Limbaje Formale, Automate şi Compilatoare

Curs 6

2018-19

Curs 6

Curs 6

Eliminarea redenumirilor ($A \to B,\ A, B \in N$)

- Intrare: $G = (N, T, S, P)$
- Ieşire: $G' = (N, T, S, P')$, $L(G') = L(G)$, P' nu conţine redenumiri

```
for(A ∈ N){
    N_0 = {A};  i = 0;
    do{
        i = i + 1;
        N_i = N_{i-1} ∪ {C|C ∈ N,  ∃B → C ∈ P, B ∈ N_{i-1}};
    } while  N_i ≠ N_{i-1};
    N_A = N_i; //N_A = {X ∈ N|A ⇒* X}
}
P' = {X → α ∈ P|α ∉ N}
for(X → α_1|α_2|...|α_n ∈ P')
    for(A ∈ N  &&  X ∈ N_A, X ≠ A)
        P' = P' ∪ {A → α_1|α_2|...|α_n}
```

Exemplu

$G = (\{x, y, z\}, \{a, b, c\}, x, P)$, unde P:

- $x \to y|ax|a$
- $y \to z|by|b$
- $z \to cz|c$

$N_x = \{x, y, z\}$, $N_y = \{y, z\}$, $N_z = \{z\}$

Gramatica echivalentă fără redenumiri $G' = (\{x, y, z\}, \{a, b, c\}, x, P')$ unde P':

- $x \to ax|a|by|b|cz|c$
- $y \to by|b|cz|c$
- $z \to cz|c$

Curs 6

Forma normală Chomsky

Definiţie 1

O gramatică este în *formă normală Chomsky* *dacă regulile sale au forma:*

$A \rightarrow BC$, $A \rightarrow a$ *(şi eventual* $S \rightarrow \epsilon$*)* $(A, B, C \in N$ *şi* $a \in T)$.

Teorema 1

Orice limbaj independent de context poate fi generat de o gramatică în formă normală Chomsky.

Demonstraţie

- Se elimină regulile de ştergere şi redenumirile

Demonstraţie

- Se elimină regulile de ştergere şi redenumirile
- Se elemină regulile care nu sunt în formă normală Chomsky: Dacă $A \rightarrow x_1 x_2 \ldots x_n$, $n > 1$ este o astfel de regulă atunci o înlocuim cu $A \rightarrow Y_1 Y_2 \ldots Y_n$ unde:
  - $Y_i = x_i$, dacă $x_i \in N$ (neterminalii rămân la fel)
  - $Y_i = x_a$ dacă $x_i = a \in T$ (x_a este neterminal nou) şi se adaugă regula $x_a \rightarrow a$

Demonstraţie

- Se elimină regulile de ştergere şi redenumirile
- Se elemină regulile care nu sunt în formă normală Chomsky: Dacă $A \to x_1 x_2 \ldots x_n$, $n > 1$ este o astfel de regulă atunci o înlocuim cu $A \to Y_1 Y_2 \ldots Y_n$ unde:
  - $Y_i = x_i$, dacă $x_i \in N$ (neterminalii rămân la fel)
  - $Y_i = x_a$ dacă $x_i = a \in T$ (x_a este neterminal nou) şi se adaugă regula $x_a \to a$
- O regulă de forma $A \to Y_1 Y_2 \ldots Y_n$, dacă $n > 2$, o înlocuim cu:
  - $A \to Y_1 Z_1$
  - $Z_1 \to Y_2 Z_2$
  - $\ldots\ldots$
  - $Z_{n-3} \to Y_{n-2} Z_{n-2}$
  - $Z_{n-2} \to Y_{n-1} Y_n$,

    unde $Z_1, Z_2, \ldots, Z_{n-2}$ sunt neterminali noi.

Exemplu

$G = (\{S, A\}, \{a, b, c\}, S, P)$, unde P:

- $S \to aSb|cAc$
- $A \to cA|c$

Gramatica echivalentă în formă normală Chomsky

$G = (\{S, A, x_a, x_b, Z_1, Z_2\}, \{a, b, c\}, S, P')$, unde P':

- $S \to x_a Z_1 | x_c Z_2$
- $Z_1 \to S x_b$
- $Z_2 \to A x_c$
- $A \to x_c A | c$
- $x_a \to a$
- $x_b \to b$
- $x_c \to c$

Curs 6

Algoritmul Cocke Younger Kasami (CYK)

- Problema recunoaşterii în gramatici de tip 2: dată o gramatică de tip 2 si un cuvânt w, să se decidă dacă $w \in L(G)$
- Problema recunoaşterii în gramatici în formă normală Chomsky se poate rezolva cu algoritmul CYK în timp $O(n^3)$.

