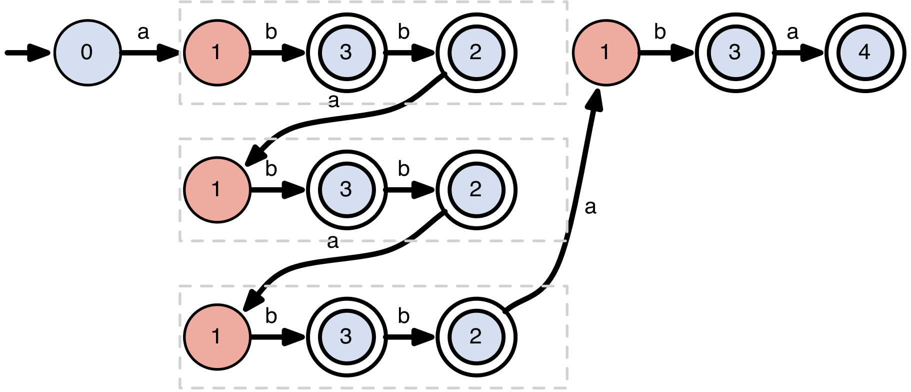
Berechenbarkeitstheorie

11. Vorlesung



Dr. Franziska Jahnke
Institut für Mathematische Logik und Grundlagenforschung
WWU Münster

Kontextfreies Pumpinglemma

- Wie können wir bestimmen, dass eine Sprache nicht kontextfrei ist?

Kontextfreies Pumpinglemma

- Wie können wir bestimmen, dass eine Sprache nicht kontextfrei ist?

Definition

Eine Sprache heißt **kf-pumpbar** gdw., es gibt ein $k \in \mathbf{N}$, sodass für alle Wörter $w \in L$ mit $|w| \geq k$ es eine Zerlegung von $w = uvxyz$ gibt, für die gilt:

1. $|vxy| \leq k$,
2. $\forall i \geq 0\colon uv^ixy^iz \in L$,
3. $|vy| > 0$.

Kontextfreies Pumpinglemma

- Wie können wir bestimmen, dass eine Sprache nicht kontextfrei ist?

Definition

Eine Sprache heißt **kf-pumpbar** gdw., es gibt ein $k \in \mathbf{N}$, sodass für alle Wörter $w \in L$ mit $|w| \geq k$ es eine Zerlegung von $w = uvxyz$ gibt, für die gilt:

1. $|vxy| \leq k$,
2. $\forall i \geq 0\colon uv^ixy^iz \in L$,
3. $|vy| > 0$.

(Anm.: Aus pumpbar folgt kf-pumpbar)

Kontextfreies Pumpinglemma

- Wie können wir bestimmen, dass eine Sprache nicht kontextfrei ist?

Definition

Eine Sprache heißt **kf-pumpbar** gdw., es gibt ein $k \in \mathbf{N}$, sodass für alle Wörter $w \in L$ mit $|w| \geq k$ es eine Zerlegung von $w = uvxyz$ gibt, für die gilt:

1. $|vxy| \leq k$,
2. $\forall i \geq 0\colon uv^ixy^iz \in L$,
3. $|vy| > 0$.

(Anm.: Aus pumpbar folgt kf-pumpbar)

Kontextfreies Pumpinglemma

$$L \in \mathsf{CFL} \quad \Rightarrow \quad L \text{ ist kf-pumpbar}$$

Beispiel

$$L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$$

Beispiel $L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$, $|vy| > 0$ gilt
$\exists i \geq 0\colon uv^ixy^iz \notin L$

Beispiel $L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$,$|vy| > 0$ gilt
$\exists i \geq 0\colon uv^ixy^iz \notin L$

- wir wählen $w \in L$ in Abhängigkeit von k, in unserem Fall $w = \mathtt{a}^k\mathtt{b}^k\mathtt{a}^k\mathtt{b}^k$
- für jede Zerteilung $w = uvxyz$ hat uv^0xy^0z immer die Form $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^*\mathtt{b}^*$

Beispiel $L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$,$|vy| > 0$ gilt
$\exists i \geq 0\colon uv^ixy^iz \notin L$

- wir wählen $w \in L$ in Abhängigkeit von k, in unserem Fall $w = \mathtt{a}^k\mathtt{b}^k\mathtt{a}^k\mathtt{b}^k$
- für jede Zerteilung $w = uvxyz$ hat uv^0xy^0z immer die Form $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^*\mathtt{b}^*$

Fall A liegen beide Pumpstellen im ersten r, erhalte ich $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^k\mathtt{b}^k$, wobei der Anfang ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Beispiel $L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$,$|vy| > 0$ gilt $\exists i \geq 0\colon uv^ixy^iz \notin L$

- wir wählen $w \in L$ in Abhängigkeit von k, in unserem Fall $w = \mathtt{a}^k\mathtt{b}^k\mathtt{a}^k\mathtt{b}^k$
- für jede Zerteilung $w = uvxyz$ hat uv^0xy^0z immer die Form $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^*\mathtt{b}^*$

Fall A liegen beide Pumpstellen im ersten r, erhalte ich $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^k\mathtt{b}^k$, wobei der Anfang ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Fall B liegen beide Pumpstellen im letzten r, erhalte ich $\mathtt{a}^k\mathtt{b}^k\mathtt{a}^*\mathtt{b}^*$, wobei das Ende ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Beispiel $L = \{rr \mid r \in \{\mathtt{a},\mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$,$|vy| > 0$ gilt $\exists i \geq 0\colon uv^ixy^iz \notin L$

