kontextfreie Sprachen: Normalformen

Zur Erinnerung: Kontextfreie Sprachen sind diejenigen, die von Grammatiken erzeugt werden, die auf allen linken Regelseiten nur je ein Nichtterminalsymbol haben.

Aufgrund der Bedingung für Sprachen vom Typ 1, die linke Seite jeder Regel muss kürzer sein als die rechte, kann in diesen Sprachen ϵ nicht abgeleitet werden.

Es gilt aber: jede kontextfreie Grammatik G, die Regeln der Form $A \rightarrow \epsilon$ enthält, kann in eine äquivalente Grammatik G' ohne solche Regeln umgeformt werden, so dass gilt: $L(G) = L(G') \cup \{\epsilon\}$.

Dazu wird V zerlegt in $V_1 = \{A \mid A \Rightarrow^* \epsilon\}$ und $V_0 = V \backslash V_1$.

Dann werden alle Regeln der Form $A \rightarrow \epsilon$ aus P entfernt und für alle Regeln der Form $B \rightarrow xAy$ die Regel $B \rightarrow xy$ eingefügt.

kontextfreie Sprachen: Normalformen

<u>Definition</u>:

Eine epsilon-freie, kontextfreie Grammatik G ist in *Chomsky-Normalform, CNF*, wenn alle Regeln entweder die Form $A \rightarrow BC$ oder $A \rightarrow a$ haben.

Zu jeder kontextfreien Grammatik gibt es eine äquivalente in Chomsky-Normalform.

Umformung:

- Eliminierung von Regeln der Form $A \rightarrow B$:

  Falls es eine Menge von Nichtterminalen gibt mit $B_1 \rightarrow B_2, B_2 \rightarrow ..., B_k \rightarrow B1$, ersetze alle B_i durch B.

  Dann Nummerierung der Nichtterminalen so, dass gilt: aus $A_i \rightarrow A_j$ folgt $i < j$, also $V = \{A_1, A_2, ...A_n\}$. Jetzt wird diese Folge von Nichtterminalen von hinten abgearbeitet: für $k = n - 1, ...1$ werden alle Regeln der Form $A_k \rightarrow A_{k'}, k' > k$ eliminiert; für die Regeln $A_{k'} \rightarrow x_1 \mid ... \mid x_l$ werden die Regeln $A_k \rightarrow x_1 \mid ... \mid x_l$ hinzugefügt.

kontextfreie Sprachen: Normalformen

- Eliminierung von Regeln der Form $A \rightarrow xay$:

  Für jedes Terminal a wird ein neues Nichtterminal B sowie einer Regel $B \rightarrow a$ eingefügt. Dann wird jedes Vorkommen von a auf einer rechten Seite (länger als 1) durch B ersetzt.

- bleibt noch: Regeln der Form $A \rightarrow B_1 B_2 ... B_k, k > 2$:

  Für jede dieser Regeln werden neue Variablen $C_2, ..., C_{k-1}$ und Regeln $A \rightarrow B_1 C_2, C_2 \rightarrow B_2 C_3, ...,$ $C_{k-1} \rightarrow B_{k-1} B_k$ eingefügt.

kontextfreie Sprachen: Kellerautomaten, PDAs

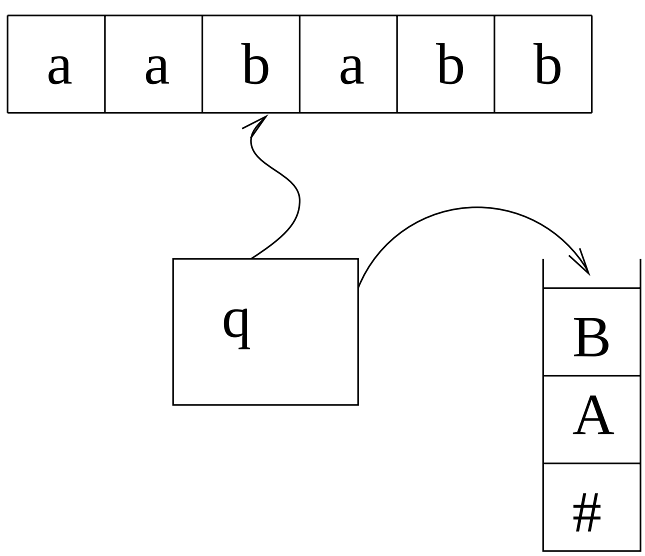
Sei $L = \{a_1, .., a_n\$a_n, .., a_1 \mid a_i \in \Sigma\}$.

