Limbaje Formale, Automate şi Compilatoare

Curs 6

2013-14

Curs 6

1. Eliminarea recursivităţii stângi în gramatici de tip 2
2. Automate pushdown
3. Legătura dintre automatele pushdown şi limbajele de tip 2
4. Automate pushdown deterministe

Curs 6

Recursivitate stângă

- Un neterminal A este stâng recursiv dacă există măcar o derivare $A \Rightarrow^{+} A\beta$. Dacă gramatica G conţine cel puţin un neterminal stâng recursiv, G este stâng recursivă.
- Un neterminal A este stâng recursiv imediat dacă există o regulă $A \rightarrow A\alpha \in P$.

Eliminarea recursivitatii stângi imediate:

- Fie $A \rightarrow A\alpha_1 | A\alpha_2 \ldots | A\alpha_k | \beta_1 | \ldots \beta_n$ toate regulile care încep cu A ($\beta_1, \ldots, \beta_n$ nu încep cu A). Fie P_A mulţimea acestor reguli.
- Gramatica G' în care A nu este stâng recursiv imediat:
  - $G' = (N \cup \{A'\}, T, S, P')$
    $P' = P \setminus P_A \cup \{A' \rightarrow \alpha_1 A' | \ldots \alpha_k A' | \epsilon, A \rightarrow \beta_1 A' | \ldots \beta_n A'\}$

Eliminarea recursivităţii stângi

- Intrare: $G = (N, T, S, P)$ în formă redusă
- Ieşire: $G' = (N', T, S', P')$, $L(G') = L(G)$, fără recursie stângă

Se ordonează N; fie $N' = N = \{A_1, A_2, \ldots, A_n\}$
for(i = 1; i <=n; i++)$\{$
    while($\exists\ A_i \to A_j\alpha \in P : j <= i-1$) $\{$
        $P = P - \{A_i \to A_j\alpha\}$;
        for($A_j \to \beta \in P$) $P = P \cup \{A_i \to \beta\alpha\}$;
    $\}$
    Se elimină recursia stângă imediată pentru A_i
$\}$
N' este obţinută din N prin adăugarea tuturor neterminalilor nou introduşi iar P' este noua mulţime de reguli

Exemplu

$G = (\{A_1, A_2, A_3\}, \{a, b, c\}, A_1, P)$, unde P:

- $A_1 \rightarrow A_2 a | b$
- $A_2 \rightarrow A_3 b$
- $A_3 \rightarrow A_1 c | c$

Gramatica echivalentă care nu este stâng recursivă:

$G' = (\{A_1, A_2, A_3, A_3'\}, \{a, b, c\}, A_1, P')$, unde P':

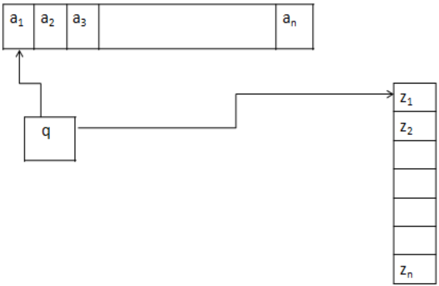
- $A_1 \rightarrow A_2 a | b$
- $A_2 \rightarrow A_3 b$
- $A_3 \rightarrow bcA_3' | cA_3'$
- $A_3' \rightarrow bacA_3' | \epsilon$

Curs 6

1. Eliminarea recursivității stângi în gramatici de tip 2
2. Automate pushdown
3. Legătura dintre automatele pushdown și limbajele de tip 2
4. Automate pushdown deterministe

Automate pushdown

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



Automate pushdown-definiţie

Definiţie 1

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

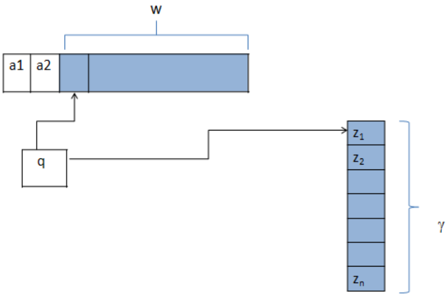
- Q *este mulţimea (finită) a stărilor*
- Σ *este alfabetul de intrare*
- Γ *este alfabetul memoriei pushdown (stivei)*
- $q_0 \in Q$ *este starea iniţială*
- $z_0 \in \Gamma$ *este simbolul iniţial din stivă*
- $F \subseteq Q$ *este mulţimea stărilor finale*
- $\delta : Q \times (\Sigma \cup \{\epsilon\})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

Modelul este nedeterminist

Configuraţia unui automat pushdown

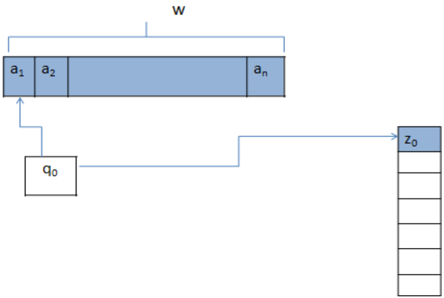
Configuraţie: $(q, w, \gamma) \in Q \times \Sigma^* \times \Gamma^*$

$1 : \gamma$ (primul simbol din γ) reprezintă vârful stivei



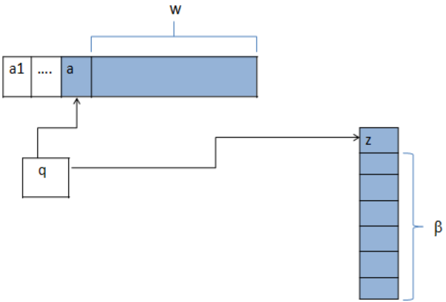
Automate pushdown

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



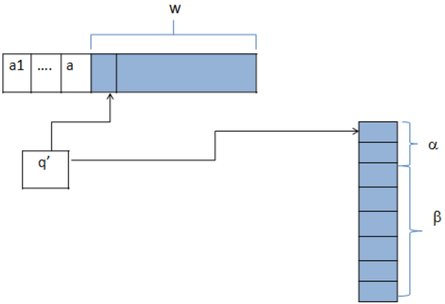
Relaţia de tranziţie între configuraţii

- 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 1

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 1

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 1

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 1

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 1

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 2

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 2

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

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

Demonstraţie

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

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)

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)

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 3

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 3

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

- $G = (\{x\}, \{a, b\}, x, \{x \to axb, x \to ab\})$
- 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. $\delta(q, b, b) = \{(q, \epsilon)\}$

Gramatica echivalentă cu un automat pushdown

Teorema 4

Pentru orice automat pushdown M există o gramatică G astfel încât $L(G) = L_\epsilon(M)$

Gramatica echivalentă cu un automat pushdown

Teorema 4

Pentru orice automat pushdown M există o gramatică G astfel încât $L(G) = L_\epsilon(M)$

- Fie $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, \emptyset)$
- Construim $G = (N, \Sigma, S, P)$ astfel:
  - $N = \{[qzp] | p, q \in Q, z \in \Gamma\} \cup \{S\}$
  - P conţine toate regulile de forma:
    - $S \to [q_0 z_0 q]$, $\forall q \in Q$
    - dacă $(p, \epsilon) \in \delta(q, a, z)$, atunci:
      $[qzp] \to a$
    - dacă $(p, z_1 z_2 \ldots z_m) \in \delta(q, a, z)$, atunci, pentru orice secvenţă de stări $q_1, \ldots, q_m \in Q$:
      $[qzq_m] \to a[pz_1q_1][q_1z_2q_2] \ldots [q_{m-1}z_mq_m]$
- Are loc: $[qzp] \Rightarrow^+ w \Leftrightarrow (q, w, z) \vdash^+ (p, \epsilon, \epsilon)$

Curs 6

1. Eliminarea recursivităţii stângi în gramatici de tip 2
2. Automate pushdown
3. Legătura dintre automatele pushdown şi limbajele de tip 2
4. Automate pushdown deterministe

Definiţie 2

Automatul pushdown $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ *este determinist dacă funcţia de tranziţie* $\delta : Q \times (\Gamma \cup \{\epsilon\}) \times \Gamma \longrightarrow 2^{Q \times \Gamma^*}$ *îndeplineşte condiţiile:*

1. $|\delta(q, a, z)| = 1, \forall a \in \Sigma \cup \{\epsilon\},\ \forall q \in Q,\ \forall z \in \Gamma$
2. *Dacă* $\delta(q, \epsilon, z) \neq \emptyset$ *atunci* $\delta(q, a, z) = \emptyset, \forall a \in \Sigma$

Un automat pushdown determinist poate avea ϵ-tranziţii

Definiţie 2

Automatul pushdown $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ *este determinist dacă funcţia de tranziţie* $\delta : Q \times (\Gamma \cup \{\epsilon\}) \times \Gamma \longrightarrow 2^{Q \times \Gamma^*}$ *îndeplineşte condiţiile:*

1. $|\delta(q, a, z)| = 1, \forall a \in \Sigma \cup \{\epsilon\},\ \forall q \in Q,\ \forall z \in \Gamma$
2. *Dacă* $\delta(q, \epsilon, z) \neq \emptyset$ *atunci* $\delta(q, a, z) = \emptyset, \forall a \in \Sigma$

Un automat pushdown determinist poate avea ϵ-tranziţii

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

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. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

$L(M) = \{waw^R | w \in \{0, 1\}^*\}$

$\mathcal{L}_{2DET}$ - Limbaje de tip 2 deterministe

$\mathcal{L}_{2DET} = \{L | \exists M$ automat pushdown determinist astfel ca $L = L(M)\}$.

- Clasa $\mathcal{L}_{2DET}$ este o clasă proprie a clasei de limbaje $\mathcal{L}_2$ ($\mathcal{L}_{2DET} \subset \mathcal{L}_2$).
- $\{ww^R | w \in \{0, 1\}^*\} \in \mathcal{L}_2 \setminus \mathcal{L}_{2DET}$

$\mathcal{L}_{2DET}$ - Limbaje de tip 2 deterministe

$\mathcal{L}_{2DET} = \{L | \exists M$ automat pushdown determinist astfel ca $L = L(M)\}$.

- Clasa $\mathcal{L}_{2DET}$ este o clasă proprie a clasei de limbaje $\mathcal{L}_2$ ($\mathcal{L}_{2DET} \subset \mathcal{L}_2$).
- $\{ww^R | w \in \{0,1\}^*\} \in \mathcal{L}_2 \setminus \mathcal{L}_{2DET}$
  $M = (\{q_0, q_1, q_2\}, \{0,1\}, \{0,1,z\}, \delta, q_0, z, \{q_2\})$
  1. $\delta(q_0, i, z) = \{(q_0, iz)\}, (i \in \{0,1\})$
  2. $\delta(q_0, i, j) = \{(q_0, ij)\}, (i, j \in \{0,1\},\ i \neq j)$
  3. $\delta(q_0, i, i) = \{(q_0, ii),\ (q_1, \epsilon)\}$
  4. $\delta(q_1, i, i) = \{(q_1, \epsilon)\}$
  5. $\delta(q_1, \epsilon, z) = \{(q_2, \epsilon)\}$

$\mathcal{L}_{2DET}$ - Limbaje de tip 2 deterministe

Definiţie 3

O gramatică G este deterministă dacă:

- *Orice regulă este de forma* $A \to a\alpha$, *unde* $a \in T$ *iar* $\alpha \in (N \cup T)^*$
- *Pentru orice* $A \in N$, *dacă* $A \to a\alpha$, $A \to b\alpha'$ *sunt reguli, atunci* $a \neq b$

Pentru orice gramatică deterministă G există un automat pushdown determinist M astfel ca $L(G) = L(M)$