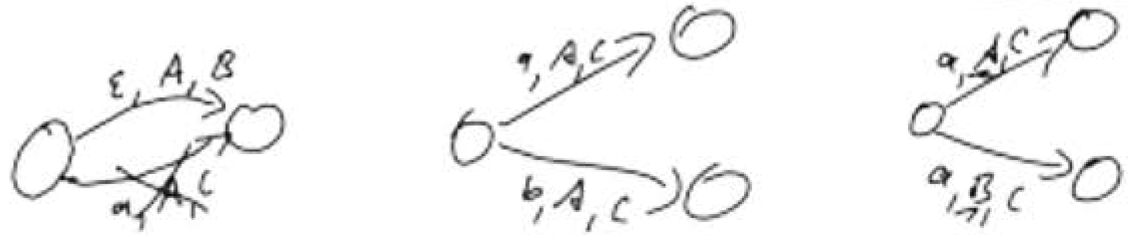
2.3.6 Deterministisch kontextfreie Sprachen

Def Ein <u>det. Kellerautomat</u> (DPDA) ist ein PDA_E, bei dem für alle $z \in Z$, $a \in \Sigma$, $A \in \Gamma$ gilt:

$$|\delta(z,a,A)| + |\delta(z,\varepsilon,A)| \le 1.$$

<u>Bsp:</u>



<u>Bem:</u> Bei det. Kellerautomaten ist Akzeptanz durch leeren Keller nicht äquivalent mit Akzeptanz durch Endzustand.

<u>Grund:</u> Enthält die Sprache Wörter, deren echte Präfixe auch zur Sp. gehören.

<u>Bsp:</u> $L = \{(ab)^n \mid n \geq 1\}$ $\quad L = \{ab, abab, ababab, \dots\}$

Def: Eine Sprache ist **deterministisch kontextfrei**, wenn sie von einem DPDA akzeptiert werden.

Bsp: $L = \{ w \$ w^R \mid w \in \{a,b\}^* \}$ ist det. kf

$L = \{ w w^R \mid w \in \{a,b\}^* \}$ ist nicht det. kf

Satz Die det. kf. Sprachen sind unter Komplement abgeschlossen.

Bew: Gegeben DPDA $M = (Z, \Sigma, \Gamma, \delta, z_0, \#, E)$, konstruiere $M' = (Z, \Sigma, \Gamma, \delta, z_0, \#, \underline{Z - E})$. Dann gilt $N(M') = \overline{N(M)}$

$x \in N(M')$ $\underline{\text{gdw}}$ $(z_0, x, \#) \underline{\vdash^*} (z, \varepsilon, \alpha)$ $\alpha \in \Gamma^*$, $z \in Z - E$

$\underline{\text{gdw}}$ $(z_0, x, \#) \vdash^* (z, \varepsilon, \alpha)$ $z \notin E$

$\underline{\text{gdw}}$ $x \notin N(M)$

$\square$

<u>Bem</u>: Das impliziert, dass kf Spr $\supsetneq$ det kf Spr.

<u>Satz</u> Die det. kf. Spr. sind nicht unter Schnitt und Vereinigung abgeschlossen.

<u>Bew</u>:

<u>Schnitt</u>: $L_1 = \{ a^n b^n c^m \mid n, m \geq 1 \}$ $L_2 = \{ a^n b^m c^m \mid n, m \geq 1 \}$ sind beide det. kf und $L_1 \cap L_2$ ist nicht kf.

<u>Vereinigung</u>: Wäre sie unter V. abgeschlossen, dann würde wg. $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$ gelten, dass die det. kf. Spr. auch unter Schnitt abgeschlossen wäre ↯ □

<u>Bem</u>: Die det. kf. Spr. sind gerade die sogenannten LR(k)-Sprachen.

<u>Bem</u>: LR(k)-Spr. können in linearer Zeit erkannt werden.

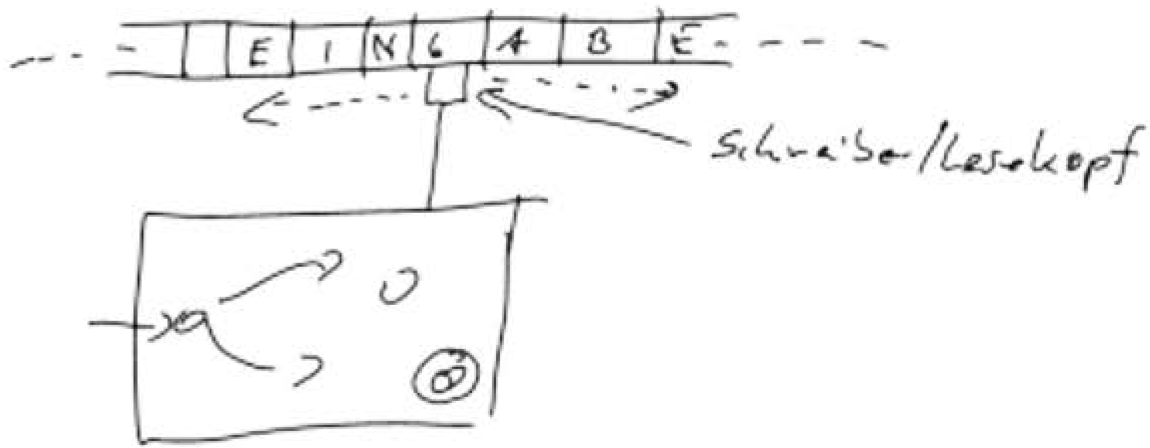
2.4 Typ 1 und Typ 0 Sprachen

Automatenmodell?

