

CF nyelvtanok átalakításai

Az elemzők bizonyos feltételeket teljesítő CF nyelvtanokon hatékonyabban működnek, mint általános CF nyelvtanokon

Ezért elemzés előtt érdemes a nyelvtanokat megfelelően átalakítani

A következő átalakításokra lesz szükségünk:

- felesleges szimbólumok elhagyása
- ϵ -mentesítés
- láncszabály mentesítés

Felesleges szimbólumok elhagyása

Legyen $G = (N, T, R, S)$ egy CF nyelvtan és $A \in N \cup T$

- A -t **befejezhetőnek** nevezünk, ha van olyan $u \in T^*$ szó, melyre $A \Rightarrow^* u$
- A -t **elérhetőnek** nevezzük, ha van olyan A -t tartalmazó u mondatforma, melyre $S \Rightarrow^* u$

A -t **hasznosnak** nevezzük ha befejezhető és elérhető

A nem hasznos szimbólumok feleslegesen bonyolítják a nyelvtant, mert nem vesznek részt semmilyen terminális szó levezetésében, ezért érdemes őket törölni a nyelvtanból

Ha egy G' CF nyelvtan ugyanazt a nyelvet generálja mint G akkor azt mondjuk, hogy a két nyelvtan **ekvivalens**

G -hez megadható egy ekvivalens G' CF nyelvtan úgy, hogy G' -ben minden szimbólum hasznos

Felesleges szimbólumok elhagyása

Bizonyítás

1. A nem befejezhető szimbólumok elhagyása

- Számoljuk ki a befejezhető szimbólumok B halmazát a következő algoritmussal:

- BEFEJEZHETO-SZIMBOLUMOK(G)
 $B = T$;
while B változik **do**
 for minden $A \rightarrow X_1 \dots X_k$ R -beli szabályra **do**
 if $\{X_1, \dots, X_k\} \subseteq B$ **then** $B = B \cup \{A\}$;
return B

$S \rightarrow \cancel{AD} \mid C$
 $A \rightarrow aA \mid a$
 $\cancel{D \rightarrow bD}$
 $C \rightarrow c$

$B = \{a, b, c\}$
 $B = \{a, b, c, C, A\}$
 $B = \{a, b, c, C, A, S\}$

- Ekkor B pontosan azon G -beli szimbólumokat tartalmazza, melyek befejezhetőek
- Hagyjuk el G -ből az összes olyan szabályt ami tartalmaz B -n kívüli nemterminálist
- Belátható, hogy a kapott $\bar{G} = (N \cap B, T, \bar{R}, S)$ nyelvtan ekvivalens G -vel és minden nemterminálisa befejezhető
 - Itt azért annyi feltevéssel éltünk, hogy a nyelvtanunk levezet legalább egy terminális szót, azaz $S \in B$

Felesleges szimbólumok elhagyása

Bizonyítás (folyt.)

2. A nem elérhető szimbólumok elhagyása

- Számoljuk ki az elérhető szimbólumok E halmazát a következő algoritmussal:

- ELERHETO-SZIMBOLUMOK(\bar{G})
 $E = \{S\};$
while E változik **do**
 for minden $A \rightarrow X_1 \dots X_k$ \bar{R} -beli szabályra **do**
 if $A \in E$ **then** $E = E \cup \{X_1, \dots, X_k\};$
return E

$S \rightarrow C$
 ~~$A \rightarrow aA \mid a$~~
 $C \rightarrow c$

$E = \{S\}$
 $E = \{S, C\}$
 $E = \{S, C, c\}$

- Ekkor E pontosan azokat a \bar{G} -beli szimbólumokat tartalmazza, melyek elérhetők
- Hagyjuk el \bar{G} -ből az összes olyan szabályt ami tartalmaz E -n kívüli szimbólumokat
- Belátható, hogy a kapott G' nyelvtan ekvivalens G -vel és minden szimbóluma használható (a nem elérhető szimbólumok elhagyása nem vezet be nem befejezhető nemterminálisokat)
- **Megjegyzés:** a sorrend fontos, mert a nem befejezhető szimbólumok elhagyása bevezethet nem elérhető szimbólumokat

ε -szabályok elhagyása

Legyen $G = (N, T, R, S)$ egy CF nyelvtan és $A \in N$

- Az $A \rightarrow \varepsilon$ alakú szabályokat **ε -szabályoknak** nevezzük
- Ha $A \Rightarrow^* \varepsilon$, akkor A -t **törölhetőnek** nevezzük
- G -t **ε -mentesnek** nevezzük ha nincs benne ε -szabály (kivéve esetleg az $S \rightarrow \varepsilon$ szabályt, de ekkor S nem fordulhat elő szabály jobb oldalán, **röviden: KES**)

Tetszőleges $G = (N, T, R, S)$ CF nyelvtanhoz megadható egy ekvivalens ε -mentes G' CF nyelvtan

Bizonyítás

Először meghatározzuk a törölhető nemterminálisok D halmazát az alábbi algoritmussal:

TOROLHETO-NEMTERMINALISOK(G)

$D = \emptyset$;

for minden $A \rightarrow \varepsilon$ R -beli szabályra

$D = D \cup \{A\}$

while D változik **do**

for minden $A \rightarrow X_1 \dots X_k$ R -beli szabályra **do**

if $\{X_1, \dots, X_k\} \subseteq D$ **then** $D = D \cup \{A\}$;

return D

$S \rightarrow AB$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow bB \mid A$

$D = \emptyset$

$D = \{A\}$

$D = \{A, B\}$

$D = \{A, B, S\}$

ε -szabályok elhagyása

Bizonyítás (folyt.)

Ötlet:

- Tfh. A törölhető és $B \rightarrow \alpha A \beta$ a G egy szabálya
- Adjuk a G -hez a $B \rightarrow \alpha \beta$ szabályt
- Ahelyett, hogy G alkalmazná a $B \rightarrow \alpha A \beta$ -t és utána törölné A -t, alkalmazhatja közvetlenül az új $B \rightarrow \alpha \beta$ szabályt

A konstrukció:

- Hagyjuk el G -ből az összes $A \rightarrow \varepsilon$ alakú szabályt
- Minden további $A \rightarrow w$ szabályt helyettesítsük az összes olyan $A \rightarrow w'$ szabály halmazával, ahol $w' \neq \varepsilon$ és w' a következőképpen áll elő w -ből:
 - **w -ből minden lehetséges módon elhagyjuk a törölhető (azaz **D** -beli) szimbólumokat**
- A kapott \bar{G} ε -mentes és ugyanazt a nyelvet generálja mint G (kivéve az esetleges ε -t)
- Ha a kiindulási G nyelvtan nem generálja az üres szót, akkor G' legyen a \bar{G}
- Egyébként \bar{G} -be felveszünk egy új S' kezdőszimbólumot és az $S' \rightarrow \varepsilon$ meg az $S' \rightarrow S$ szabályt, és legyen G' a kapott nyelvtan

$$S \rightarrow AB$$
$$A \rightarrow aA \mid \varepsilon$$
$$B \rightarrow bB \mid A$$
$$S \rightarrow AB \mid A \mid B$$
$$A \rightarrow aA \mid a$$
$$B \rightarrow bB \mid b \mid A$$

(ha A -t és B -t is elhagynánk ε -szabályt kapnánk)

(ha A -t elhagynánk ε -szabályt kapnánk)

Láncszabályok elhagyása

Legyen $G = (N, T, R, S)$ egy CF nyelvtan és $A, B \in N$

- Az $A \rightarrow B$ alakú szabályokat **láncszabályoknak** nevezzük
- Ha $A \Rightarrow^* B$ csupán láncszabályok alkalmazásával, akkor azt mondjuk, hogy **B lánclevezethető A -ból**
- **Megjegyzés:** A lánclevezethető A -ból, mert $A \Rightarrow^* A$

G -hez megadható egy ekvivalens láncszabálymentes G' CF nyelvtan

Bizonyítás

Minden A nemterminálusra számoljuk ki az A -ból lánclevezethető nemterminálisok N_A halmazát a következő algoritmussal:

LANCLEVEZETHETO(G, A)

$N_A = \{A\};$

while N_A változik **do**

for minden $B \rightarrow C$ R -beli láncszabályra **do**

if $B \in N_A$ **then** $N_A = N_A \cup \{C\};$

return N_A

$S \rightarrow AB \mid A \mid B$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b \mid A$

$N_S = \{S, A, B\}$

$N_A = \{A\}$

$N_B = \{B, A\}$

Láncszabályok elhagyása

Bizonyítás

Ötlet:

- Tfh. B lánclevezethető A -ból és legyen $B \rightarrow \alpha$ a G egy nem láncszabálya
- Adjuk a G -hez a $A \rightarrow \alpha$ szabályt
- Ahelyett, hogy G levezetné A -ból a B -t és utána alkalmazná a $B \rightarrow \alpha$ szabályt
- alkalmazhatja közvetlenül az új $A \rightarrow \alpha$ szabályt

A konstrukció

- Minden A nemterminálisra, $B \in N_A$ -ra és $B \rightarrow \alpha$ R -beli nem láncszabályra vegyük fel G -be az $A \rightarrow \alpha$ szabályt
- Ezután hagyjuk el az összes G -beli láncszabályt
- Legyen a kapott nyelvtan G'

G' láncszabálymentes és ekvivalens G -vel

$S \rightarrow AB \mid A \mid B$	$N_S = \{S, A, B\}$	$S \rightarrow AB aA a bB b$
$A \rightarrow aA \mid a$	$N_A = \{A\}$	$A \rightarrow aA a$
$B \rightarrow bB \mid b \mid A$	$N_B = \{B, A\}$	$B \rightarrow bB b aA a$

CF nyelvtanok átalakítása

Minden L CF nyelv generálható egy G' CF nyelvtannal úgy, hogy G'

- minden szimbóluma használható
- ε -szabály mentes és
- láncszabálymentes

Azaz G minden szabályának jobb oldala vagy egy terminális vagy egy legalább 2 hosszú szó + KES

Bizonyítás

Hajtsuk végre G -n a következő átalakításokat:

- ε -mentesítés
- láncszabálymentesítés
- nem hasznos szimbólumok elhagyása

Ez legyen az elő lépés, mert
behozhat láncszabályokat

Ez legyen a következő, mert behozhat
nem hasznos szimbólumokat

Chomsky normálforma

Legyen $G = (N, T, R, S)$ egy CF nyelvtan

G Chomsky normálformában van ha minden szabálya

- $A \rightarrow a$ (a egy terminális) vagy
- $A \rightarrow BC$ alakú (B és C nemterminálisok)
- + KES

G -hez megadhadható egy ekvivalens Chomsky normálformában lévő G' CF nyelvtan

Bizonyítás

Korábban láttuk, hogy megadható egy olyan \bar{G} CF nyelvtan ami ekvivalens G -vel és a szabályainak a jobb oldala

- egy terminális vagy
- legalább 2 hosszú
- +KES

\bar{G} szabályinak jobboldalait átalakítjuk úgy, hogy a kapott G' ekvivalens \bar{G} -vel és Chomsky normálformában van

Chomsky normálforma

Bizonyítás (folyt.)

Minden $a \in \Sigma$ -ra vegyünk fel egy X_a nemterminálist és az $X_a \rightarrow a$ szabályt

Minden \bar{G} -ben lévő nem $A \rightarrow a$ alakú szabály jobboldalán az a terminális minden egyes előfordulását helyettesítsük X_a -val

Majd minden egyes így átalakított szabályra, mely jobboldalának hossza > 2 , végezzük el az alábbiakat:

- Legyen a szabály $A \rightarrow A_1 \dots A_n$ ($n > 2$)
- Vezessük be az $A \rightarrow A_1 B_1$
 $B_1 \rightarrow A_2 B_2$
 $\dots B_{n-2} \rightarrow A_{n-1} A_n$ szabályokat, ahol B_1, \dots, B_{n-2} új nemterminálisok
- Az $A \rightarrow A_1 \dots A_n$ szabályt cseréljük ki az új szabályokkal
- Legyen G' az így kapott nyelvtan
- Belátható, hogy G' ekvivalens \bar{G} -vel és Chomsky normálformában van

Chomsky normálforma

$$S \rightarrow ASA \mid aB$$

$$A \rightarrow B \mid S$$

$$B \rightarrow b \mid \varepsilon$$

$$S \rightarrow ASA \mid AS \mid SA \mid S \mid aB \mid a$$

$$A \rightarrow B \mid S$$

$$B \rightarrow b$$

$$S \rightarrow ASA \mid AS \mid SA \mid aB \mid a$$

$$A \rightarrow b \mid ASA \mid AS \mid SA \mid aB \mid a$$

$$B \rightarrow b$$

$$D = \{A, B\}$$

$$N_S = \{S\}$$

$$N_A = \{A, B, S\}$$

$$N_B = \{B\}$$

Minden szimbólum hasznos

Chomsky normálforma

$$S \rightarrow ASA \mid AS \mid SA \mid X_a B \mid a$$

$$A \rightarrow b \mid ASA \mid AS \mid SA \mid X_a B \mid a$$

$$X_a \rightarrow a$$

$$B \rightarrow b$$

$$S \rightarrow AB_1 \mid AS \mid SA \mid X_a B \mid a$$

$$B_1 \rightarrow SA$$

$$A \rightarrow b \mid AB_1 \mid AS \mid SA \mid X_a B \mid a$$

$$X_a \rightarrow a$$

$$B \rightarrow b$$

A környezetfüggetlen nyelvek szóproblémája

Legyen az L nyelv egy $G = (N, T, R, S)$ Chomsky normálformában lévő nyelvtannal adott

L szóproblémája **polinom időben** eldönthető

Bizonyítás

Legyen $u = a_1 \dots a_n$ ($a_i \in \Sigma, i = 1, \dots, n$)

A következő, ún. **CYK (Cocke–Younger–Kasami)** algoritmus nagyságrendileg n^3 lépésben eldönti, hogy $u \in L$ teljesül-e

Ötlet: kiszámoljuk minden $l = 1, \dots, n$ -re, $i = 1, \dots, n - l + 1$ -re és $A \in N$ nemterminálisra, hogy deriválható-e A -ból az u i -ik pozíción kezdődő l hosszú részsza

Világos, hogy $u \in L$ akkor és csak akkor ha S -ből deriválható az u első pozícióján kezdődő n -hosszú részsza (azaz az u)

A CYK algoritmus

Bizonyítás (folyt.)

Legyenek G nemterminálisai: A_1, \dots, A_r , ahol $A_1 = S$

Legyen $D[n, n, r]$ egy Boolean típusú tömb

$D[l, i, p]$ akkor és csak akkor *igaz* ha A_p -ből levezethető az u a_i -n kezdődő l hosszú részszava

D -ben kezdetben minden elem *hamis*-ra van állítva

for $i = 1, \dots, n$

- for** minden $A_j \rightarrow a_i$ szabályra

- legyen $D[1, i, j] = igaz$

% inicializálás

for $l = 2, \dots, n$

% a vizsgált rész-szó hossza

- for** $i = 1, \dots, n - l + 1$

% a részszó kezdőpozíciója

- for** $j = 1, \dots, l - 1$

% a részszó partícionálása

- for** minden $A_a \rightarrow A_b A_c$ szabályra % $1 \leq a, b, c \leq r$

- if** $D[j, i, b] = igaz$ és $D[l - j, i + j, c] = igaz$ **then** legyen

- $D[l, i, a] = igaz$

if $D[n, 1, 1] = igaz$ **then**

- $u \in L(G)$

else

- $u \notin L(G)$

Legyen $L = \{a^n b^n \mid n \geq 1\}$ és G az alábbi Chomsky normálformában lévő L -et generáló nyelvtan:

$S \rightarrow AS' \mid AB$

$S' \rightarrow SB$

$A \rightarrow a$

$B \rightarrow b$

Legyen $A_1 = S, A_2 = S', A_3 = A, A_4 = B$, és $u = aabb$

Az alábbi táblázat i -ik sorának j -ik oszlopa pontosan akkor tartalmaz egy A_p nemterminálist ha $D[i, j, p] = igaz$

4	S			
3	\emptyset	S'		
2	\emptyset	S	\emptyset	
1	A	A	B	B
	a	a	b	b

Mivel a táblázat 4. sorának 1. oszlopa tartalmazza $A_1 = S$ -t, kapjuk, hogy $D[4, 1, 1] = igaz$, azaz $u \in L$

Ez valóban igaz: $S \Rightarrow AS' \Rightarrow ASB \Rightarrow AABB \Rightarrow^* aabb$

Veremautomata

l t t j ö n a b e m e n e t

olvasó
fej

Vezérlő

γ_6
 γ_5
 γ_4
 γ_3
 γ_2
 γ_1

Az **olvasó fej** jobbra mozog
(esetleg helyben marad)

A **vezérlőnek** véges a memóriája
(állapothalmaza)

A számításhoz felhasználhat egy
(potenciálisan végtelen) **vermet** is

Veremautomata

Egy M **veremautomata** egy $(Q, \Sigma, \Gamma, \delta, q_0, \$, F)$ rendszer, ahol Q, Σ, q_0, F ugyanazok, mint véges automata esetén,

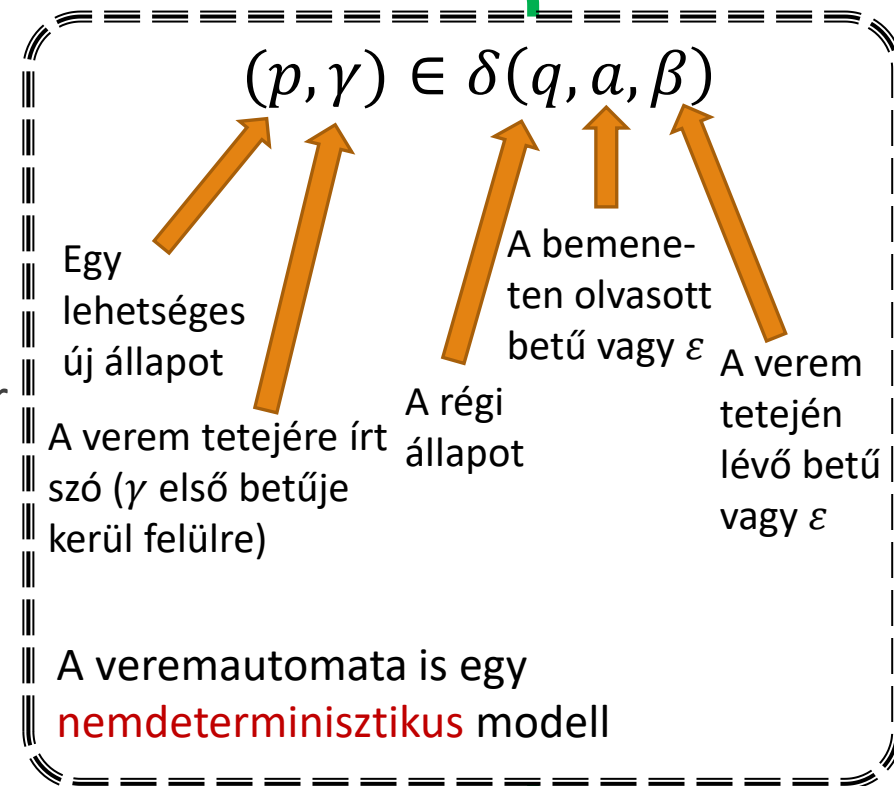
- Γ a **verem ábécé**
- $\$ \in \Gamma$, a **verem alját** jelző betű
- δ az **átmenetfüggvény**, ami $Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon$ -ből képez a $Q \times \Gamma_\varepsilon^*$ véges részhalmazainak halmazába

Ha $(p, \gamma) \in \delta(q, a, \beta)$, ahol $a \in \Sigma_\varepsilon, \beta \in \Gamma_\varepsilon$, és $\gamma \in \Gamma^*$, akkor
 $q \xrightarrow{a, \beta / \gamma} p$ az M egy **átmenete**

A δ most is egyértelműen leírható az **átmeneti diagrammal**

- A csúcsok az állapotok, és két csúcs közte egy $a, \beta / \gamma$ hármassal címkézett éllel megfelel egy átmenetnek

M **megadása** átmeneti diagrammal: megjelöljük a kezdő- és végállapotokat



Veremautomata

Megy q állapotból induló **futása** egy w szón: átmenetek egy

$$q_1 \xrightarrow{a_1, \beta_1 / \gamma_1} q_2 \xrightarrow{a_2, \beta_2 / \gamma_2} q_3 \dots q_n \xrightarrow{a_n, \beta_n / \gamma_n} q_{n+1}$$

