Limbaje Formale, Automate şi Compilatoare

Curs 1

2022-23

Limbaje Formale, Automate şi Compilatoare - Curs 1

Limbaje Formale, Automate şi Compilatoare

Titulari curs:

- O. Captarencu: otto@info.uaic.ro

  `http://profs.info.uaic.ro/~otto/lfac.html`

- A. Moruz:mmoruz@info.uaic.ro

Sistem evaluare

- 7 seminarii, 6 laboratoare;
- AS = activitatea la seminar (nota de la 0 la 10);
- AL = activitatea la laborator (nota de la 0 la 10);
- T = test scris in sesiune : (nota de la 1 la 10) ;

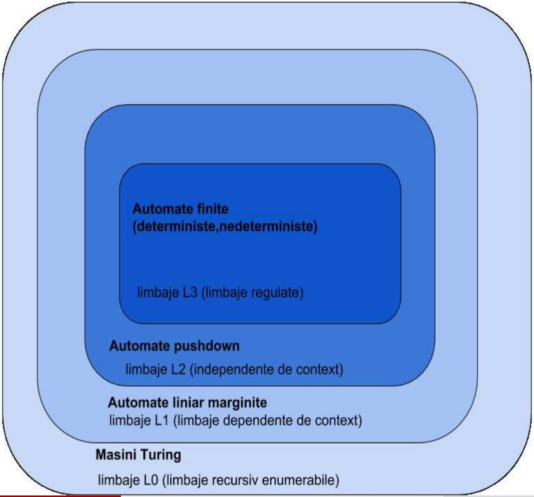
  Punctajul final se obţine astfel:

  P = 2.5 * AS + 2.5 * AL + 5 * T
- Condiţii miminale de promovare: (AS + AL)$\geq$ 10, AS $\geq$ 3, AL $\geq$ 3, $T \geq 5$;
- Nota finală se va stabili conform distribuţiei Gauss

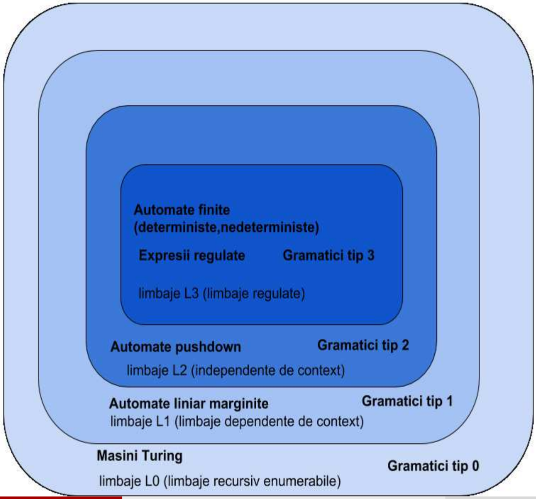
Sistem evaluare

- AS = activitatea la seminar (max 10 puncte):
  - activitatea in timpul seminarului (rezolvare probleme): max 2 puncte (+ bounsuri)
  - test scris: max 8 puncte (nu se reia în sesiunea de re-examinare)
- AL = activitatea la laborator (nota proiect)

Tematica cursului (partea I): Limbaje formale si automate



Tematica cursului (partea I): Limbaje formale si automate



Tematica cursului (partea I):Limbaje formale si automate

- Limbaje şi gramatici
- Limbaje regulate; gramatici, automate , expresii regulate
- Limbaje independente de context; gramatici, automate pushdown

Tematica cursului (partea II)

- Limbaje de programare: proiectare şi implementare
- Analiza lexicală
- Analiza sintactică
- Traducere în cod intermediar

Bibliografie (selecţii)

1. A. V. Aho, M. S. Lam, R. Sethi, J. D. Ullman: Compilers: Principles, Techniques, and Tools. Boston: Addison-Wesley, 2007
2. Gh. Grigoras. Constructia compilatoarelor - Algoritmi fundamentali, Ed. Universitatii Al. I. ”Cuza Iasi”, ISBN 973-703-084-2, 274 pg., 2005
3. Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D. (2006). Introduction to Automata Theory, Languages, and Computation (3rd ed.). Addison-Wesley
4. J. Toader - Limbaje formale şi automate, Editura Matrix Rom, Bucuresti, 1999.
5. J. Toader, S. Andrei - Limbaje formale şi teoria automatelor. Teorie şi practică, Editura Universitatii ”Al. I. Cuza”, Iasi, 2002.

Limbaje Formale, Automate şi Compilatoare - Curs 1

Alfabet, cuvânt, mulţime de cuvinte

- Alfabet: V o mulţime finită (elementele lui V = simboluri)

Alfabet, cuvânt, multţime de cuvinte

- Alfabet: V o mulţime finită (elementele lui V = simboluri)
- Cuvânt: şir finit de simboluri
    - cuvântul nul este notat cu ϵ sau λ.

Alfabet, cuvânt, multţime de cuvinte

- Alfabet: V o mulţime finită (elementele lui V = simboluri)
- Cuvânt: şir finit de simboluri
  - cuvântul nul este notat cu ϵ sau λ.
- Lungimea unui cuvânt u: numarul simbolurilor sale. Notaţie: $|u|$. $|\epsilon| = 0$

Alfabet, cuvânt, multţime de cuvinte

- Alfabet: V o mulţime finită (elementele lui V = simboluri)
- Cuvânt: şir finit de simboluri
  - cuvântul nul este notat cu ϵ sau λ.
- Lungimea unui cuvânt u: numarul simbolurilor sale. Notaţie: $|u|$.
  $|\epsilon| = 0$
- V^* - multimea tuturor cuvintelor peste alfabetul V, inclusiv ϵ.
  $\{0, 1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots\}$

Alfabet, cuvânt, multţime de cuvinte

- Alfabet: V o mulţime finită (elementele lui V = simboluri)
- Cuvânt: şir finit de simboluri
  - cuvântul nul este notat cu ϵ sau λ.
- Lungimea unui cuvânt u: numarul simbolurilor sale. Notaţie: $|u|$.

  $|\epsilon| = 0$
- V^* - multimea tuturor cuvintelor peste alfabetul V, inclusiv ϵ.

  $\{0, 1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots\}$
- V^+ - multimea tuturor cuvintelor nenule peste alfabetul V

  $\{0, 1\}^+ = \{0, 1, 00, 01, 10, 11, 000, 001, \ldots\}$

Operaţii pe cuvinte

- Concatenarea a doua cuvinte x, y: cuvântul $x \cdot y$ obţinut din simbolurile lui x, în ordinea în care apar, urmate de cele ale lui y de asemenea în ordinea în care apar:
  $x = 0100, y = 100, x \cdot y = 0100100$
  $x = 000, y = \epsilon, x \cdot y = 000$

Operaţii pe cuvinte

- Concatenarea a doua cuvinte x, y: cuvântul $x \cdot y$ obţinut din simbolurile lui x, în ordinea în care apar, urmate de cele ale lui y de asemenea în ordinea în care apar:
  $x = 0100, y = 100, x \cdot y = 0100100$
  $x = 000, y = \epsilon, x \cdot y = 000$
- Concatenarea este asociativă

Operaţii pe cuvinte

- Concatenarea a doua cuvinte x, y: cuvântul $x \cdot y$ obţinut din simbolurile lui x, în ordinea în care apar, urmate de cele ale lui y de asemenea în ordinea în care apar:
  $x = 0100$, $y = 100$, $x \cdot y = 0100100$
  $x = 000$, $y = \epsilon$, $x \cdot y = 000$
- Concatenarea este asociativă
- $(V^*, \cdot)$ este monoid (ϵ este element neutru), se numeşte monoidul liber generat de V.

- Fie V un alfabet. O submulţime $L \subseteq V^*$ este un limbaj (formal) peste alfabetul V (sau V-limbaj) dacă L are o descriere (matematică) finită.
- O descriere poate fi:

- Fie V un alfabet. O submulţime $L \subseteq V^*$ este un limbaj (formal) peste alfabetul V (sau V-limbaj) dacă L are o descriere (matematică) finită.
- O descriere poate fi:
  - neformală:
    - multimea cuvintelor peste alfabetul $\{0, 1\}$ care contin un numar par de 0.
    - $\{a^n b^n | n \in N\}$.
    - $\{w \in \{0, 1\}^* | w \textit{ se termina in } 00\}$.

- Fie V un alfabet. O submulţime $L \subseteq V^*$ este un limbaj (formal) peste alfabetul V (sau V-limbaj) dacă L are o descriere (matematică) finită.
- O descriere poate fi:
  - neformală:
    - multimea cuvintelor peste alfabetul $\{0, 1\}$ care contin un numar par de 0.
    - $\{a^n b^n | n \in N\}$.
    - $\{w \in \{0, 1\}^* | w \textit{ se termina in } 00\}$.
  - formală (descriere matematică):
    - o descriere inductivă a cuvintelor
    - o descriere generativă a cuvintelor (gramatică generativă)
    - o descriere a unei metode de recunoaştere a cuvintelor din limbaj (automat finit, automat pushdown, etc.)
    - descriere algebrică (expresii regulate)

Operaţii cu limbaje

- Operatiile cu multimi (reuniune, intersectie etc)
- Produs de limbaje: $L_1 \cdot L_2 = \{u \cdot v | u \in L_1, v \in L_2\}$

  Exemplu:

  $L_1 = \{a^n, n \geq 1\}$, $L_2 = \{b^n, n \geq 1\}$

  $L_1 \cdot L_2 = \{a^n b^m, n \geq 1, m \geq 1\}$
- Iteraţia (produsul Kleene): $L^* = \bigcup_{n \geq 0} L^n$, unde:
  - $L^0 = \{\epsilon\}$
  - $L^{n+1} = L^n \cdot L$

  Exemplu:

  $L = \{a\}$, $L^0 = \{\epsilon\}, L^1 = L, L^2 = \{aa\}, \ldots, L^n = \{a^n\}$

  $L^* = \{a^n, n \geq 0\}$

Limbaje Formale, Automate şi Compilatoare - Curs 1

Gramatici

Definiţie 1

O gramatica este un sistem $G = (N, T, S, P)$, unde:

- *N şi T sunt două alfabete disjuncte:*
  - *N este multimea neterminalilor*
  - *T este multimea terminalilor*
- *$S \in N$ este simbolul de start (neterminalul iniţial)*
- *P este o multime finita de reguli (producţii) de forma $x \to y$, unde $x, y \in (N \cup T)^*$ şi x conţine cel puţin un neterminal.*

Derivare

Definiţie 2

Fie $G = (N, T, S, P)$ *o gramatica şi* $u, v \in (N \cup T)^*$.

Spunem că v este derivat direct (într-un pas) de la u prin aplicarea regulii $x \rightarrow y$*, şi notăm* $u \Rightarrow v$*, dacă* $\exists p, q \in (N \cup T)^*$ *astfel încât* $u = pxq$ *şi* $v = pyq$.

Derivare

Definiţie 2

Fie $G = (N, T, S, P)$ *o gramatica şi* $u, v \in (N \cup T)^*$.
Spunem că *v este derivat direct (într-un pas) de la u* *prin aplicarea regulii* $x \to y$, *şi notăm* $u \Rightarrow v$, *dacă* $\exists p, q \in (N \cup T)^*$ *astfel încât* $u = pxq$ *şi* $v = pyq$.

- Daca $u_1 \Rightarrow u_2 \ldots \Rightarrow u_n, n > 1$, spunem ca u_n este derivat din u_1 în G şi notam $u1 \Rightarrow^+ u_n$.

Derivare

Definiţie 2

Fie $G = (N, T, S, P)$ *o gramatica şi* $u, v \in (N \cup T)^*$.

Spunem că *v este derivat direct (într-un pas) de la u* *prin aplicarea regulii* $x \rightarrow y$, *şi notăm* $u \Rightarrow v$, *dacă* $\exists p, q \in (N \cup T)^*$ *astfel încât* $u = pxq$ *şi* $v = pyq$.

- Daca $u_1 \Rightarrow u_2 \ldots \Rightarrow u_n, n > 1$, spunem ca u_n este derivat din u_1 în G şi notam $u1 \Rightarrow^+ u_n$.
- Scriem $u \Rightarrow^* v$ dacă $u \Rightarrow^+ v$ sau $u = v$.

Limbaj generat

Definiţie 3

Limbajul generat de gramatica G este:

$$L(G) = \{w \in T^* | S \Rightarrow^+ w\}$$

Limbaj generat

Definiţie 3

Limbajul generat de gramatica G este:

$$L(G) = \{w \in T^*|S \Rightarrow^+ w\}$$

Definiţie 4

Două gramatici G_1 şi G_2 sunt echivalente dacă $L(G_1) = L(G_2)$.

Exemplu

- $G = (N, T, S, P)$, $N = \{S, S_1, X\}$, $T = \{a, b, c\}$, P constă din:
    1. $S \to abc$
    2. $S \to aS_1Xc$
    3. $S_1 \to abc$
    4. $cX \to Xc$
    5. $bX \to bb$
- $L(G) = \{abc, a^2b^2c^2\}$
- Gramatică echivalentă cu un singur neterminal ?

Exemplu

- $L = \{a^n b^n | n \geq 1\}$
- Definiţia inductivă:
  - $ab \in L$
  - Daca $X \in L$, atunci $aXb \in L$
  - Nici un alt cuvânt nu face parte din L

Exemplu

- $L = \{a^n b^n | n \geq 1\}$
- Definiţia inductivă:
  - $ab \in L$
  - Daca $X \in L$, atunci $aXb \in L$
  - Nici un alt cuvânt nu face parte din L
- Definiţia generativă:
  - $G = (\{X\}, \{a, b\}, X, P)$, unde $P = \{X \rightarrow aXb, X \rightarrow ab\}$
  - Derivarea cuvântului $a^3 b^3$:

    $X \Rightarrow aXb \Rightarrow a(aXb)b \Rightarrow aa(ab)bb$

Exemplu

- $L = \{a^n b^n c^n | n \geq 1\}$
- $G = (N, T, S, P)$, $N = \{S, X\}$, $T = \{a, b, c\}$, P constă din:
  1. $S \rightarrow abc$
  2. $S \rightarrow aSXc$
  3. $cX \rightarrow Xc$
  4. $bX \rightarrow bb$
- Derivarea cuvântului $a^3b^3c^3$:

  $S \Rightarrow^{(2)} a\underline{S}Xc \Rightarrow^{(2)} aa\underline{S}XcXc \Rightarrow^{(1)} aaab\underline{cX}cXc \Rightarrow^{(3)}$
  $aaa\underline{bX}ccXc \Rightarrow^{(4)} aaabbc\underline{cX}c \Rightarrow^{(3)} aaabb\underline{cX}cc \Rightarrow^{(3)}$
  $aaab\underline{bX}ccc \Rightarrow^{(4)} aaabbbccc = a^3b^3c^3$

Limbaje Formale, Automate şi Compilatoare - Curs 1

Ierarhia lui Chomsky

1. **Gramatici de tip 0 (generale)**

   Nu exista restrictii asupra regulilor

Ierarhia lui Chomsky

1. **Gramatici de tip 0 (generale)**

   Nu exista restrictii asupra regulilor

2. **Gramatici de tip 1 (dependente de context)**

   reguli de forma $pxq \rightarrow pyq$ unde $x \in N$, $y \neq \epsilon$, $p, q \in (N \cup T)^*$,

