \emptyset\}$	$L(M) = \{w \mid \widetilde{\delta}(\varepsilon\text{-}H\ddot{u}lle(S), w) \cap E \neq \emptyset\}$

3. Äquivalenz von DFAs, NFAs und regulären Grammatiken

Äquivalenz der Formalismen



Beweisstruktur

Seien

- $\mathcal{D}$ = Menge der durch DFAs akzeptierten Sprachen
- $\mathcal{N}$ = Menge der durch NFAs akzeptierten Sprachen
- $\mathcal{R}$ = Menge der regulären Sprachen.

Beweisstruktur

Seien

- $\mathcal{D}$ = Menge der durch DFAs akzeptierten Sprachen
- $\mathcal{N}$ = Menge der durch NFAs akzeptierten Sprachen
- $\mathcal{R}$ = Menge der regulären Sprachen.

Wir haben gezeigt

- $\mathcal{D} \subseteq \mathcal{R}$
- $\mathcal{R} \subseteq \mathcal{N}$
- $\mathcal{N} \subseteq \mathcal{D}$.

Beweisstruktur

Seien

- $\mathcal{D}$ = Menge der durch DFAs akzeptierten Sprachen
- $\mathcal{N}$ = Menge der durch NFAs akzeptierten Sprachen
- $\mathcal{R}$ = Menge der regulären Sprachen.

Wir haben gezeigt

- $\mathcal{D} \subseteq \mathcal{R}$
- $\mathcal{R} \subseteq \mathcal{N}$
- $\mathcal{N} \subseteq \mathcal{D}$.

Daraus folgt $\mathcal{D} = \mathcal{R} = \mathcal{N}$.

Aufgabe: DFA angeben

Spielfeld:

1	2	3

Aktionen:

- ℓ und r: Wechseln um ein Feld nach ℓinks bzw. $r$echts.
- ℓ bei Feld 1 und r bei Feld 3 bleiben wirkungslos.

Aufgabe: DFA angeben

Spielfeld:

1	2	3

Aktionen:

- ℓ und r: Wechseln um ein Feld nach ℓinks bzw. $r$echts.
- ℓ bei Feld 1 und r bei Feld 3 bleiben wirkungslos.

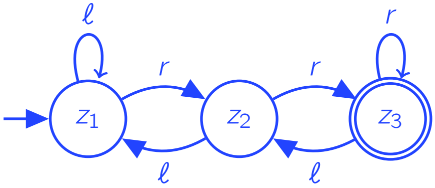
Geben Sie einen DFA über dem Alphabet $\Sigma = \{\ell, r\}$ an, der alle Folgen von Spielzügen akzeptiert, sodass der Spieler auf Feld 1 beginnt und auf Feld 3 das Spiel beendet.

Aufgabe: DFA angeben

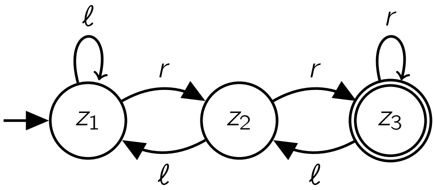
Antwort:

Aufgabe: DFA angeben

Antwort:

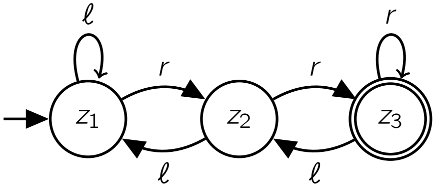


Aufgabe: Reguläre Grammatik angeben

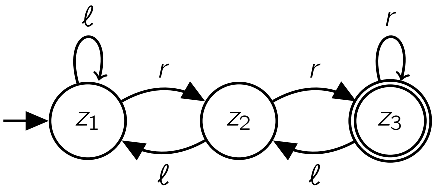


Geben Sie zum gezeigten DFA eine reguläre Grammatik an, welche die vom DFA akzeptierte Sprache erzeugt.

Aufgabe: Reguläre Grammatik angeben



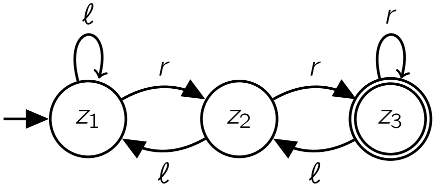
Aufgabe: Reguläre Grammatik angeben



Zur Erinnerung

Für DFA $M = (Z, \Sigma, \delta, z_0, E)$ konstruiere reguläre Grammatik $G = (V, \Sigma, P, S)$ mit
$V := Z$,
$S := z_0$,
$P := \{z_i \to az_j \mid \delta(z_i, a) = z_j\}$
$\quad \cup \{z_i \to \varepsilon \mid z_i \in E\}$.