sorozata, $n \geq 0$, úgy, hogy

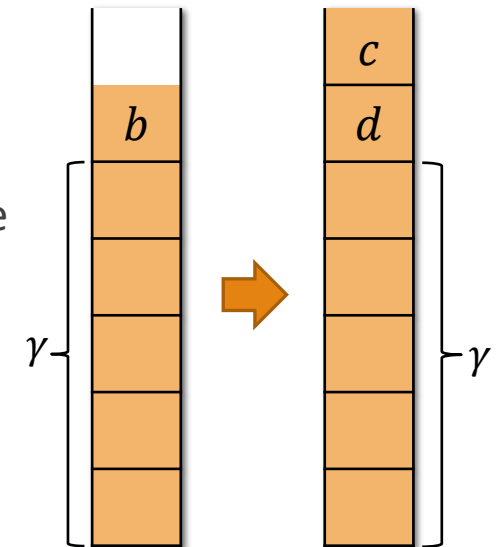
- $q_1 = q$,
- $w = a_1 a_2 \dots a_n$ és
- minden átmenet **kompatibilis** az aktuális veremtartalommal: ha az átmenet β_i -t olvas, akkor a verem tetején β_i van (ami lehet ε is), amit az átmenet kicserél γ_i -re úgy annak a jobbról első betűje kerül alulra

Ez a futás **sikeres**, ha $q_{n+1} \in F$

M **elfogadja** w -t: M -nek van q_0 -ból induló sikeres futása a w -n úgy, hogy kezdetben a veremben csak $\$$ van

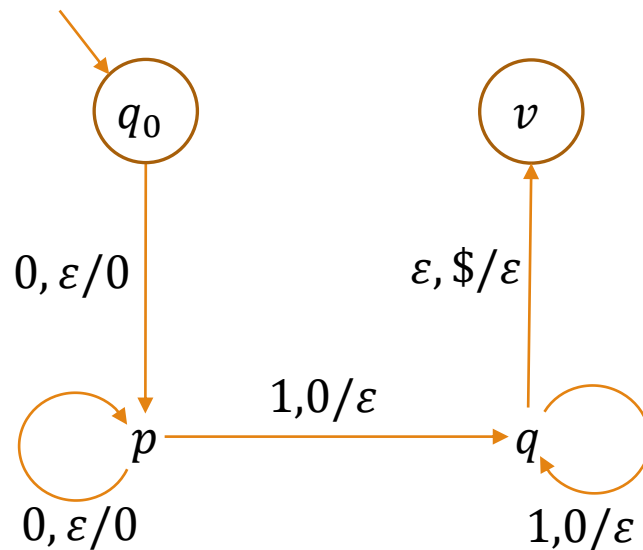
Az M által **felismert nyelv**: $L(M) = \{w \in \Sigma^* \mid M \text{ elfogadja } w\text{-t}\}$

Például egy $q \xrightarrow{a, b/cd} p$ átmenet kompatibilis az alábbi veremtartalommal ($b, c, d \in \Gamma$)

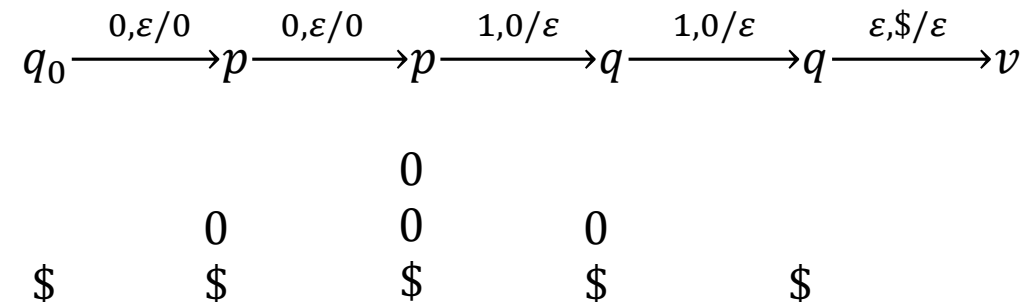


Veremautomata – Példa

Egy veremautomata a $\{0^n 1^n \mid n \geq 0\}$ nyelv felismerésére, \$ a verem alja



A veremautomata egy futása a 0011 szón, alatta a megfelelő veremtartalmak:



- Ez a futás kezdőállapotból indul \$-t tartalmazó veremmel és sikeres is (végállapotba érkezik), tehát a veremautomata elfogadja a 0011 szót
- **Megjegyzés:** a veremautomata nem csak a nyelv szavaira juthat végállapotba, pl. az 00111 szóra is, de ezeket nem fogadja el, mert nem tudja őket végigolvasni!