- Dacă $w = a_1a_2 \dots a_n$ atunci se constuiesc mulţimile

$$V_{ij} = \{A | A \Rightarrow^+ a_i a_{i+1} \dots a_{i+j-1}\}$$

inductiv pentru $j = 1, \dots, n$

$$w \in L(G) \Leftrightarrow S \in V_{1n}$$

Algoritmul Cocke Younger Kasami

- Pentru $j = 1$:
  - V_{i1} = $\{A | A \Rightarrow^{+} a_i\} = \{A | \exists A \to a_i \in P\}$
- Pentru $j > 1$, V_{ij}:
  - Dacă $A \Rightarrow^{+} a_i a_{i+1} \dots a_{i+j-1}$:

    $A \Rightarrow BC \Rightarrow^{+} a_i a_{i+1} \dots a_{i+j-1}$ şi

    $B \Rightarrow^{+} a_i a_{i+1} \dots a_{i+k-1}$ $(B \in V_{ik})$

    $C \Rightarrow^{+} a_{i+k} a_{i+k+1} \dots a_{i+j-1}$ $(C \in V_{i+k\ j-k})$

    unde $1 \leq i \leq n + 1 - j$, $1 \leq k \leq j - 1$
  - $V_{ij} = \bigcup_{k=1}^{j-1} \{A | A \to BC \in P, B \in V_{ik}, C \in V_{i+k\ j-k}\}$

Algoritmul Cocke Younger Kasami

- Notaţie:

$$\{A | A \rightarrow BC \in P, B \in V_{ik}, C \in V_{i+k\ j-k}\} = V_{ik} \circ V_{i+k\ j-k}$$

- Atunci:

  pentru $2 \leq j \leq n, 1 \leq i \leq n + 1 - j$:

$$V_{ij} = \bigcup_{k=1}^{j-1} (V_{ik} \circ V_{i+k\ j-k})$$

Algoritmul Cocke Younger Kasami

- Intrare: $G = (N, T, S, P)$ în formă normală Chomsky, $w = a_1 a_2 \ldots a_n$
- Ieşire: $w \in L(G)$?

```
for(i=1; i<=n; i++)
   V_i1 = {A|∃A → a_i ∈ P};
for(j=2; j<=n; j++)
   for (i=1; i<=n+1-j; i++){
       V_ij = ∅;
       for(k=1; k<=j-1; k++)
          V_ij = V_ij ∪ (V_ik ∘ V_{i+k j-k});
   }
if(S ∈ V_1n) w ∈ L(G) else w ∉ L(G)
```

$V_{i1} = \{A | \exists A \to a_i \in P\}$; $V_{ij} = \emptyset$; $V_{ij} = V_{ij} \cup (V_{ik} \circ V_{i+k\ j-k})$; $S \in V_{1n}$, $w \in L(G)$, $w \notin L(G)$

Exemplu

$G = (\{S, X, Y, Z\}, \{a, b, c\}, S, P)$, unde P:

- $S \rightarrow XY$
- $X \rightarrow XY|a$
- $Y \rightarrow YZ|a|b$
- $Z \rightarrow c$

$w = abc$

Exemplu

$G = (\{S, X, Y, Z\}, \{a, b, c\}, S, P)$, unde P:

- $S \rightarrow XY$
- $X \rightarrow XY|a$
- $Y \rightarrow YZ|a|b$
- $Z \rightarrow c$

$w = abc$

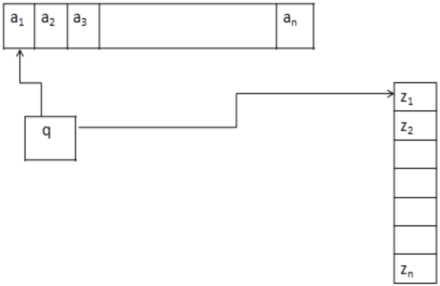
$V_{11} = \{X, Y\}$	$V_{12} = \{S, X\}$	$V_{13} = \{S, X\}$
$V_{21} = \{Y\}$	$V_{22} = \{Y\}$	
$V_{31} = \{Z\}$		

$S \in V_{13} \Leftrightarrow abc \in L(G)$

Curs 6

Automate pushdown

- Automat finit + memorie pushdown (stiva)
- Model fizic:



Automate pushdown-definiţie

Definiţie 2

Un automat pushdown este un 7-uplu: $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$*:*

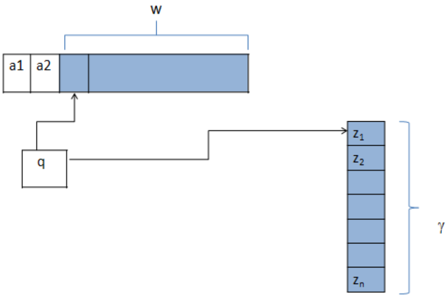
- Q *este mulţimea (finită) a stărilor*
- Σ *este alfabetul de intrare*
- Γ *este alfabetul memoriei pushdown (stivei)*
- $q_0 \in Q$ *este starea iniţială*
- $z_0 \in \Gamma$ *este simbolul iniţial din stivă*
- $F \subseteq Q$ *este mulţimea stărilor finale*
- $\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \to 2^{Q \times \Gamma^*}$

Modelul este nedeterminist

Configuraţia unui automat pushdown

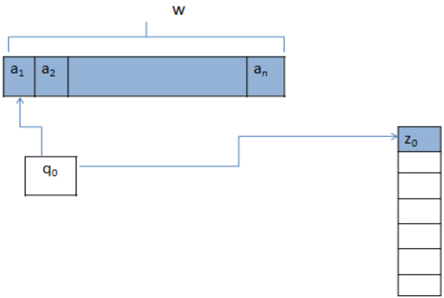
Configuraţie: $(q, w, \gamma) \in Q \times \Sigma^* \times \Gamma^*$

1 : γ (primul simbol din γ) reprezintă vârful stivei



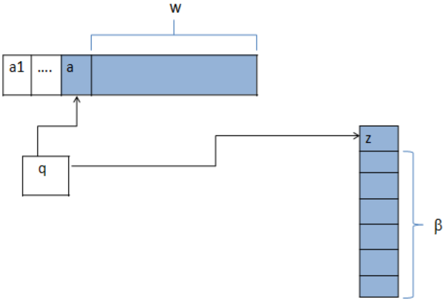
Automate pushdown

Configuraţie iniţială: $(q_0, w, z_0) \in Q \times \Sigma^* \times \Gamma^*$



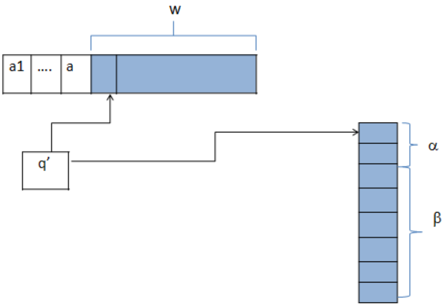
Relaţia de tranziţie între configuraţii

- Configuraţia curentă $(q, aw, z\beta)$ şi $(q', \alpha) \in \delta(q, a, z)$
  $(q, q' \in Q,\ a \in \Sigma \cup \{\epsilon\},\ z \in \Gamma,\ \alpha, \beta \in \Gamma^*)$



Relaţia de tranziţie între configuraţii

- $(q, aw, z\beta) \vdash (q', w, \alpha\beta)$



Relaţia de tranziţie între configuraţii

Fie $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ un automat pushdown.

- Relaţia de tranziţie între configuraţii:
  $(q, aw, z\beta) \vdash (q', w, \alpha\beta)$ dacă $(q', \alpha) \in \delta(q, a, z)$
  $(q, q' \in Q,\ a \in \Sigma \cup \{\epsilon\},\ z \in \Gamma,\ \alpha, \beta \in \Gamma^*)$

- Calcul: închiderea reflexivă şi tranzitivă a relaţiei de mai sus: dacă $C_1, \ldots, C_n$ configuraţii astfel încât:

$$C_1 \vdash C_2 \vdash \ldots \vdash C_n$$

  se scrie: $C_1 \vdash^+ C_n$ dacă $n \geq 2$, $C_1 \vdash^* C_n$, dacă $n \geq 1$

Limbajul recunoscut

Prin stări finale (dacă $F \neq \emptyset$)

$L(M) = \{w \in \Sigma^* | (q_0, w, z_0) \vdash^* (q, \epsilon, \gamma),\ q \in F,\ \gamma \in \Gamma^*\}$

Prin golirea stivei (dacă $F = \emptyset$)

$L_\epsilon(M) = \{w \in \Sigma^* | (q_0, w, z_0) \vdash^* (q, \epsilon, \epsilon),\ q \in Q\}$

Exemplu

Automat care recunoaşte limbajul $\{a^n b^n | n \geq 1\}$:

$$M = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, z\}, \delta, q_0, z, \{q_2\})$$

1. $\delta(q_0, a, z) = \{(q_0, az)\}$
2. $\delta(q_0, a, a) = \{(q_0, aa)\}$
3. $\delta(q_0, b, a) = \{(q_1, \epsilon)\}$
4. $\delta(q_1, b, a) = \{(q_1, \epsilon)\}$
5. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

Exemple

- Un automat pushdown ce recunoaşte limbajul $\{waw^R | w \in \{0,1\}^*\}$

Exemple

- Un automat pushdown ce recunoaşte limbajul $\{waw^R | w \in \{0,1\}^*\}$
  - Fiecare 0 sau 1 citit se introduce în stivă
  - *a* la intrare produce pregătirea scoaterii a câte un simbol din stiva dacă el coincide cu cel din intrare

Exemple

- Un automat pushdown ce recunoaşte limbajul $\{waw^R | w \in \{0,1\}^*\}$
  - Fiecare 0 sau 1 citit se introduce în stivă
  - a la intrare produce pregătirea scoaterii a câte un simbol din stiva dacă el coincide cu cel din intrare

$M = (\{q_0, q_1, q_2\}, \{0, 1, a\}, \{0, 1, z\}, \delta, q_0, z, \{q_2\})$

1. $\delta(q_0, i, z) = \{(q_0, iz)\}, (i \in \{0, 1\})$
2. $\delta(q_0, i, j) = \{(q_0, ij)\}, (i, j \in \{0, 1\})$
3. $\delta(q_0, a, i) = \{(q_1, i)\}$
4. $\delta(q_1, i, i) = \{(q_1, \epsilon)\}$
5. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

Exemple

- Un automat pushdown ce recunoaşte limbajul $\{waw^R | w \in \{0,1\}^*\}$
  - Fiecare 0 sau 1 citit se introduce în stivă
  - *a* la intrare produce pregătirea scoaterii a câte un simbol din stiva dacă el coincide cu cel din intrare

$M = (\{q_0, q_1, q_2\}, \{0, 1, a\}, \{0, 1, z\}, \delta, q_0, z, \{q_2\})$

1. $\delta(q_0, i, z) = \{(q_0, iz)\}$, $(i \in \{0, 1\})$
2. $\delta(q_0, i, j) = \{(q_0, ij)\}$, $(i, j \in \{0, 1\})$
3. $\delta(q_0, a, i) = \{(q_1, i)\}$
4. $\delta(q_1, i, i) = \{(q_1, \epsilon)\}$
5. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

- Un automat pushdown ce recunoaşte limbajul $\{ww^R | w \in \{0,1\}^*\}$?

Exemple

- Un automat pushdown ce recunoaşte limbajul $\{waw^R | w \in \{0,1\}^*\}$
  - Fiecare 0 sau 1 citit se introduce în stivă
  - a la intrare produce pregătirea scoaterii a câte un simbol din stiva dacă el coincide cu cel din intrare

$M = (\{q_0, q_1, q_2\}, \{0, 1, a\}, \{0, 1, z\}, \delta, q_0, z, \{q_2\})$

1. $\delta(q_0, i, z) = \{(q_0, iz)\}, (i \in \{0, 1\})$
2. $\delta(q_0, i, j) = \{(q_0, ij)\}, (i, j \in \{0, 1\})$
3. $\delta(q_0, a, i) = \{(q_1, i)\}$
4. $\delta(q_1, i, i) = \{(q_1, \epsilon)\}$
5. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

- Un automat pushdown ce recunoaşte limbajul $\{ww^R | w \in \{0,1\}^*\}$?
- Un automat pushdown ce recunoaşte limbajul $\{ww | w \in \{0,1\}^*\}$?