Aufgabe: Reguläre Grammatik angeben



Zur Erinnerung

Für DFA $M = (Z, \Sigma, \delta, z_0, E)$ konstruiere reguläre Grammatik $G = (V, \Sigma, P, S)$ mit

$V := Z$,

$S := z_0$,

$$P := \{z_i \to az_j \mid \delta(z_i, a) = z_j\} \cup \{z_i \to \varepsilon \mid z_i \in E\}.$$

Antwort:

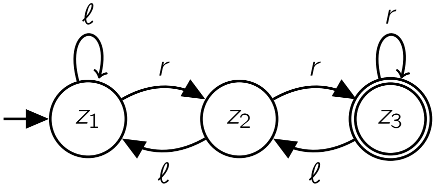
$G = (V, \Sigma, P, S)$ mit

$V = \{z_1, z_2, z_3\}$

$S = z_1$

$P =$

Aufgabe: Reguläre Grammatik angeben



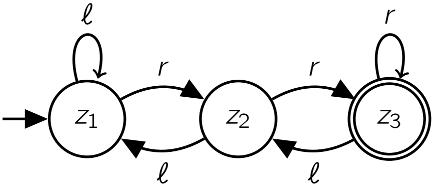
Zur Erinnerung

Für DFA $M = (Z, \Sigma, \delta, z_0, E)$ konstruiere reguläre Grammatik $G = (V, \Sigma, P, S)$ mit
$V := Z$,
$S := z_0$,
$P := \{z_i \to az_j \mid \delta(z_i, a) = z_j\}$
$\cup \{z_i \to \varepsilon \mid z_i \in E\}$.

Antwort:

$G = (V, \Sigma, P, S)$ mit
$V = \{z_1, z_2, z_3\}$
$S = z_1$
$P = \{z_1 \to \ell z_1, z_1 \to r z_2,$
$z_2 \to \ell z_1, z_2 \to r z_3,$
$z_3 \to \ell z_2, z_3 \to r z_3,$

Aufgabe: Reguläre Grammatik angeben



Zur Erinnerung

Für DFA $M = (Z, \Sigma, \delta, z_0, E)$ konstruiere reguläre Grammatik $G = (V, \Sigma, P, S)$ mit
$V := Z$,
$S := z_0$,
$P := \{z_i \to a z_j \mid \delta(z_i, a) = z_j\}$
$\quad \cup \{z_i \to \varepsilon \mid z_i \in E\}$.

Antwort:

$$\begin{aligned}
G &= (V, \Sigma, P, S) \text{ mit} \\
V &= \{z_1, z_2, z_3\} \\
S &= z_1 \\
P &= \{z_1 \to \ell z_1, z_1 \to r z_2, \\
&\qquad z_2 \to \ell z_1, z_2 \to r z_3, \\
&\qquad z_3 \to \ell z_2, z_3 \to r z_3, \\
&\qquad z_3 \to \varepsilon\}
\end{aligned}$$

Aufgabe: NFA angeben

$G = (\{S, A, D, E\}, \{\ell, r\}, P, S)$ beschreibt Spielzugfolgen, mit
$P = \{S \to \ell A \mid rS,\ A \to \ell D \mid \ell E \mid \ell \mid rS,\ D \to \ell \mid \ell D \mid rA \mid \ell E,\ E \to \ell \mid \ell D\}$.
Konstruieren Sie einen NFA M mit $L(M) = L(G)$.

Aufgabe: NFA angeben

$G = (\{S, A, D, E\}, \{\ell, r\}, P, S)$ beschreibt Spielzugfolgen, mit
$P = \{S \to \ell A \mid rS,\ A \to \ell D \mid \ell E \mid \ell \mid rS,\ D \to \ell \mid \ell D \mid rA \mid \ell E,\ E \to \ell \mid \ell D\}$.
Konstruieren Sie einen NFA M mit $L(M) = L(G)$.

Zur Erinnerung

Für reguläre Grammatik (V, Σ, P, S)
konstruiere NFA $(Z, \Sigma, \delta, \{S\}, E)$ mit

$$\begin{aligned}
Z &:= V \cup \{z_E\} \\
E &:= \{z_E\} \cup \{S \mid \text{falls } S \to \varepsilon \in P\} \\
\delta(A, a) &:= \{B \mid A \to aB \in P\} \\
&\quad \cup \{z_E \mid A \to a \in P\} \\
\delta(z_E, a) &:= \emptyset
\end{aligned}$$

Aufgabe: NFA angeben

$G = (\{S, A, D, E\}, \{\ell, r\}, P, S)$ beschreibt Spielzugfolgen, mit
$P = \{S \to \ell A \mid rS,\ A \to \ell D \mid \ell E \mid \ell \mid rS,\ D \to \ell \mid \ell D \mid rA \mid \ell E,\ E \to \ell \mid \ell D\}$.
Konstruieren Sie einen NFA M mit $L(M) = L(G)$.

Zur Erinnerung

Für reguläre Grammatik (V, Σ, P, S) konstruiere NFA $(Z, \Sigma, \delta, \{S\}, E)$ mit

$$\begin{aligned} Z &:= V \cup \{z_E\} \\ E &:= \{z_E\} \cup \{S \mid \text{falls } S \to \varepsilon \in P\} \\ \delta(A, a) &:= \{B \mid A \to aB \in P\} \\ &\quad\ \cup \{z_E \mid A \to a \in P\} \\ \delta(z_E, a) &:= \emptyset \end{aligned}$$

Antwort:

$M = (\{S, A, D, E, z_E\}, \{\ell, r\}, \delta, \{S\}, \{z_E\})$
mit

$$\begin{aligned} \delta(S, \ell) &= \{A\} & \delta(D, r) &= \{A\} \\ \delta(S, r) &= \{S\} & \delta(E, \ell) &= \{D, z_E\} \\ \delta(A, \ell) &= \{D, E, z_E\} & \delta(E, r) &= \emptyset \\ \delta(A, r) &= \{S\} & \delta(z_E, \ell) &= \emptyset \\ \delta(D, \ell) &= \{D, E, z_E\} & \delta(z_E, r) &= \emptyset \end{aligned}$$

Aufgabe: NFA angeben

Zustandsgraph: