Formális nyelvek és a fordítóprogramok alapjai

6. előadás

Előadó: Nagy Sára, mesteroktató
Algoritmusok és Alkalmazásaik Tanszék

Emlékeztető

Chomsky féle hierarchia:

$\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0$

G=(N,T,P,S) grammatika 2-es típusú, ha szabályai

$\mathbf{A} \rightarrow \mathbf{u}$ alakúak, ahol $A \in N$, $u \in (N \cup T)^*$

Ezeket nevezzük ***környezetfüggetlen*** grammatikáknak. Ilyenekkel írható le a programozási nyelvek szintaxisa.

2-típusú grammatikák normál formája

Definíció:

Egy $G=(N,T,P,S)$ környezetfüggetlen grammatikát **Chomsky normálformájúnak** mondunk, ha szabályai

- **$A \to a$,** ahol $A \in N$ és $a \in T$ vagy
- **$A \to BC$** alakúak, ahol $A,B,C \in N$.
- $S \to \varepsilon$, de ekkor S nem fordul elő egyetlen szabály jobboldalán sem.

Chomsky normál forma

Tétel:

Minden **környezetfüggetlen** grammatikához megkonstruálható egy vele ekvivalens **Chomsky normálformájú** grammatika.

Megjegyzés:

- A 2-es típusú grammatikák Chomsky normálformára hozásának algoritmusa nem a tananyag része.
- Chomsky normálformájú grammatikákhoz megadható olyan elemző program, amely $O(n^3)$ időben eldönti a szóproblémát (Cocke-Younger-Kasami algoritmus).
- Bizonyos állítások bizonyítását elég elvégezni a normálformájú grammatikákra.

Bar-Hillel lemma (pumpáló lemma)

Minden L környezetfüggetlen nyelvhez megadható két nyelvtől függő természetes szám p és q úgy, hogy

$\forall$ u $\in$ L szóra, ha $\ell(u) > p$, akkor u felírható

$u = vxwyz$

alakban, ahol v, x, w, y, z $\in T^*$ és

- $\ell(xwy) \leq q$,
- $xy \neq \varepsilon$,
- **$vx^iwy^iz \in L$, $\forall\, i \geq 0$ esetén.**

Megjegyzés: A lemmát nem bizonyítjuk, de a bizonyításhoz szükséges, hogy a 2-es típusú nyelvekhez létezik Chomsky-normálformájú grammatika.

Következmény

Van olyan nyelv, amely nem környezetfüggetlen.

Például **L = { $a^n b^n c^n$ | n > 0}** $\notin \mathcal{L}_2$.

Tegyük fel indirekt, hogy ∃ p,q a Bar-Hillel lemmának megfelelő konstansok.

Legyen k>p és k>q is. Ekkor $u=a^k b^k c^k$ > p.

A lemma szerint az u szó vxwyz alakban felbontható kell legyen, úgy, hogy $\ell(xwy) \leq q < k$ és x és y párhuzamosan beiterálható.

De ekkor xy-ban nem lehet mindhárom betűből, így vwz nem lehet eleme L-nek, ami ellentmondás.

Szóprobléma eldöntése

Tétel:

Minden $G=(N,T,P,S)$ környezetfüggetlen grammatika esetében eldönthető,hogy egytetszőleges $u \in T^*$ szó benne van-e a G grammatika által generált nyelvben vagy sem.

Másképpen $u \in L(G)$ igaz-e?

Szóprobléma eldöntése

<u>Bizonyítás:</u>

Az általánosság megszorítása nélkül feltehetjük, hogy G Chomsky normálformában van.

Az, hogy az üres szó benne van-e nyelveben az attól függ, hogy van-e $S \to \varepsilon$ szabály G-ben.

Ha u nem az üres szó, akkor $k = \ell(u) - 1 + \ell(u)$ lépésben levezethető kell legyen G-ben.

Mivel a k lépésben levezethető szavak halmaza véges, ezért eldönthető, hogy u benne van-e ebben a halmazban.

Veremautomata

Definíció:

$A = (Z, Q, T, \delta, z_0, q_0, F)$ rendezett hetest veremautomatának nevezzük, ahol

- Z a verem szimbólumok ábécéje,
- Q az állapotok nem üres véges halmaza,
- T az input szimbólumok ábécéje,
- $\delta: Z \times Q \times (T \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Z^* \times Q)$ leképezés az állapot-átmeneti függvény, ahol δ véges részhalmazokba képez,
- $z_0 \in Z$ a kezdő veremszimbólum,
- $q_0 \in Q$ a kezdőállapot,
- $F \subseteq Q$ elfogadó állapotok halmaza.

Veremautomata állapot-átmenete

Egy lépésben mindig kell egy jelet olvasni a verem tetejéről és csak egy jelet lehet elérni. Az input szalagról is egy jelet lehet olvasni, de nem kötelező.

Megváltoztatható az automata aktuális állapota, illetve a verem teteje. Egy lépésben egy egész sorozatot is beírhatunk a verembe.

Példák:

- **δ(#,q,a) = {(#a,q)}**

Jelentése: Ha # van a verem tetején és ‚a' betű jön az inputon, akkor tegyük be ‚a'-t a verembe. Ne változtassunk az állapoton.

- **δ(#,q,a) = {(ε,q)}**

Jelentése: Ha # van a verem tetején és ‚a' betű jön az inputon, akkor töröljük #-t a veremből. Ne változtassunk az állapoton.

- **δ(#,q,a) = {(#,r)}**

Jelentése: Ha # van a verem tetején és ‚a' betű jön az inputon, akkor ne változtassuk a verem tartalmát. Viszont váltsunk állapotot.

- **δ(#,q,ε) = {(#bb,r)}**

Jelentése: Ha # van a verem tetején és nem olvasunk az inputról, akkor tegyünk a verembe két ‚b' betűt és váltsunk állapotot is.

Példa

Legyen $L=\{a^n b^n \mid n \geq 1\}$.

Ezt a nyelvet felismerő veremautomata:

$A = (\{\#,a\}, \{q_0, q_1, q_2\}, \{a,b\}, \delta, \#, q_0, \{q_2\})$

$\delta(\#,q_0,a) = \{(\#a,q_0)\}$

$\delta(a,q_0,a) = \{(aa,q_0)\}$

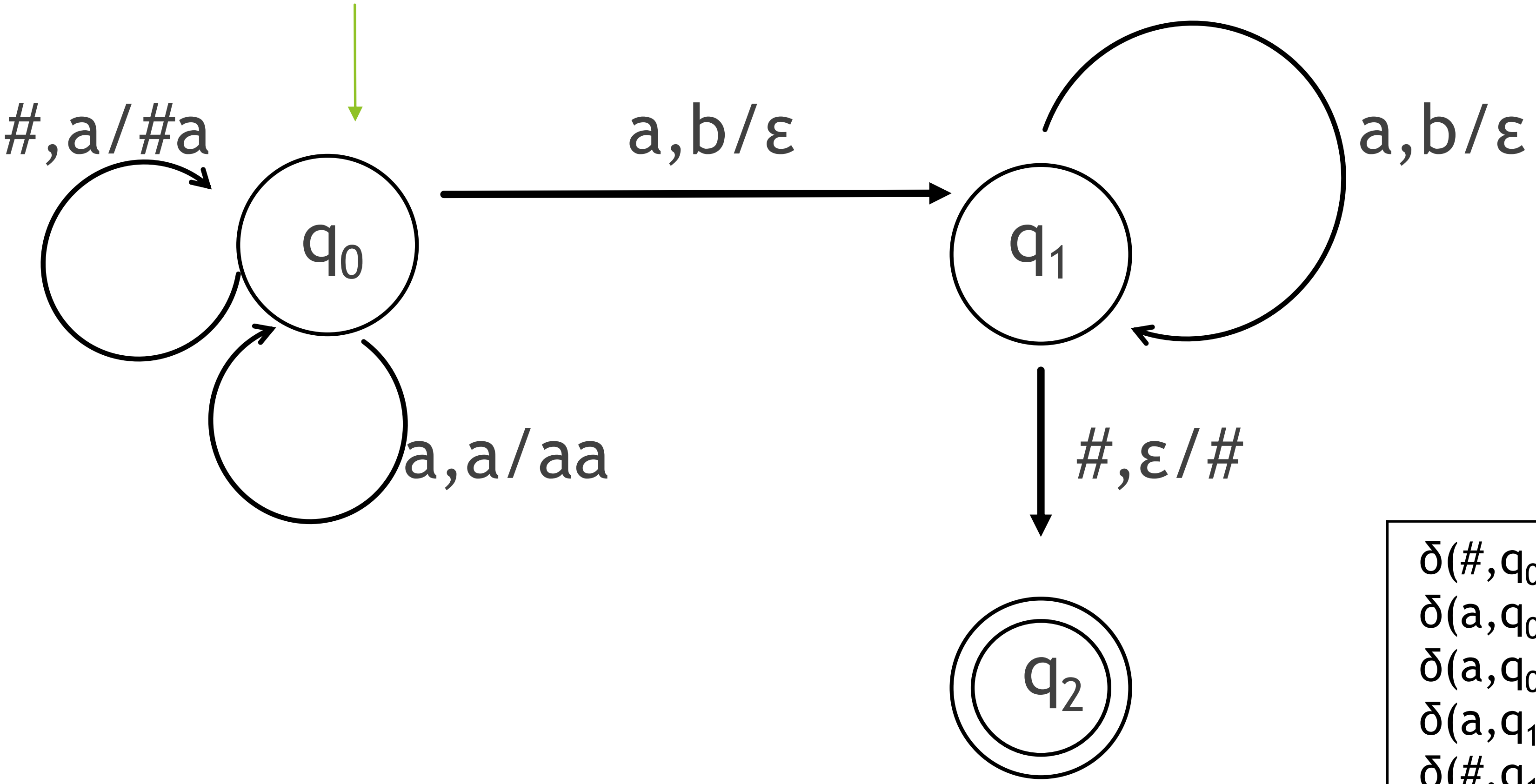
$\delta(a,q_0,b) = \{(\varepsilon,q_1)\}$

$\delta(a,q_1,b) = \{(\varepsilon,q_1)\}$

$\delta(\#,q_1,\varepsilon) = \{(\#,q_2)\}$

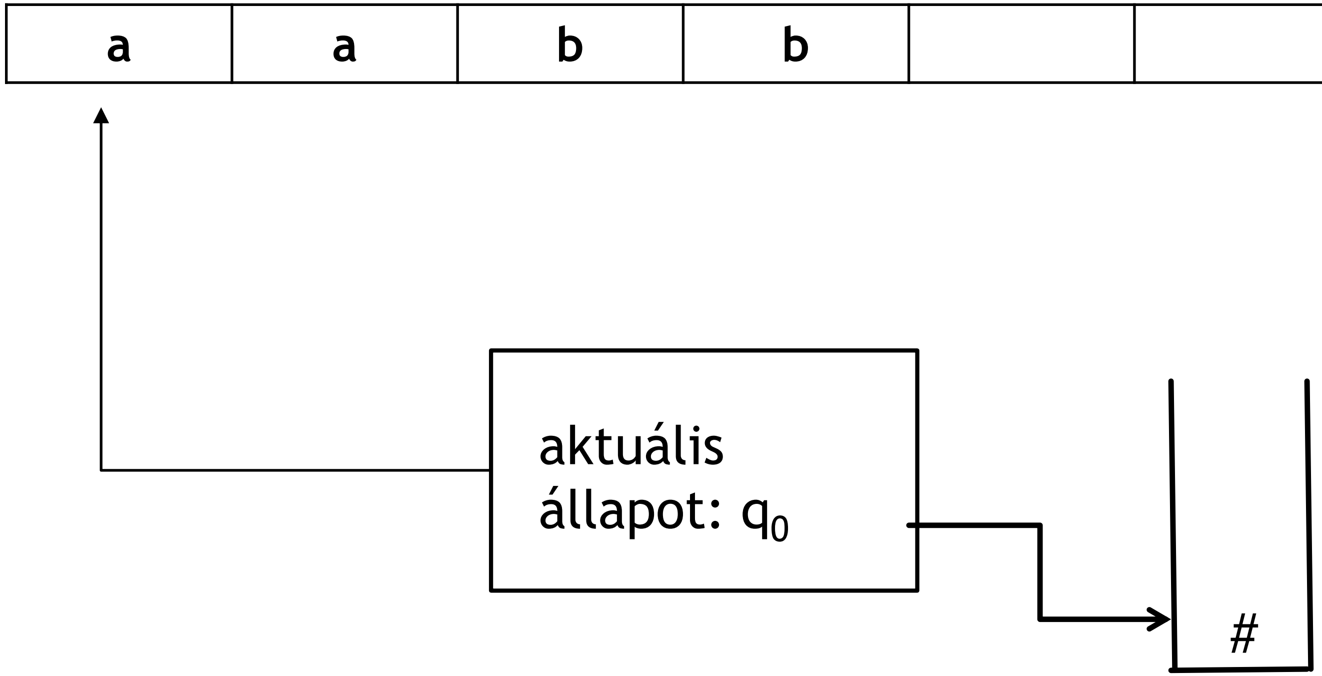
Megjegyzés: Itt most csak egy eleműek a halmazok. Gyakran ilyenkor nem írjuk ki a halmaz jelet.

Példa – veremautomata gráffal



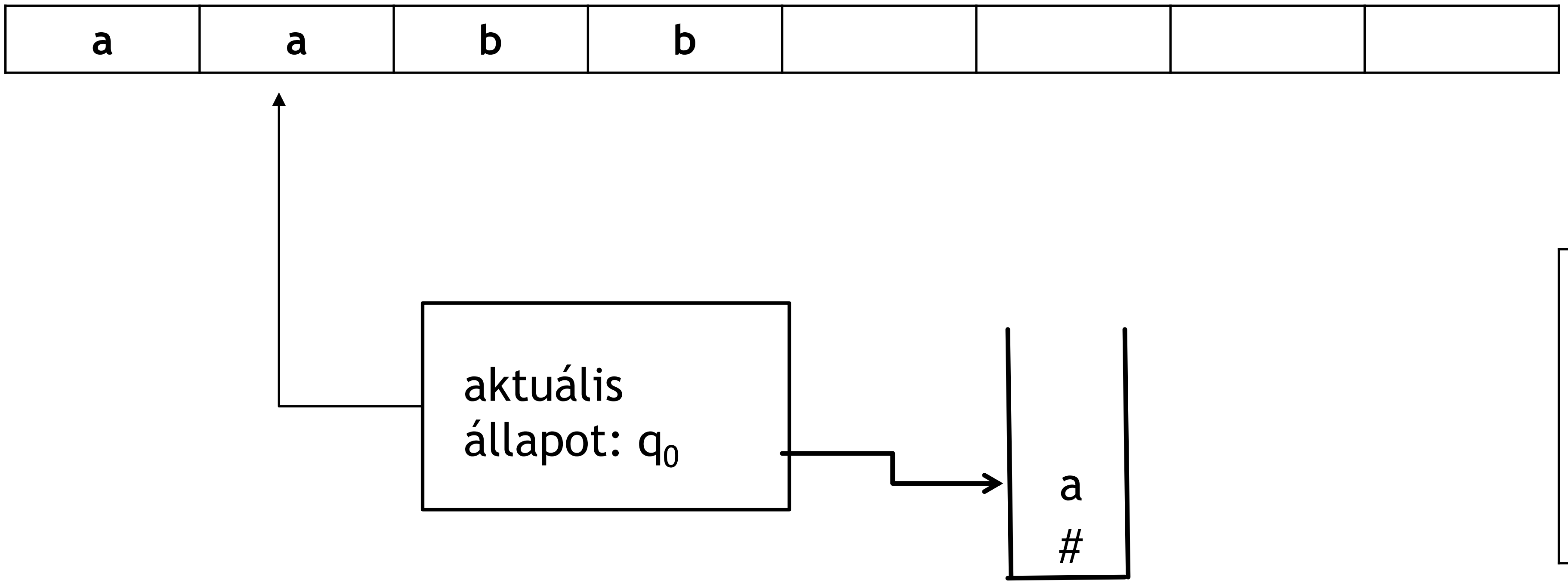
$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\varepsilon) = (\#,q_2)$

Veremautomata működése



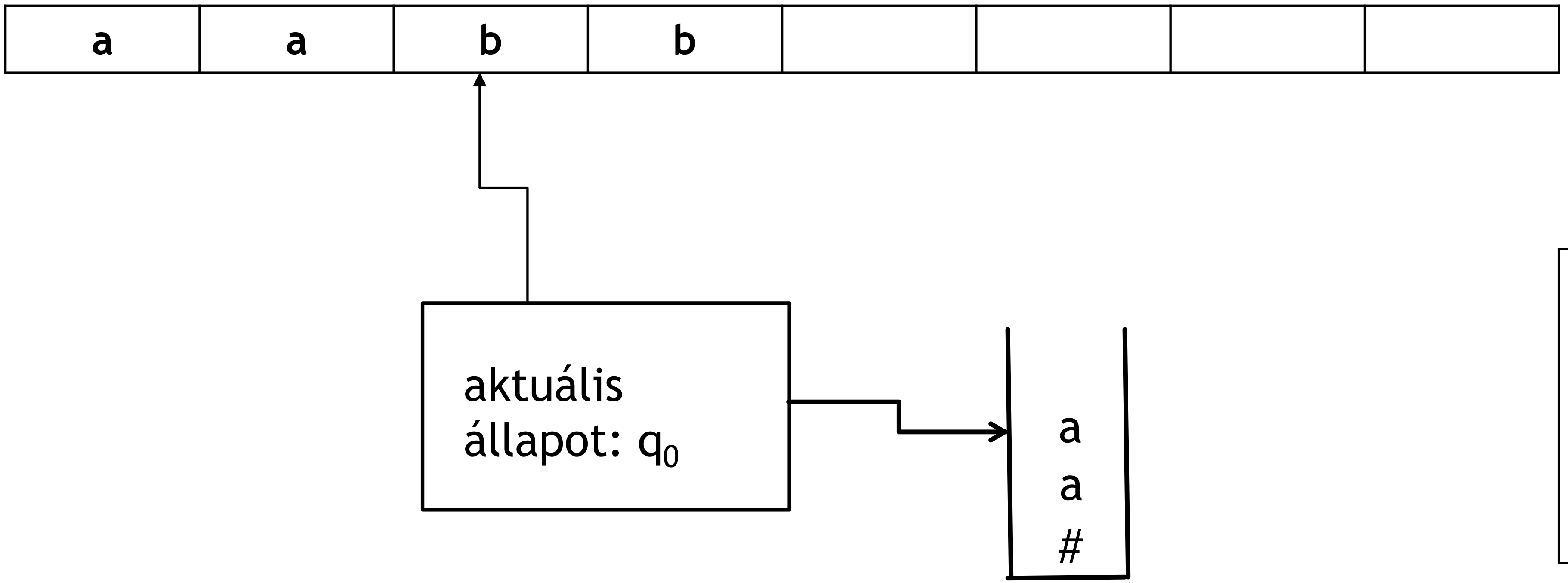
$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\varepsilon) = (\#,q_2)$

Veremautomata működése



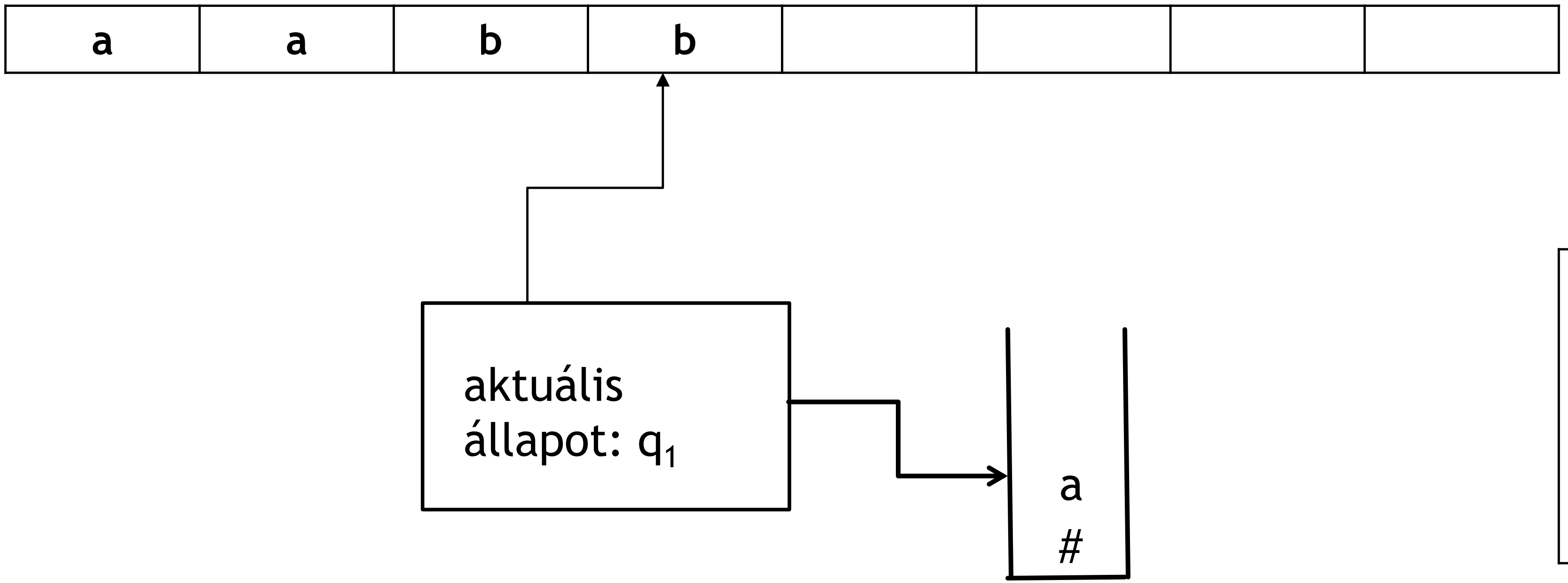
$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\varepsilon) = (\#,q_2)$

Veremautomata működése



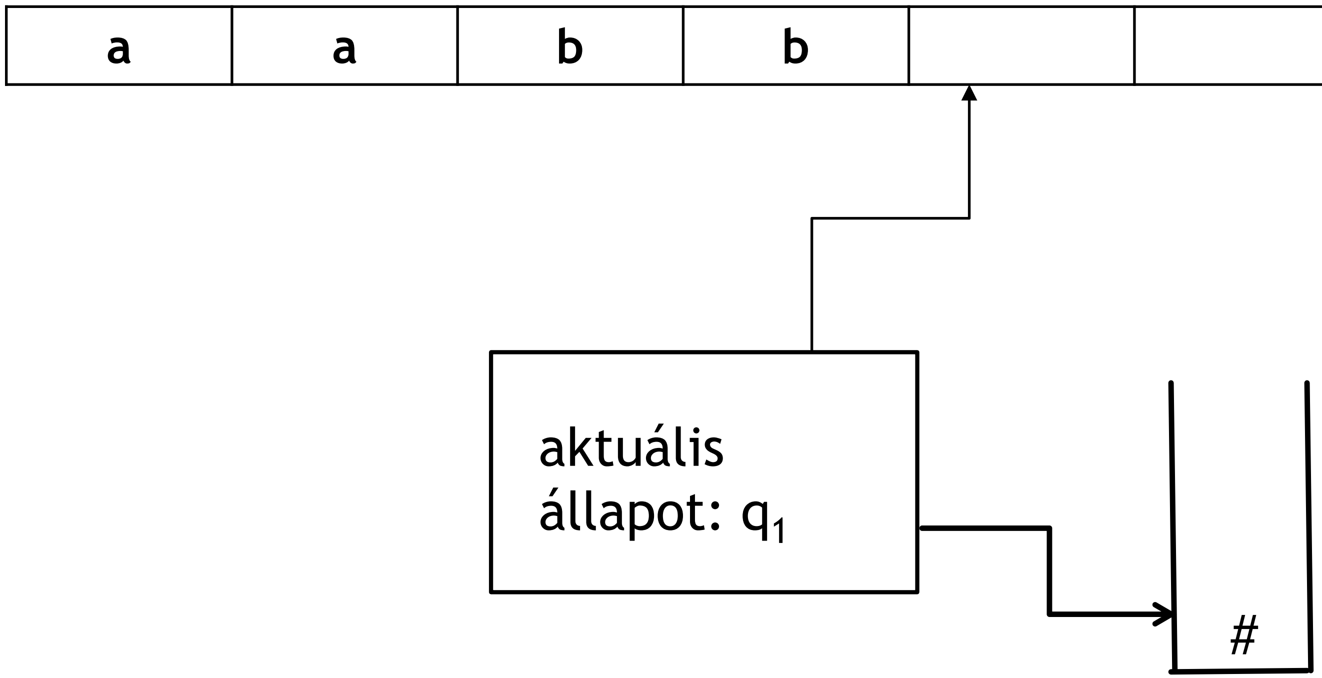
$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\varepsilon) = (\#,q_2)$

Veremautomata működése



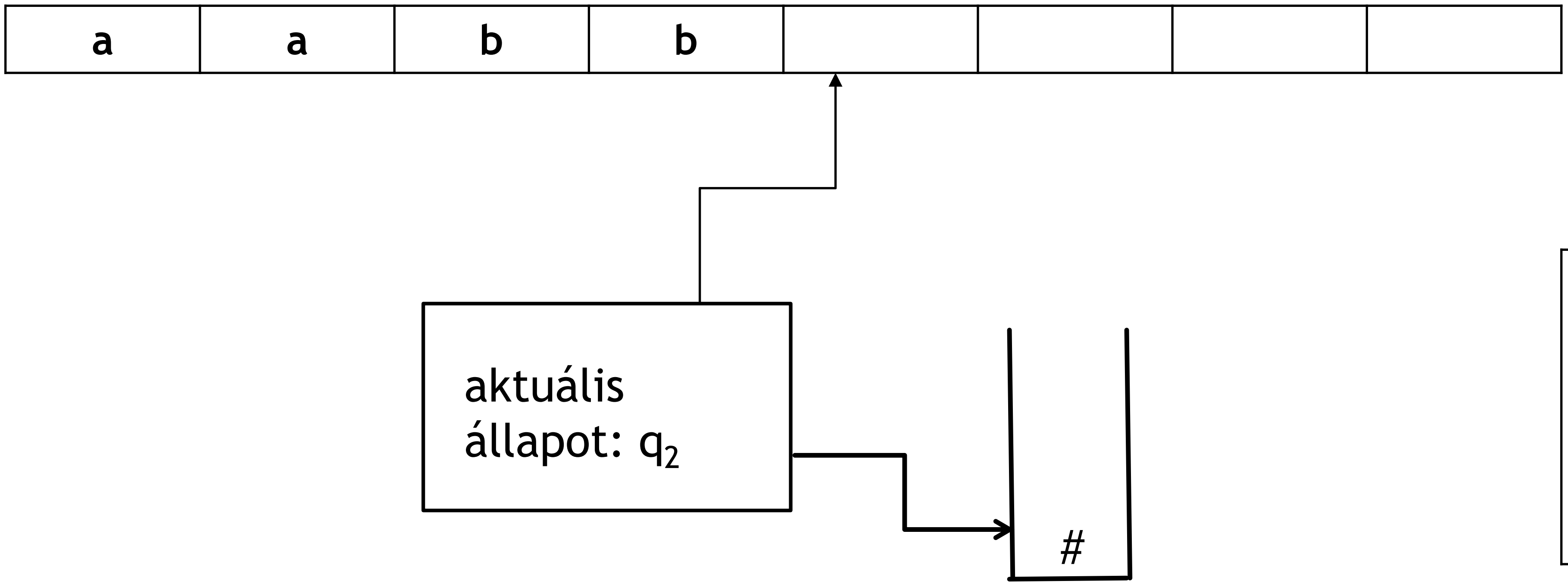
$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\varepsilon) = (\#,q_2)$

Veremautomata működése



$$\delta(\#,q_0,a) = (\#a,q_0)$$
$$\delta(a,q_0,a) = (aa,q_0)$$
$$\delta(a,q_0,b) = (\varepsilon,q_1)$$
$$\delta(a,q_1,b) = (\varepsilon,q_1)$$
$$\delta(\#,q_1,\varepsilon) = (\#,q_2)$$

Veremautomata működése



$\delta(\#,q_0,a) = (\#a,q_0)$
$\delta(a,q_0,a) = (aa,q_0)$
$\delta(a,q_0,b) = (\varepsilon,q_1)$
$\delta(a,q_1,b) = (\varepsilon,q_1)$
$\delta(\#,q_1,\ \varepsilon) = (\#,q_2)$

Az aktuális állapot elfogadó
és a szót végig olvastuk.
Tehát jó a szó.

Veremautomata - alternatív jelöléssel

Ha $\delta(z,q,a) = \{(w_1,r_1),\ldots, (w_k,r_k)\}$, akkor ezt a leképezést a következő szabályhalmazzal is jelölhetjük:

$\mathbf{zqa} \rightarrow \mathbf{w_i r_i}$,ahol $1 \leq i \leq k$.

Ha $\delta(z,q,\varepsilon) = \{(w_1,r_1),\ldots, (w_k,r_k)\}$, akkor ezt a leképezést a következő szabályhalmazzal is jelölhetjük:

$\mathbf{zq} \rightarrow \mathbf{w_i r_i}$,ahol $1 \leq i \leq k$.

Tehát a szabályok baloldala **ZQT** vagy **ZQ** alakú és

a jobboldala **Z*Q** alakú.

Veremautomata - alternatív jelöléssel

<u>Példa:</u>

Legyen $L=\{a^n b^n \mid n \geq 1\}$.

Ezt a nyelvet felismerő veremautomata:

$A = (\{\#,a\}, \{q_0, q_1, q_2\}, \{a,b\}, \delta, \#, q_0, \{q_2\})$

$$\#q_0a \to \#aq_0$$

$$aq_0a \to aaq_0$$

$$aq_0b \to q_1$$

$$aq_1b \to q_1$$

$$\#q_1 \to \#q_2$$

Konfiguráció

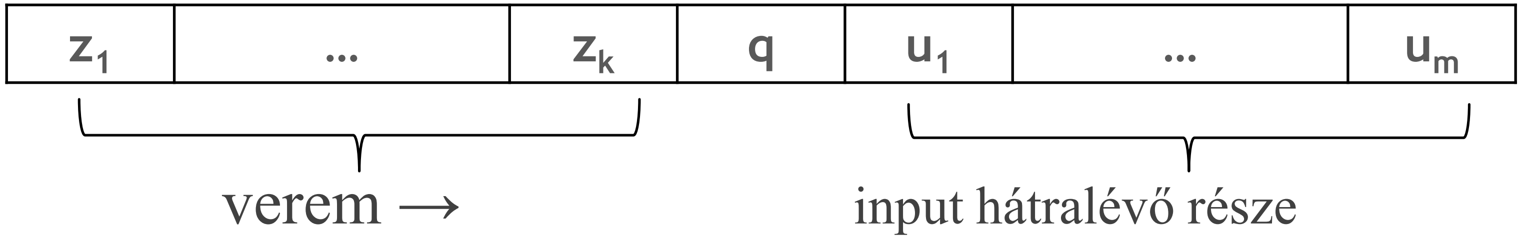
Legyen A = (Z, Q, T, δ, z_0, q_0, F) egy veremautomata és legyen **α** ∈ **Z*QT*** .

Azt mondjuk **α** az A veremautomata egy **konfigurációja.**

(A konfiguráció a veremautomata egy pillanatnyi állapotát írja le.)

Ha **α=zqu**, ahol z ∈ Z* és q ∈ Q és u ∈ T* és

$z=z_1 \ldots z_k$ és $u=u_1 \ldots u_m$, akkor z_1 a verem alján és z_k a tetején lévő karakter és u az input szöveg még el nem olvasott része, ahol u_1 a soron következő karakter.



Kezdő konfiguráció: z_0q_0w ,ahol w∈ T* az elemzendő szó.

Közvetlen redukció - definíció

Legyen $A = (Z, Q, T, \delta, z_0, q_0, F)$ egy veremautomata és legyenek $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbf{Z^*QT^*}$ konfigurációk.
(Konfiguráció: verem, aktuális állapot, input hátralévő része.)