→ Turing-Maschine (Alan M. Turing, 1912–1954)

→ zur Beschreibung allgemeiner Berechnungsprozesse

Intuitiv:



Def

Eine Turing-Maschine (TM) ist 7-Tupel

$$M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$$

- Z Zustandsmenge
- Σ Eingabealphabet
- $\Gamma \supseteq \Sigma$ Arbeitsalphabet
- $\delta : Z \times \Gamma \to Z \times \Gamma \times \{L, R, N\}$ (det. TM)

  $\delta : Z \times \Gamma \to \mathcal{P}(Z \times \Gamma \times \{L, R, N\})$ (nicht-det TM)

  Überführungsfkt.
- z_0 Startzustand
- $\square \in \Gamma - \Sigma$ Blanksymbol
- $E \subseteq Z$ Endzustände

Informal:

$$\delta(z, a) \stackrel{\ni}{=} (z', b, x) \quad \text{bedeutet}$$

wenn M im Zustand z ist und a liest, dann gehe in den Zustand z', schreibe b auf das Band und bewege den Kopf in Richtung $x \in \{L, R, N\}$ (links, rechts, stehen bleiben).

Def Eine Konfiguration einer TM ist ein Wort

$k \in \Gamma^* Z \Gamma^*$

Bsp

Zustand von M

$$\underbrace{a_1 a_2 \dots a_k}_{\text{links vom R/W-Kopf}} \; z_i \; a_{k+1} \; \underbrace{a_{k+2} \dots a_n}_{\text{rechts vom Kopf}}$$

Zeichen unter dem Kopf

Def

Die Nachfolgekonfigurationsrelation $\vdash$ für det. TMs ist wie folgt def.:

$$a_1 \dots a_{m-1} a_m z b_1 b_2 \dots b_n \vdash \begin{cases} a_1 a_2 \dots a_m z' c b_2 \dots b_n, & m \geq 0, n \geq 1 \\ \quad \text{falls } \delta(z, b_1) = (z', c, N) \\ a_1 a_2 \dots a_m c z' b_2 \dots b_n, & m \geq 0, \underline{n \geq 2} \\ \quad \text{falls } \delta(z, b_1) = (z', c, R) \\ a_1 a_2 \dots a_m c z' \square, & m \geq 0, n = 1 \\ \quad \text{falls } \delta(z, b_1) = (z', c, R) \\ a_1 a_2 \dots a_{m-1} z' a_m c b_2 \dots b_n, & m \geq 1, n \geq 1 \\ \quad \text{falls } \delta(z, b_1) = (z', c, L) \\ z' \square c b_2 \dots b_n, & m = 0, n \geq 1 \\ \quad \text{falls } \delta(z, b_1) = (z', c, L) \end{cases}$$

Für nicht-det. TMs entsprechend

Def Die von einer TM M akz. Spr. ist wie folgt def.:

$$T(M) = \{ x \in \Sigma^* \mid z_0 x \vdash^* \alpha z \beta,\ \alpha, \beta \in \Gamma^*,\ z \in E \}$$

Für Typ 1-Spr. brauchen wir eine Beschränkung des Arbeitsbandes → "Linear Bounded Automaton" (LBA).

Wie merkt die TM, dass sie am rechten Rand ist?

Erweiterung von Σ zu $\Sigma' = \Sigma \cup \underline{\{ \hat{a} \mid a \in \Sigma \}}$.

Eingabe: $a_1 a_2 \dots a_{n-1} \hat{a}_n$

Def Eine nicht-det. TM heißt linear beschränkt, wenn für alle $a_1 a_2 \dots a_{n-1} a_n \in \Sigma^*$ und alle Konfigurationen $\alpha z \beta$ mit $z_0 \underline{a_1 a_2 \dots a_{n-1} \hat{a}_n} \vdash^* \alpha z \beta$ gilt $|\alpha \beta| = n$. Die akz. Spr. ist wie folgt def.:

$$T(M) = \{ a_1 a_2 \dots a_{n-1} a_n \in \Sigma^* \mid z_0 a_1 a_2 \dots a_{n-1} \hat{a}_n \vdash^* \alpha z \beta,\ \alpha, \beta \in \Gamma^*,\ z \in E \}$$

<u>Satz</u> Die von LBAs akz. Spr. sind genau die Typ1-Sprachen.

<u>Bew</u>:

($\Leftarrow$) Sei L eine Typ1-Spr mit $L = L(G)$, $G = (V, \Sigma, P, S)$.

Prinzipieller Aufbau der TM:

Schleife:

1) TM wählt eine Regel der Grammatik $u \to v$ aus
2) sucht die TM ein beliebiges Vorkommen von v
3) Die TM ersetzt v durch u (ggfs. mit Linksverschiebung des Strings rechts von v)
4) Falls nur noch S auf dem Band steht, gehe in Endzustand