Echivalenţa definiţiilor privind recunoaşterea

Teorema 2

Pentru orice automat pushdown M cu $F = \emptyset$, există un automat pushdown M' cu stări finale astfel ca $L(M') = L_\epsilon(M)$.

Echivalenţa definiţiilor privind recunoaşterea

Teorema 2

Pentru orice automat pushdown M cu $F = \emptyset$, există un automat pushdown M' cu stări finale astfel ca $L(M') = L_\epsilon(M)$.

Dacă $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, \emptyset)$, considerăm $M' = (Q \cup \{q_f, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \{q_f\})$ cu δ':

Echivalenţa definiţiilor privind recunoaşterea

Teorema 2

Pentru orice automat pushdown M cu $F = \emptyset$, există un automat pushdown M' cu stări finale astfel ca $L(M') = L_\epsilon(M)$.

Dacă $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, \emptyset)$, considerăm $M' = (Q \cup \{q_f, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \{q_f\})$ cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)

Echivalenţa definiţiilor privind recunoaşterea

Teorema 2

Pentru orice automat pushdown M cu $F = \emptyset$, există un automat pushdown M' cu stări finale astfel ca $L(M') = L_\epsilon(M)$.

Dacă $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, \emptyset)$, considerăm $M' = (Q \cup \{q_f, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \{q_f\})$ cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)
2. $\delta'(q, a, z) = \delta(q, a, z)$, $\forall q \in Q,\ a \in \Sigma \cup \{\epsilon\},\ z \in \Gamma$ (M' face aceleaşi tranziţii ca şi M)

Echivalenţa definiţiilor privind recunoaşterea

Teorema 2

Pentru orice automat pushdown M cu $F = \emptyset$, există un automat pushdown M' cu stări finale astfel ca $L(M') = L_\epsilon(M)$.

Dacă $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, \emptyset)$, considerăm $M' = (Q \cup \{q_f, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \{q_f\})$ cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)
2. $\delta'(q, a, z) = \delta(q, a, z), \forall q \in Q,\ a \in \Sigma \cup \{\epsilon\},\ z \in \Gamma$ (M' face aceleaşi tranziţii ca şi M)
3. $\delta'(q, \epsilon, z_0') = \{(q_f, \epsilon)\}, \forall q \in Q$ (M' va trece în starea finală doar dacă stiva lui M este vidă)

Echivalenţa definiţiilor privind recunoaşterea

Teorema 3

Pentru orice automat pushdown M cu $F \neq \emptyset$, există un automat pushdown M' cu $F = \emptyset$ astfel ca $L_\epsilon(M') = L(M)$.

Echivalenţa definiţiilor privind recunoaşterea

Teorema 3

Pentru orice automat pushdown M cu $F \neq \emptyset$, există un automat pushdown M' cu $F = \emptyset$ astfel ca $L_\epsilon(M') = L(M)$.

Dacă $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$, considerăm

$M' = (Q \cup \{q_\epsilon, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \emptyset)$

Demonstrație

$M' = (Q \cup \{q_\epsilon, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \emptyset)$, cu δ':

Demonstraţie

$M' = (Q \cup \{q_\epsilon, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \emptyset)$, cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)

Demonstrație

$M' = (Q \cup \{q_\epsilon, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \emptyset)$, cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configurația inițială a lui M)
2. a) $\delta'(q, a, z) = \delta(q, a, z)$, $\forall q \in Q,\ a \in \Sigma,\ z \in \Gamma$ (M' face aceleași tranziții ca și M, pentru orice simbol întâlnit)

   b) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z)$, dacă $q \in Q \setminus F,\ z \in \Gamma$ (se fac aceleași ϵ-tranziții ca în M, dacă starea nu este finală)

   c) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z) \cup \{(q_\epsilon, \epsilon)\}$, $q \in F,\ z \in \Gamma$ (daca M ajunge într-o stare finală, M' poate trece într-o stare specială)