Azt mondjuk, hogy az A veremautomata az $\boldsymbol{\alpha}$ konfigurációt a $\boldsymbol{\beta}$ konfigurációra **redukálja** közvetlenül (jelölés: $\alpha \underset{\mathbf{A}}{\Rightarrow} \beta$),

ha van olyan $z \in Z$, $q,p \in Q$, $a \in T \cup \{\varepsilon\}$ és $r,u \in Z^*$, $w \in T^*$ szó, hogy **zqa** $\rightarrow$ **up** egy szabály és

$\boldsymbol{\alpha}$ = **rzqa**w és $\boldsymbol{\beta}$ = **rup**w teljesül.

Redukció - definíció

Legyen A = (Z, Q, T, δ, z_0, q_0, F) egy veremautomata és legyenek $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ **∈ Z*QT*** .

(Konfiguráció: verem, aktuális állapot, input hátralévő része.)

Azt mondjuk, hogy az A veremautomata az $\boldsymbol{\alpha}$ konfigurációt a $\boldsymbol{\beta}$ konfigurációra redukálja (jelölés: $\alpha \underset{\mathbf{A}}{\Rightarrow}^* \beta$),

ha vagy $\alpha=\beta$ vagy létezik $\alpha_1,\ldots,\alpha_k$ konfiguráció sorozat, hogy $\alpha_1=\alpha$ és $\alpha_k=\beta$ és $\alpha_i \underset{\mathbf{A}}{\Rightarrow} \alpha_{i+1}$ $1 \le i \le k-1$.

Szó levezetése:

Legyen $L=\{a^n b^n \mid n \geq 1\}$.

Ezt a nyelvet felismerő veremautomata:

$A = (\{\#,a\}, \{q_0, q_1, q_2\}, \{a,b\}, \delta, \#, q_0, \{q_2\})$

$\#q_0a \rightarrow \#aq_0$

$aq_0a \rightarrow aaq_0$

$aq_0b \rightarrow q_1$

$aq_1b \rightarrow q_1$

$\#q_1 \rightarrow \#q_2$

u = aabb

$$\#q_0aabb \underset{A}{\Rightarrow} \#aq_0abb \underset{A}{\Rightarrow} \#aaq_0bb \underset{A}{\Rightarrow} \#aq_1b \underset{A}{\Rightarrow} \#q_1 \underset{A}{\Rightarrow} \#q_2$$

Veremautomata által elfogadott nyelv

Elfogadó állapottal felismerhető nyelv:

$$L(A) := \{ u \in T^* \mid \exists\ z_0 q_0 u \underset{A}{\Rightarrow}^* wr \text{ és } r \in F \text{ és } w \in Z^* \}.$$

Megjegyzés: Ez azt jelenti, hogy van olyan működése a veremautomatának, hogy kezdő konfigurációból indulva végig olvasva az inputot elfogadóállapotba jut.

Üres veremmel felismerhető nyelv:

$$N(A) := \{ u \in T^* \mid \exists\ z_0 q_0 u \underset{A}{\Rightarrow}^* r \text{ és } r \in Q \}.$$

Megjegyzés: Ez azt jelenti, hogy van olyan működése a veremautomatának, hogy kezdő konfigurációból indulva végig olvasva az inputot teljesen kiüríti a vermet.

Determinisztikus veremautomata

Egy veremautomatát **determinisztikus**nak mondunk, ha minden $\alpha \in Z^{+}QT^{*}$ konfiguráció esetén egyetlen konfiguráció vezethető le közvetlenül α -ból.

Megjegyzés: Ez azt jelenti, hogy nincs két olyan szabály, amelynek azonos a baloldala, valamint, ha zq egy baloldal, akkor nincs zqa baloldal egyetlen terminálisra sem.

Determinisztikus és nemdeterminisztikus veremautomaták kapcsolata

A determinisztikus veremautomatával felismerhető nyelvek családja <u>szűkebb</u>, mint a nemdeterminisztikussal felismerhető nyelvek családja.

(Például a szimmetrikus szavak nem ismerhetők fel determinisztikus veremautomatával.)

Szimmetrikus szavakat felismerő veremautomata - alternatív jelöléssel

Példa: Legyen $L=\{uu^{-1} \mid u \in \{a,b\}^{+}\}$.

Ezt a nyelvet felismerő veremautomata:

$A = (\{\#,a,b\}, \{q_0,q_1,q_2\}, \{a,b\}, \delta, \#, q_0, \{q_2\})$

$\#q_0a \rightarrow \#aq_0$ $\quad$ $\#q_0b \rightarrow \#bq_0$

$aq_0a \rightarrow aaq_0$ $\quad$ $bq_0b \rightarrow bbq_0$

$aq_0b \rightarrow abq_0$ $\quad$ $bq_0a \rightarrow baq_0$

$aq_0a \rightarrow q_1$ $\quad$ $bq_0b \rightarrow q_1$

$aq_1a \rightarrow q_1$ $\quad$ $bq_1b \rightarrow q_1$

$\#q_1 \rightarrow q_2$

Szó levezetése:

$\#q_0a \to \#aq_0$ $\quad$ $\#q_0b \to \#bq_0$

$aq_0a \to aaq_0$ $\quad$ $bq_0b \to bbq_0$

$aq_0b \to abq_0$ $\quad$ $bq_0a \to baq_0$

$aq_0a \to q_1$ $\quad$ $bq_0b \to q_1$

$aq_1a \to q_1$ $\quad$ $bq_1b \to q_1$

$\#q_1 \to q_2$

u = abba

$\#q_0abba \underset{A}{\Rightarrow} \#aq_0bba \underset{A}{\Rightarrow} \#abq_0ba \underset{A}{\Rightarrow} \#abbq_0a \underset{A}{\Rightarrow} \#abbaq_0$ // nem jó

$\#q_0abba \underset{A}{\Rightarrow} \#aq_0bba \underset{A}{\Rightarrow} \#abq_0ba \underset{A}{\Rightarrow} \#aq_1a \underset{A}{\Rightarrow} \#q_1 \underset{A}{\Rightarrow} q_2$ //jó a szó

A kétféle elfogadás kapcsolata.