Problem:

DFAs und NFAs haben kein “Gedächtnis”, beim Lesen von $ kann nur auf die Zustände zurückgegriffen werden, aber die sind endlich.

Lösung: Kellerautomaten:

Ein Kellerautomat liest ebenfalls ein Eingabeband von links nach rechts, aber auch in jedem Schritt das obere Symbol eines “Kellers” (*stack*). In Abhängigkeit vom Zustand, vom Eingabesymbol und vom Kellersymbol wird in den neuen Zustand übergegangen und ein neues Kellersymbol geschrieben.



Kellerautomaten

Definition:

Ein *nichtdeterministischer Kellerautomat* (*Pushdown Automata, PDA*) ist ein 6-Tupel $M = (K, \Sigma, \Gamma, \Delta, q_0, \#)$ mit:

K eine endliche Menge von Zuständen,

Σ ein Alphabet, das Eingabealphabet,

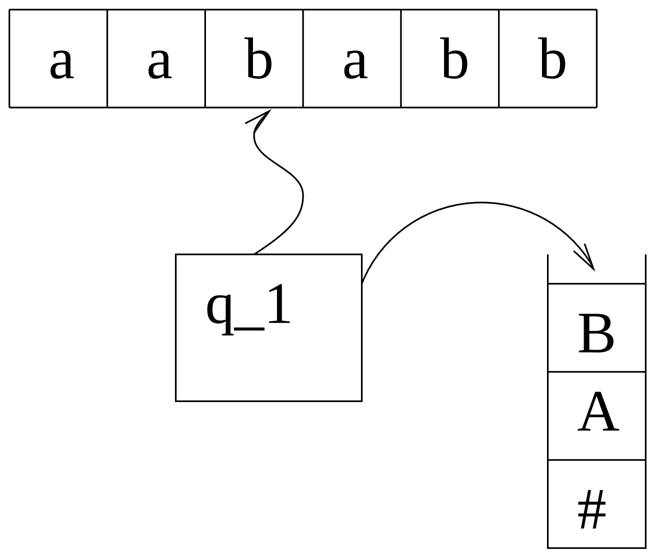
Γ ein Alphabet, das Kelleralphabet,

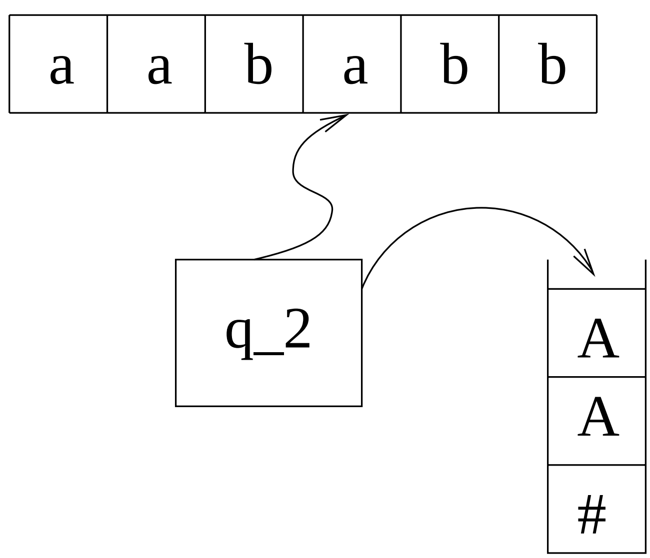
Δ eine Relation von $K \times (\Sigma \cup \{\epsilon\}) \times \Gamma$ nach $K \times \Gamma$, die Zustandsübergangsrelation,

$q_0 \in K$ der Startzustand,

$\#$ das unterste Kellerzeichen.