Demonstraţie

$M' = (Q \cup \{q_\epsilon, q'_0\}, \Sigma, \Gamma \cup \{z'_0\}, \delta', q'_0, z'_0, \emptyset)$, cu δ':

1. $\delta'(q'_0, \epsilon, z'_0) = \{(q_0, z_0 z'_0)\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)
2. a) $\delta'(q, a, z) = \delta(q, a, z)$, $\forall q \in Q,\ a \in \Sigma,\ z \in \Gamma$ (M' face aceleaşi tranziţii ca şi M, pentru orice simbol întâlnit)
   b) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z)$, dacă $q \in Q \setminus F,\ z \in \Gamma$ (se fac aceleaşi ϵ-tranziţii ca în M, dacă starea nu este finală)
   c) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z) \cup \{(q_\epsilon, \epsilon)\}$, $q \in F,\ z \in \Gamma$ (daca M ajunge într-o stare finală, M' poate trece într-o stare specială)
3. $\delta'(q, \epsilon, z'_0) = \{(q_\epsilon, \epsilon)\}$, dacă $q \in F$ (cazul 2(c), în situaţia în care în stivă este z'_0)

Demonstraţie

$M' = (Q \cup \{q_\epsilon, q_0'\}, \Sigma, \Gamma \cup \{z_0'\}, \delta', q_0', z_0', \emptyset)$, cu δ':

1. $\delta'(q_0', \epsilon, z_0') = \{(q_0, z_0 z_0')\}$ (fără să citească niciun simbol, M' trece în configuraţia iniţială a lui M)
2. a) $\delta'(q, a, z) = \delta(q, a, z)$, $\forall q \in Q,\ a \in \Sigma,\ z \in \Gamma$ (M' face aceleaşi tranziţii ca şi M, pentru orice simbol întâlnit)
   b) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z)$, dacă $q \in Q \setminus F,\ z \in \Gamma$ (se fac aceleaşi ϵ-tranziţii ca în M, dacă starea nu este finală)
   c) $\delta'(q, \epsilon, z) = \delta(q, \epsilon, z) \cup \{(q_\epsilon, \epsilon)\}$, $q \in F,\ z \in \Gamma$ (daca M ajunge într-o stare finală, M' poate trece într-o stare specială)
3. $\delta'(q, \epsilon, z_0') = \{(q_\epsilon, \epsilon)\}$, dacă $q \in F$ (cazul 2(c), în situaţia în care în stivă este z_0')
4. $\delta'(q_\epsilon, \epsilon, z) = \{(q_\epsilon, \epsilon)\}$, dacă $z \in \Gamma \cup \{z_0'\}$ (M' rămâne în starea q_ϵ şi se extrage vârful stivei)

Curs 6

Automatul pushdown echivalent cu o gramatică de tip 2

Teorema 4

Pentru orice gramatică G există un automat pushdown M fără stări finale astfel încât $L_\epsilon(M) = L(G)$

Automatul pushdown echivalent cu o gramatică de tip 2

Teorema 4

Pentru orice gramatică G există un automat pushdown M fără stări finale astfel încât $L_\epsilon(M) = L(G)$

- Fie $G = (N, T, S, P)$
- Construim $M = (\{q\}, T, N \cup T, \delta, q, S, \emptyset)$ unde:
  1. $\delta(q, \epsilon, A) = \{(q, \beta) | A \to \beta \in P\}, \forall A \in N$
  2. $\delta(q, a, a) = \{(q, \epsilon)\}, \forall a \in T$
  3. $\delta(q, x, y) = \emptyset$, în restul cazurilor
- $w \in L(G) \Leftrightarrow S \Rightarrow^+ w \Leftrightarrow (q, w, S) \vdash^+ (q, \epsilon, \epsilon) \Leftrightarrow w \in L_\epsilon(M)$
- M simulează derivările extrem stângi din G

Exemplu

- $G = (\{x\}, \{a, b\}, x, \{x \rightarrow axb, x \rightarrow ab\})$
- Automatul pushdown echivalent:

$$M = (\{q\}, \{a, b\}, \{a, b, x\}, \delta, q, x, \emptyset)$$

1. $\delta(q, \epsilon, x) = \{(q, axb), (q, ab)\}$
2. $\delta(q, a, a) = \{(q, \epsilon)\}$
3. $\delta(q, b, b) = \{(q, \epsilon)\}$