<u>Lemma1:</u>

Bármely A veremautomatához megadható A' veremautomata úgy, hogy N(A')=L(A).

<u>Lemma2:</u>

Bármely A veremautomatához megadható A' veremautomata úgy, hogy L(A')=N(A).

Megjegyzés: Ez azt jelenti, hogy ha egy nyelvhez építhető elfogadó állapottal felismerő veremautomata, akkor építhető üresveremmel felismerő veremautomata és fordítva.

A 2-es nyelvcsalád és a veremautomaták kapcsolata

<u>Tétel:</u>

Ha $L \in \mathcal{L}_2$, akkor megadható egy A veremautomata úgy, hogy L=N(A), azaz $\mathcal{L}_2 \subseteq \mathcal{L}_{1V}$.

A 2-es nyelvcsalád és a veremautomaták kapcsolata

Bizonyítás:

Legyen G=(N,T,P,S) egy környezetfüggetlen (2-es típusú) grammatika, amelyre L=L(G).

Ekkor A=(T∪N,{q_0},T,δ,S,q_0,∅), ahol δ a következő:

- $Xq_0 \rightarrow w^{-1}q_0$ akkor és csak akkor, ha $X \rightarrow w \in P$, **X∈N**, w∈(T∪N)*;
- $aq_0a \rightarrow q_0$ akkor és csak akkor, ha **a ∈ T**.

Megjegyzés: A egy egyállapotú üresveremmel elfogadó automata.

A 2-es nyelvcsalád és a veremautomaták kapcsolata

Bizonyítás folytatása:

A verem segítségével egy az elemzendő szó egy legbal levezetését szimuláljuk.

Ha nemterminális van a verem tetején, akkor valamelyik rá vonatkozó szabály jobboldalára cseréljük.

Ha terminális van a verem tetején, akkor az aktuális inputtal egyeztetjük, ha azonosak, akkor kivesszük a terminálist a veremből és tovább lépünk az inputban.

Példa

Legyen $L=\{a^n b^n \mid n \geq 1\}$ és $L=L(G)$, ahol

$G=(\{S\},\{a,b\},P,S)$

P: $S \rightarrow aSb$

$S \rightarrow ab$

Ezt a nyelvet felismerő veremautomata:

$A = (\{S\}, \{q_0\}, \{a,b\}, \delta, S, q)$

$Sq_0 \rightarrow bSaq_0$ //fordítva kell a verembe tenni

$Sq_0 \rightarrow ba$

$aq_0a \rightarrow q_0$

$bq_0b \rightarrow q_0$

Példa

$$Sq_0 \to bSaq_0$$
$$Sq_0 \to ba$$
$$aq_0a \to q_0$$
$$bq_0b \to q_0$$

u =aabb szó elemzése:

$$Sq_0aabb \underset{A}{\Rightarrow} bSaq_0aabb \underset{A}{\Rightarrow} bSq_0abb \underset{A}{\Rightarrow} bbaq_0abb \underset{A}{\Rightarrow}$$

$\underset{A}{\Rightarrow} bbq_0bb \underset{A}{\Rightarrow} bq_0b \underset{A}{\Rightarrow} q_0$,azaz u jó szó.

Példa

$Sq_0 \rightarrow bSaq_0$

$Sq_0 \rightarrow ba$

$aq_0a \rightarrow q_0$

$bq_0b \rightarrow q_0$

u =aabb szó elemzése:

<u>konfiguráció:</u> verem, aktuális állapot, input (Z*QT*)

inicializálás:

Sq_0aabb

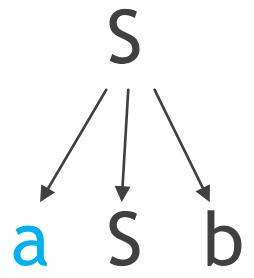
$\Downarrow$

$bSaq_0aabb$

$\Downarrow$

bSq_0abb

szintaxis fa



Példa

$Sq_0 \to bSaq_0$

$Sq_0 \to ba$

$aq_0a \to q_0$

$bq_0b \to q_0$

u =aabb szó elemzése:

Z^*QT^*

Sq_0aabb

$bSaq_0aabb$

bSq_0abb

$\Downarrow$

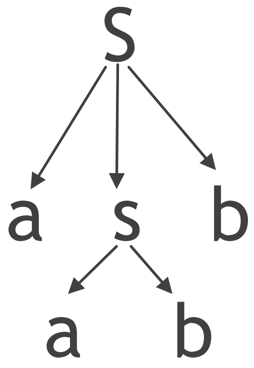
$bbaq_0abb$

bbq_0bb

bq_0b

q_0

szintaxis fa



A 2-es nyelvcsalád és a veremautomaták kapcsolata

<u>Tétel:</u>

Minden A veremautomatához megadható egy környezetfüggetlen G grammatika úgy, hogy L(G)=N(A), azaz $\mathcal{L}_{1V} \subseteq \mathcal{L}_2$.

Megjegyzés: A fordított tételt nem bizonyítjuk.

Köszönöm a figyelmet!