- wir wählen $w \in L$ in Abhängigkeit von k, in unserem Fall $w = \mathtt{a}^k\mathtt{b}^k\mathtt{a}^k\mathtt{b}^k$
- für jede Zerteilung $w = uvxyz$ hat uv^0xy^0z immer die Form $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^*\mathtt{b}^*$

Fall A liegen beide Pumpstellen im ersten r, erhalte ich $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^k\mathtt{b}^k$, wobei der Anfang ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Fall B liegen beide Pumpstellen im letzten r, erhalte ich $\mathtt{a}^k\mathtt{b}^k\mathtt{a}^*\mathtt{b}^*$, wobei das Ende ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Fall C benutzen die Pumpstellen Teile aus beiden rs, erhalte ich $\mathtt{a}^k\mathtt{b}^*\mathtt{a}^*\mathtt{b}^k$, wobei das Mittelteil ungleich $\mathtt{b}^k\mathtt{a}^k$ ist

Beispiel $L = \{rr \mid r \in \{\mathtt{a}, \mathtt{b}\}^*\}$

$\forall k \in \mathbf{N}\ \exists w \in L\colon (|w| \geq k)\ \forall uvxyz = w$ mit $|vxy| \leq k$,$|vy| > 0$ gilt $\exists i \geq 0\colon uv^ixy^iz \notin L$

- wir wählen $w \in L$ in Abhängigkeit von k, in unserem Fall $w = \mathtt{a}^k\mathtt{b}^k\mathtt{a}^k\mathtt{b}^k$
- für jede Zerteilung $w = uvxyz$ hat uv^0xy^0z immer die Form $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^*\mathtt{b}^*$

Fall A liegen beide Pumpstellen im ersten r, erhalte ich $\mathtt{a}^*\mathtt{b}^*\mathtt{a}^k\mathtt{b}^k$, wobei der Anfang ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Fall B liegen beide Pumpstellen im letzten r, erhalte ich $\mathtt{a}^k\mathtt{b}^k\mathtt{a}^*\mathtt{b}^*$, wobei das Ende ungleich $\mathtt{a}^k\mathtt{b}^k$ ist

Fall C benutzen die Pumpstellen Teile aus beiden rs, erhalte ich $\mathtt{a}^k\mathtt{b}^*\mathtt{a}^*\mathtt{b}^k$, wobei das Mittelteil ungleich $\mathtt{b}^k\mathtt{a}^k$ ist

- in jedem der Fälle ist uv^0xy^0z nicht aus L

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 | S_2$

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 | S_2$
  $\to L(G_3) = L_1 \cup L_2$

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 | S_2$
  $\to L(G_3) = L_1 \cup L_2$
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 S_2$

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 | S_2$
  $\to L(G_3) = L_1 \cup L_2$
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 S_2$
  $\to L(G_3) = L_1 \circ L_2$

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \rightarrow S_1 | S_2$
  $\rightarrow L(G_3) = L_1 \cup L_2$
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \rightarrow S_1 S_2$
  $\rightarrow L(G_3) = L_1 \circ L_2$
- definiere $G_3 = (V_1, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 plus den Regeln $S_3 \rightarrow S_3 S_3 | S_1 | \varepsilon$

Abschluss von CFL

Satz 13

CFL ist abgeschlossen unter Vereinigung, Konkatenation, und Kleene Stern.

- sei $G_1 = (V_1, \Sigma, R_1, S_1)$ kfG für L_1 und $G_2 = (V_2, \Sigma, R_2, S_2)$ kfG für L_2 (V_1 und V_2 disjunkt)
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1|S_2$
  $\to L(G_3) = L_1 \cup L_2$
- definiere $G_3 = (V_1 \cup V_2, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 und R_2 plus der Regel $S_3 \to S_1 S_2$
  $\to L(G_3) = L_1 \circ L_2$
- definiere $G_3 = (V_1, \Sigma, R_3, S_3)$ mit S_3 neues Symbol und R_3 besteht aus den Regeln R_1 plus den Regeln $S_3 \to S_3 S_3 | S_1 | \varepsilon$
  $\to L(G_3) = L_1^*$

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n, m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n, m \geq 0\}$

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n, m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n, m \geq 0\}$
- L_1 und L_2 sind kontextfrei

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n, m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n, m \geq 0\}$
- L_1 und L_2 sind kontextfrei

  Grammatik für L_1:
$$\begin{array}{lcl} S & \rightarrow & AC \\ A & \rightarrow & \mathtt{a}A\mathtt{b} \mid \varepsilon \\ C & \rightarrow & \mathtt{c}C \mid \varepsilon \end{array}$$

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n, m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n, m \geq 0\}$
- L_1 und L_2 sind kontextfrei

  Grammatik für L_1:
  $$\begin{array}{lcl} S & \rightarrow & AC \\ A & \rightarrow & \mathtt{a}A\mathtt{b} \mid \varepsilon \\ C & \rightarrow & \mathtt{c}C \mid \varepsilon \end{array}$$
- $L_1 \cap L_2 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$ ist bekanntlich nicht kontextfrei

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n,m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n,m \geq 0\}$
- L_1 und L_2 sind kontextfrei

  Grammatik für L_1:
  $$\begin{array}{lcl} S & \to & AC \\ A & \to & \mathtt{a}A\mathtt{b} \mid \varepsilon \\ C & \to & \mathtt{c}C \mid \varepsilon \end{array}$$
- $L_1 \cap L_2 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$ ist bekanntlich nicht kontextfrei
- wäre CFL abgeschlossen unter Komplement würde gelten, $\forall L_1, L_2 \in \mathsf{CFL} : \overline{\bar{L}_1 \cup \bar{L}_2} \in \mathsf{CFL}$

Satz 14

CFL ist nicht abgeschlossen unter Schnitt und Komplement.

- $L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m \mid n, m \geq 0\}$, $L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n \mid n, m \geq 0\}$
- L_1 und L_2 sind kontextfrei

  Grammatik für L_1:
  $$\begin{array}{lcl} S & \rightarrow & AC \\ A & \rightarrow & \mathtt{a}A\mathtt{b} \mid \varepsilon \\ C & \rightarrow & \mathtt{c}C \mid \varepsilon \end{array}$$
- $L_1 \cap L_2 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$ ist bekanntlich nicht kontextfrei
- wäre CFL abgeschlossen unter Komplement würde gelten, $\forall L_1, L_2 \in \mathsf{CFL}: \overline{\bar{L}_1 \cup \bar{L}_2} \in \mathsf{CFL}$
- da $\overline{\bar{L}_1 \cup \bar{L}_2} = L_1 \cap L_2$ wäre CFL dann abgeschlossen unter Schnitt $\rightarrow$ Widerspruch

Erinnerung Vorlesungsübersicht

1. Reguläre Sprachen
   - endliche Automaten
   - Nichtdeterminismus
   - reguläre Ausdrücke
2. Kontext-freie Sprachen
   - Grammatiken
   - Kellerautomaten
3. Berechenbarkeitstheorie
   - Turingmaschinen
   - Unentscheidbarkeit
4. Komplexitätstheorie
   - P und NP
   - NP-vollständige Problem

Erinnerung Vorlesungsübersicht

1. Reguläre Sprachen
   - endliche Automaten
   - Nichtdeterminismus
   - reguläre Ausdrücke
2. Kontext-freie Sprachen
   - Grammatiken
   - Kellerautomaten
3. Berechenbarkeitstheorie
   - Turingmaschinen
   - Unentscheidbarkeit
4. Komplexitätstheorie
   - P und NP
   - NP-vollständige Problem

3. Kapitel

Berechenbarkeitstheorie

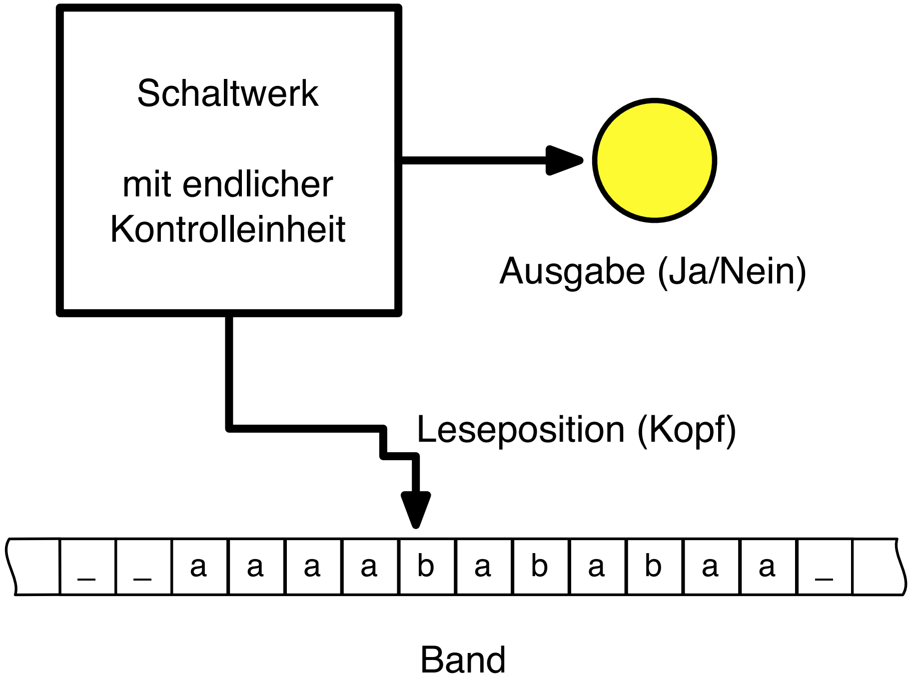
Turingmaschinen

Turingmaschinen

- **Ziel:** starkes Berechnungsmodell, welches die Arbeitsweise eines idealisierten *Computers* nachbildet
- eingeführt 1936 von Alan Turing

Turingmaschinen

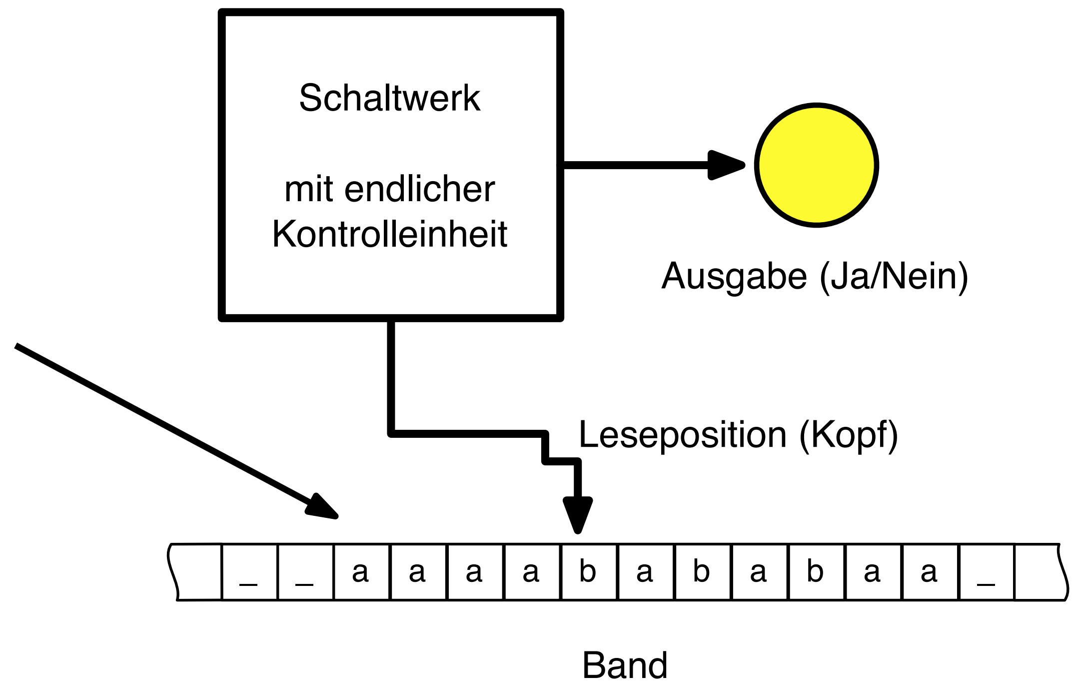
- **Ziel:** starkes Berechnungsmodell, welches die Arbeitsweise eines idealisierten *Computers* nachbildet
- eingeführt 1936 von Alan Turing



Turingmaschinen

- **Ziel:** starkes Berechnungsmodell, welches die Arbeitsweise eines idealisierten *Computers* nachbildet
- eingeführt 1936 von Alan Turing

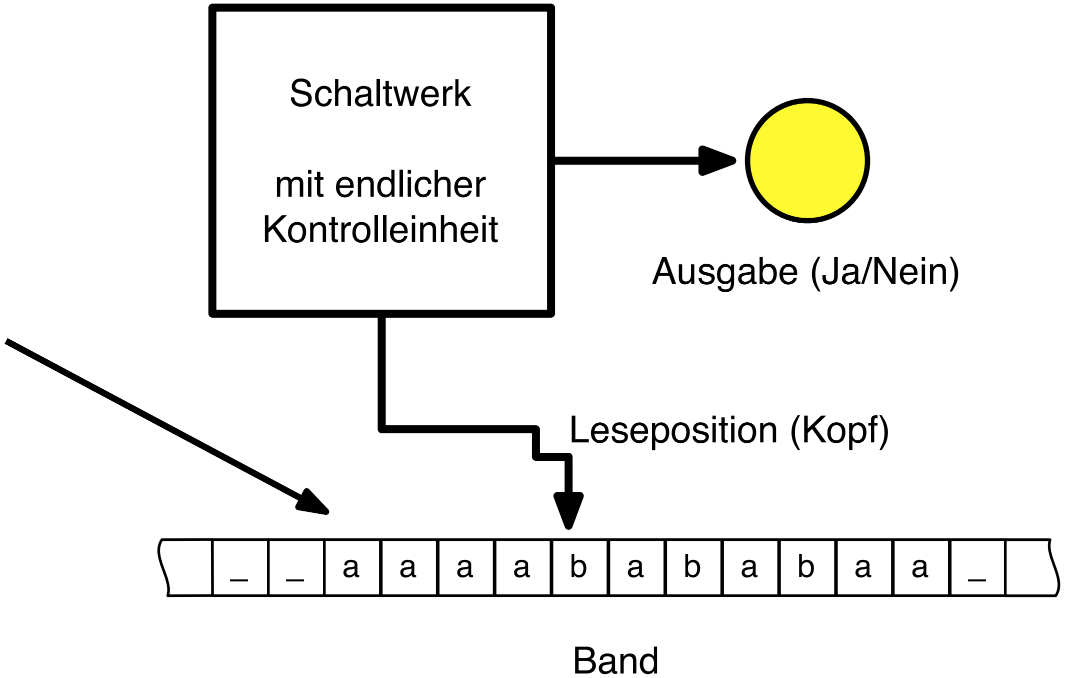
Kopf kann auf dem unbeschränktem Schreib-Lese-Band frei nach links und rechts bewegt werden werden (in 1er Schritten)



Turingmaschinen

- **Ziel:** starkes Berechnungsmodell, welches die Arbeitsweise eines idealisierten *Computers* nachbildet
- eingeführt 1936 von Alan Turing

Kopf kann auf dem unbeschränktem Schreib-Lese-Band frei nach links und rechts bewegt werden werden (in 1er Schritten)



- Übergänge vom Zeichen an Kopfposition abhängig

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta \colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta \colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand
- Bandinhalt $=$ **beschriebene Zeichen** des Bandes$=$ endliches Wort

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand
- Bandinhalt $=$ **beschriebene Zeichen** des Bandes$=$ endliches Wort
- Default Symbol des Bandes heißt Blank-Symbol, $\square$

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand
- Bandinhalt $=$ **beschriebene Zeichen** des Bandes$=$ endliches Wort
- Default Symbol des Bandes heißt Blank-Symbol, $\square$
- Schreibweise für Konfigurationen: $w_1 q w_2$

Definition

Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta \colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand
- Bandinhalt $=$ **beschriebene Zeichen** des Bandes$=$ endliches Wort
- Default Symbol des Bandes heißt Blank-Symbol, $\square$
- Schreibweise für Konfigurationen: $w_1 q w_2$

  $\rightarrow$ Bandinhalt $w_1 w_2$, Kopf hinter w_1, Zustand q

Definition

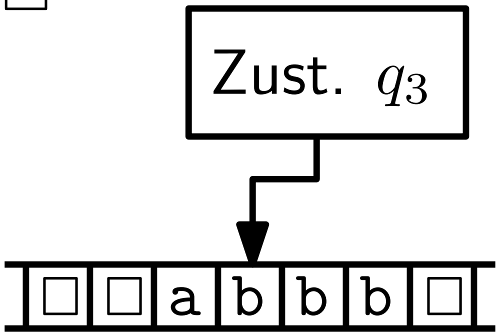
Eine **Turingmachine (TM)** ist ein 7-Tupel $(Q, \Sigma, \Gamma, \delta, q_0, q_A, q_V)$, mit:

- der Zustandsmenge Q,
- dem Eingabealphabet Σ, dem Arbeitsalphabet Γ,
- der Übergangsfunktion $\delta \colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$,
- $q_0, q_A, q_V \in Q$ mit $q_A \neq q_V$.

Konfiguration einer TM

- Konfiguration $=$ Bandinhalt $+$ Kopfposition $+$ aktueller Zustand
- Bandinhalt $=$ **beschriebene Zeichen** des Bandes$=$ endliches Wort
- Default Symbol des Bandes heißt Blank-Symbol, $\square$
- Schreibweise für Konfigurationen: $w_1 q w_2$
  $\to$ Bandinhalt $w_1 w_2$, Kopf hinter w_1, Zustand q

Beispiel: $\mathtt{a}q_3\mathtt{bbb}$



Überführung von Konfigurationen

Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

Überführung von Konfigurationen

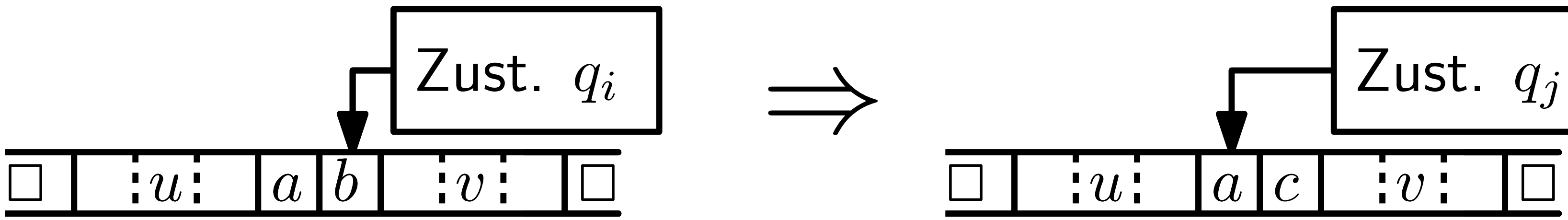
Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

① $ua\, q_i\, bv$ geht über nach $u\, q_j\, acv$, gdw.
$\delta(q_i, b) = (q_j, c, L)$

Überführung von Konfigurationen

Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

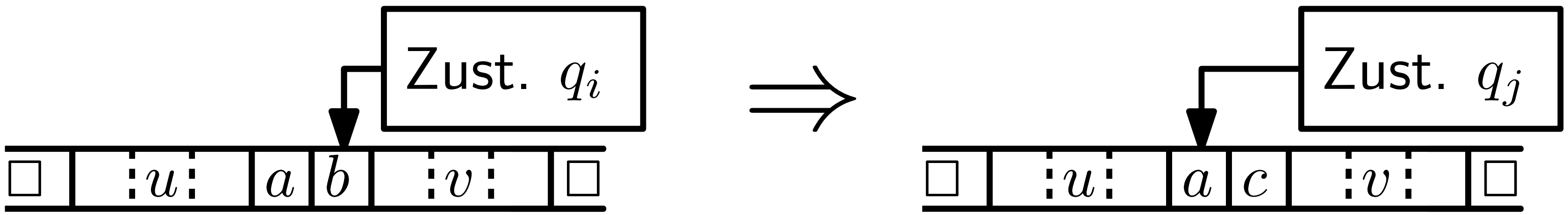
① $ua\, q_i\, bv$ geht über nach $u\, q_j\, acv$, gdw.
$\delta(q_i, b) = (q_j, c, L)$



Überführung von Konfigurationen

Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

① $ua\, q_i\, bv$ geht über nach $u\, q_j\, acv$, gdw.
$\delta(q_i, b) = (q_j, c, L)$

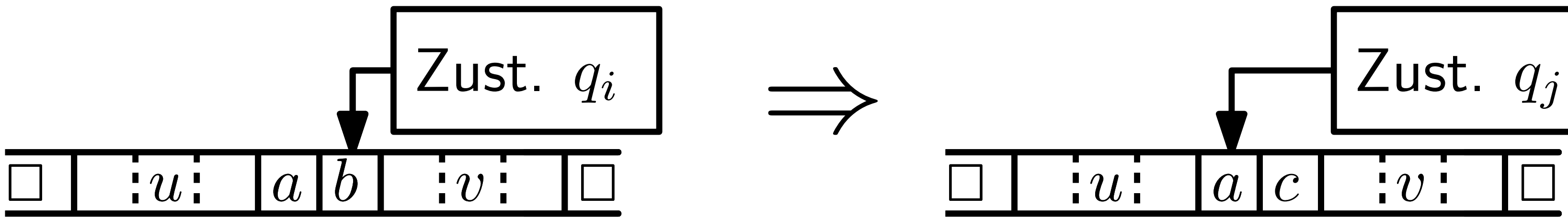


② $ua\, q_i\, bv$ geht über nach $uac\, q_j\, v$, gdw.
$\delta(q_i, b) = (q_j, c, R)$

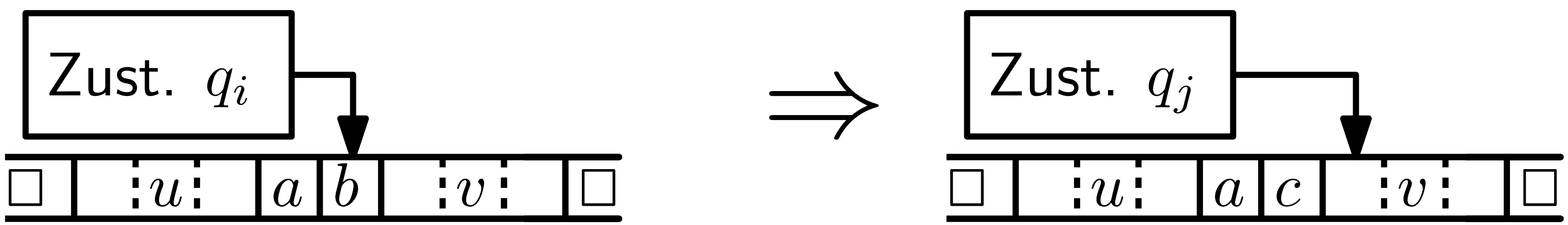
Überführung von Konfigurationen

Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

① $ua\,q_i\,bv$ geht über nach $u\,q_j\,acv$, gdw.
$\delta(q_i, b) = (q_j, c, L)$



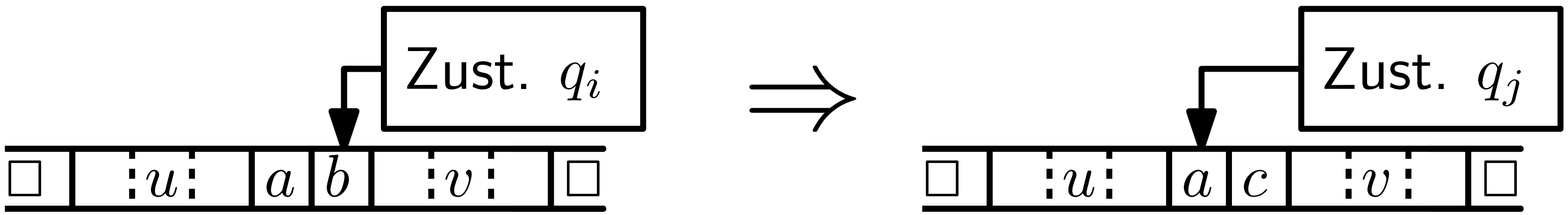
② $ua\,q_i\,bv$ geht über nach $uac\,q_j\,v$, gdw.
$\delta(q_i, b) = (q_j, c, R)$



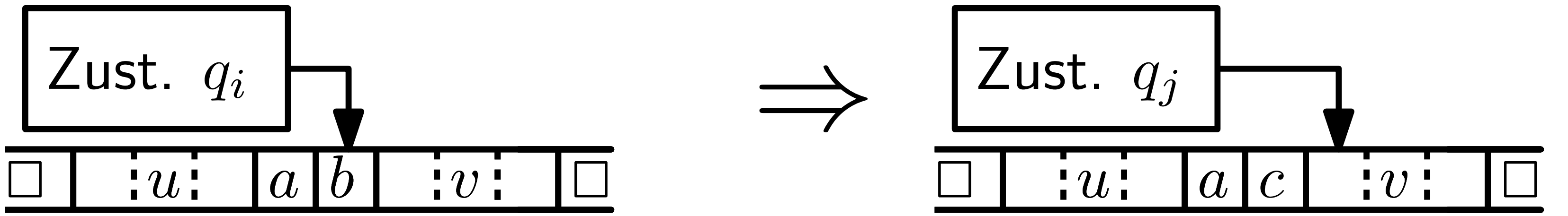
Überführung von Konfigurationen

Sei $u, v \in \Gamma^*$ und $a, b, c \in \Gamma$, sowie $q_i, q_j \in Q$

① $ua\, q_i\, bv$ geht über nach $u\, q_j\, acv$, gdw.
$\delta(q_i, b) = (q_j, c, L)$



② $ua\, q_i\, bv$ geht über nach $uac\, q_j\, v$, gdw.
$\delta(q_i, b) = (q_j, c, R)$



③ $u\, q_A v$ heißt akzeptierend, $u\, q_V v$ verwerfend, und $q_0\, w$ Startkonfiguration (für Eingabe w)

Akzeptanz bei Turingmaschinen

Akzeptanz bei Turingmaschinen

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1$: C_i geht über nach C_{i+1}
3. C_n akzeptierend

Akzeptanz bei Turingmaschinen

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

Lauf der TM

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1$: C_i geht über nach C_{i+1}
3. C_n akzeptierend

Akzeptanz bei Turingmaschinen

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

Lauf der TM

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1$: C_i geht über nach C_{i+1}
3. C_n akzeptierend

Def.:

Eine TM M **erkennt** die Sprache $\{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

Lauf der TM

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1\colon\ C_i$ geht über nach C_{i+1}
3. C_n akzeptierend

Def.:

Eine TM M **erkennt** die Sprache $\{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$

2 Möglichkeiten für Nichtakzeptanz

Akzeptanz bei Turingmaschinen

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

Lauf der TM

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1$: C_i geht über nach C_{i+1}
3. C_n akzeptierend

Def.:

Eine TM M **erkennt** die Sprache $\{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$

2 Möglichkeiten für Nichtakzeptanz

① Lauf führt zu einer Konfiguration mit q_V

Akzeptanz bei Turingmaschinen

Definition

Eine Turingmaschine M **akzeptiert ein Wort** $w \in \Sigma^*$ genau dann wenn eine Folge von nicht-verwerfenden Konfigurationen $C_1, \ldots, C_n$ existiert mit:

1. $C_1 = q_0\, w$ (Startkonfiguration)
2. $\forall i = 1, \ldots, n-1$: C_i geht über nach C_{i+1}
3. C_n akzeptierend

Lauf der TM

Def.:

Eine TM M **erkennt** die Sprache $\{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$

2 Möglichkeiten für Nichtakzeptanz

① Lauf führt zu einer Konfiguration mit q_V

② Lauf erreicht weder q_V noch q_A (M stoppt nicht)

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_A, q_V\}, \Gamma = \{\mathtt{a}, \mathtt{X}, \$, \Box\}$

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_A, q_V\}, \Gamma = \{\mathtt{a}, \mathtt{X}, \$, \square\}$
- Angabe von δ in Diagrammform

Beispiel $L = \{\texttt{a}^{2^n} \in \{\texttt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_A, q_V\}, \Gamma = \{\texttt{a}, \texttt{X}, \$, \square\}$
- Angabe von δ in Diagrammform

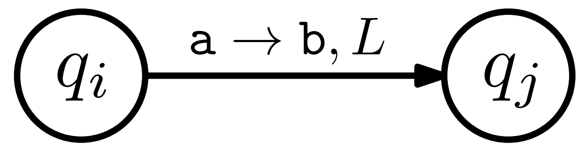
$\delta(q_i, \texttt{a}) = (q_j, \texttt{b}, L)$

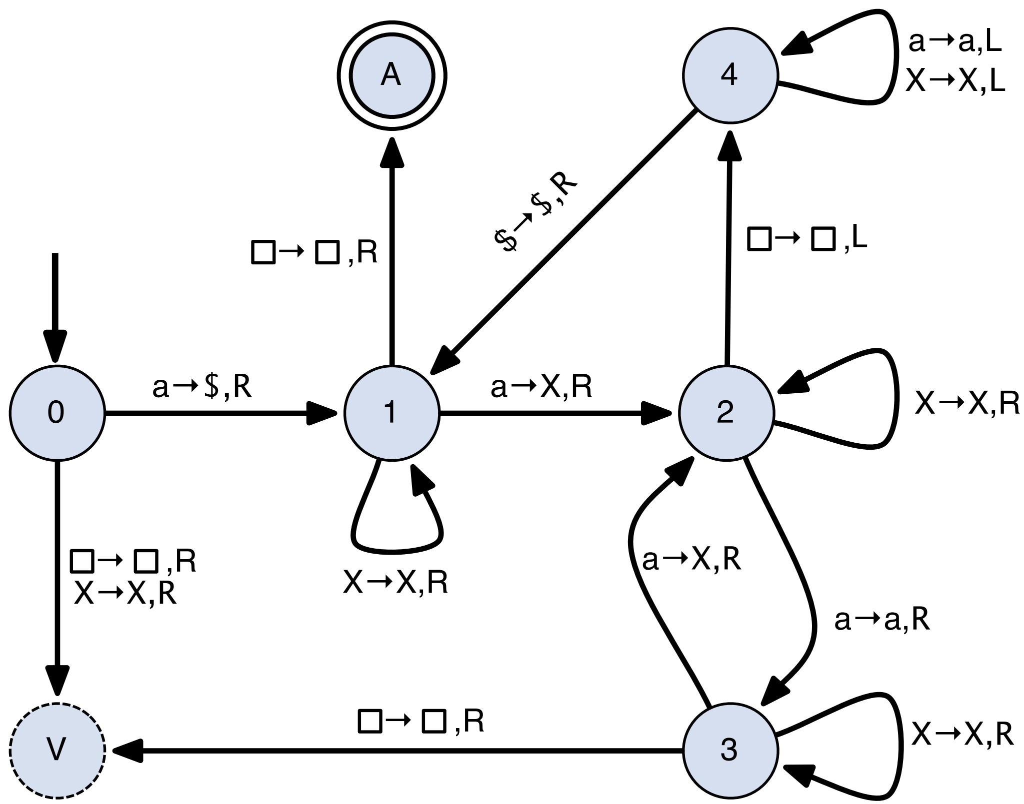
$$q_i \xrightarrow{\texttt{a} \to \texttt{b}, L} q_j$$

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_A, q_V\}, \Gamma = \{\mathtt{a}, \mathtt{X}, \$, \square\}$
- Angabe von δ in Diagrammform

$\delta(q_i, \mathtt{a}) = (q_j, \mathtt{b}, L)$



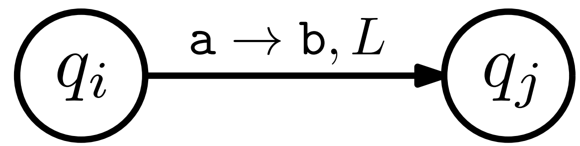


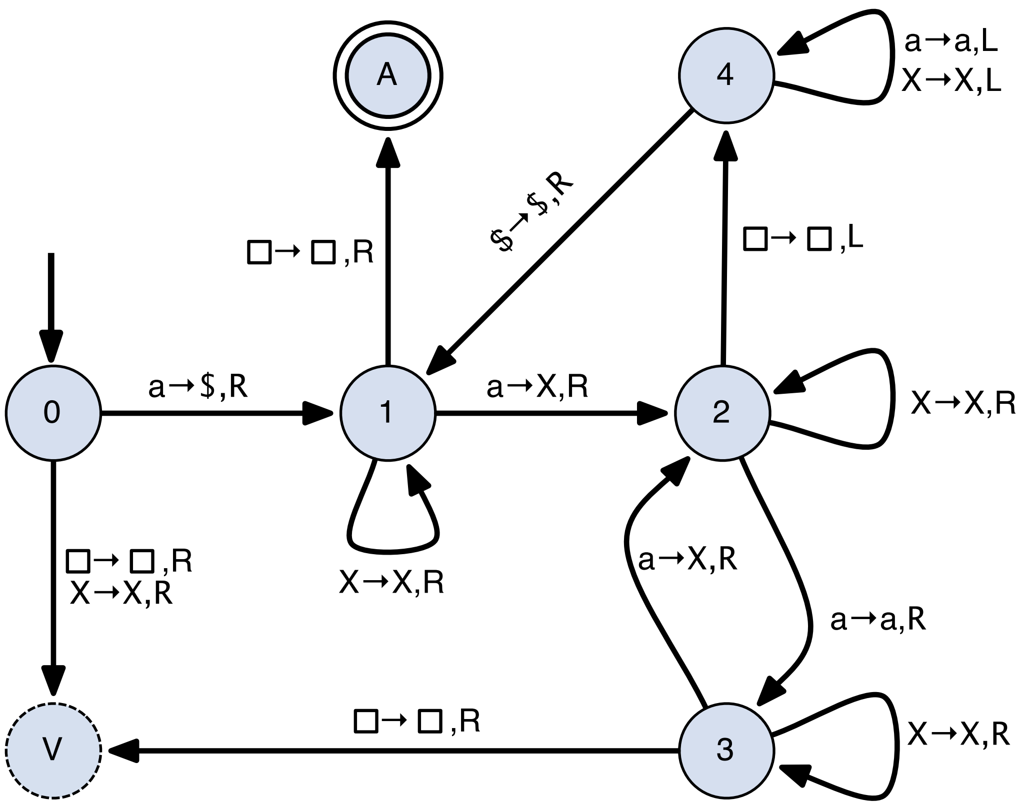
1. Markiere Anfang durch \$
2. war das das einzige Zeichen, dann akzeptiere
3. lösche jedes 2. a auf dem Band
   - 3.1. bei Misserfolg (ungerade Anzahl) verwerfe
   - 3.2. bei Erfolg (gerade Anzahl) gehe zum \$ nach links
4. Goto 2.

Beispiel $L = \{\mathtt{a}^{2^n} \in \{\mathtt{a}\}^* \mid n \geq 0\}$

- **Idee:** k ist 2er-Potenz ungleich 1 $\iff$ $k/2$ ist 2er-Potenz
- Kann ich die Eingabe wiederholt halbieren?
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_A, q_V\}, \Gamma = \{\mathtt{a}, \mathtt{X}, \$, \square\}$
- Angabe von δ in Diagrammform

$\delta(q_i, \mathtt{a}) = (q_j, \mathtt{b}, L)$





1. Markiere Anfang durch \$ q_0
2. war das das einzige Zeichen, dann akzeptiere q_1
3. lösche jedes 2. a auf dem Band q_2, q_3
   - 3.1. bei Misserfolg (ungerade Anzahl) verwerfe
   - 3.2. bei Erfolg (gerade Anzahl) gehe zum \$ nach links q_4
4. Goto 2.

Angabe von TM Programmen

Angabe von TM Programmen

① Diagrammform

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegbben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

② Modulbeschreibung

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

② Modulbeschreibung

- Programm wird verbal beschrieben

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

② Modulbeschreibung

- Programm wird verbal beschrieben
- Abfolge von Modulen (=Befehlssequenzen)

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

② Modulbeschreibung

- Programm wird verbal beschrieben
- Abfolge von Modulen (=Befehlssequenzen)
- Module sind problemlos als TM-Teilprogramm formulierbar
- Bsp.:
  Gehe ans Ende des Bandes links und ersetze jede zweite 1 durch eine 0

Angabe von TM Programmen

① Diagrammform

- Programm wird präzise angegben
- Spezifikation von Q und Γ
- Übergangsfunktion δ wird in Diagrammform angegeben
- Fehlende Übergänge im Diagramm würden nach q_V überführen

② Modulbeschreibung

- Programm wird verbal beschrieben
- Abfolge von Modulen (=Befehlssequenzen)
- Module sind problemlos als TM-Teilprogramm formulierbar
- Bsp.:
  Gehe ans Ende des Bandes links und ersetze jede zweite 1 durch eine 0
- Bsp.:
  Merke das aktuelle Zeichen im Zustand, gehe bis zum ersten $ nach rechts und ersetze $ durch das gemerkte Zeichen