Kellerautomaten





$$\Delta((q_1, b, B), (q_2, A))$$

Kellerautomaten

Situation eines PDA:

Position des Lesekopfes, Zustand und Kellerinhalt: (x, q, y, z).

“erzeugt in einem Schritt”:

$(x, q, y, z) \vdash (x', q', y', z')$ gdw. es $a \in \Sigma \cup \{\epsilon\}$, $\gamma, \delta \in \Gamma$ und $w \in \Gamma^*$ gibt mit: $x' = xa, y = ay', z = \gamma w, z' = \delta w$ und $\Delta((q, a, \gamma), (q', \delta))$.

“erzeugt”, $\vdash^*$: die reflexive, transitive Hülle von $\vdash$

Akzeptanz eines Eingabestrings durch einen PDA:

1. Die Eingabe ist vollständig abgearbeitet,

2. der Keller ist leer.

(Alternativ kann man auch Endzustände definieren...)

Ein PDA M akzeptiert eine Sprache L, wenn gilt: für alle $x \in L$ gibt es ein q mit: $(\epsilon, q_0, x, \epsilon) \vdash_M^* (x, q, \epsilon, \epsilon)$

kontextfreie Sprachen und Kellerautomaten

Ein endlicher Automat ist ein Kellerautomat mit einem leeren Kelleralphabet.

Reguläre Sprachen können also auch durch Kellerautomaten erkannt werden.

Ansonsten gilt: eine Sprache ist (maximal) kontextfrei gdw. wenn sie von einem nichtdeterministischen Kellerautomaten erkannt wird.

Deterministischer Kellerautomat: statt Relation Δ Funktion δ, statt leerem Keller definierte Endzustände.

Die deterministischen kontextfreien Sprachen sind eine echte Teilmenge der kontextfreien...

Abschlusseigenschaften von kontextfreien Sprachen

Die Klasse ist *abgeschlossen* unter

- Vereinigung
- Konkatenation
- Stern (Kleene),

nicht aber unter

- Schnitt,

  Gegenbeispiel:

  $L_1 = \{a^i b^i c^j \mid i, j \geq 0\}$ und $L_2 = \{a^k b^l c^l \mid k, l \geq 0\}$,
  $L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 0\}$: nicht kontextfrei.

- Komplement

  Beweis durch Widerspruch:

  seien zwei Sprachen L_1 und L_2 über Σ gegeben. Wenn die Komplemente $L_1' = \Sigma^* \backslash L_1$ und $L_2' = \Sigma^* \backslash L_2$ kontextfrei wären, dann auch ihre Vereinigung $L_1' \cup L_2'$. Deren Komplement ist laut deMorgan'schem Gesetz aber gerade $L_1 \cap L_2$ und nicht notwendig kontextfrei.

weiter: kontextfreie Sprachen

Aber: Der Schnitt einer regulären Sprache mit einer kontextfreien ist kontextfrei.

Anwendung:

Beweis, dass eine Sprache L nicht kontextfrei ist:

finde L_r regulär und L_n bewiesenermassen nicht kf mit $L = L_r \cap L_n$.

Es gibt *unentscheidbare* Probleme im Zusammenhang mit kontextfreien Grammatiken, z.B. ob zwei Grammatiken dieselbe Sprache erzeugen...

(das heisst nicht, dass das <u>nie</u> entschieden werden, sondern nur, dass es keine allgemeingültige Rechenvorschrift dafür geben kann!)

Anwendung: Natürliche Sprache kontextfrei?

Eher nicht...

Methode: Nachweis von kreuzweisen Abhängigkeiten wie in $L = \{xx \mid x \in \{a, b\}^*\}$.

z.Bsp. im Schweizerdeutschen und Holländischen:

Wir wollen dem Kind dem Nachbarn den Garten umzugraben zu helfen erlauben.

...