   $S \rightarrow \epsilon$, caz în care S nu apare în dreapta regulilor

Ierarhia lui Chomsky

1. **Gramatici de tip 0 (generale)**

   Nu exista restrictii asupra regulilor

2. **Gramatici de tip 1 (dependente de context)**

   reguli de forma $pxq \to pyq$ unde $x \in N$, $y \neq \epsilon$, $p, q \in (N \cup T)^*$,

   $S \to \epsilon$, caz în care S nu apare în dreapta regulilor

3. **Gramatici de tip 2 (independente de context)**

   reguli de forma $A \to y$ unde $A \in N$ şi $y \in (N \cup T)^*$

Ierarhia lui Chomsky

1. **Gramatici de tip 0 (generale)**

   Nu exista restrictii asupra regulilor

2. **Gramatici de tip 1 (dependente de context)**

   reguli de forma $pxq \to pyq$ unde $x \in N, y \neq \epsilon, p, q \in (N \cup T)^*$,

   $S \to \epsilon$, caz în care S nu apare în dreapta regulilor

3. **Gramatici de tip 2 (independente de context)**

   reguli de forma $A \to y$ unde $A \in N$ şi $y \in (N \cup T)^*$

4. **Gramatici de tip 3 (regulate)**

   reguli $A \to u$ sau $A \to uB$ unde $A, B \in N$ şi $u \in T^*$.

Exemple

Tip 1: $pxq \to pyq$ unde $x \in N$, $y \neq \epsilon$, $p, q \in (N \cup T)^*$, $S \to \epsilon$

- $G = (N, T, S, P)$, $N = \{S, A, B\}$, $T = \{a, b, c\}$, P:
  (1)$S \to aaAc$
  (2)$aAc \to aAbBc$
  (3)$bB \to bBc$
  (4)$Bc \to Abc$
  (5)$A \to a$
  Gramatica tip 1
- $G = (N, T, S, P)$, $N = \{S, X\}$, $T = \{a, b, c\}$, P:
  (1)$S \to abc$
  (2)$S \to aSXc$
  (3)$cX \to Xc$ (nu este regulă de tip 1!, gramatica va fi de tip 0)
  (4)$bX \to bb$

Exemple

Tip 2: $A \rightarrow y$ unde $A \in N$ şi $y \in (N \cup T)^*$

Tip3: $A \rightarrow u$ sau $A \rightarrow uB$ unde $A, B \in N$ şi $u \in T^*$.

- G:

  $(1) x \rightarrow ax$

  $(1) x \rightarrow xb$

  $(2) x \rightarrow \epsilon$

  (Gramatică tip 2)

- G:

  $(1) x \rightarrow ax$

  $(2) x \rightarrow bx$

  $(3) x \rightarrow \epsilon$

  (Gramatică tip 3)

Clasificarea limbajelor

- Un limbaj L este de tipul j daca exista o gramatica G de tipul j astfel incat $L(G) = L$, unde $j \in \{0, 1, 2, 3\}$.
- Vom nota cu $\mathcal{L}_j$ clasa limbajelor de tipul j, unde $j \in \{0, 1, 2, 3\}$.
- Are loc: $\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0$
- Incluziunile sunt stricte:
  - orice limbaj de tip $j + 1$ este si de tip $j \in \{0, 1, 2\}$
  - exista limbaje de tip j care nu sunt de tip $j + 1$, $j \in \{0, 1, 2\}$

Limbaje Formale, Automate şi Compilatoare - Curs 1

Gramatici de tip 3

- O gramatică $G = (N, T, S, P)$ este de tip 3 dacă regulile sale au forma: $A \to u$ sau $A \to uB$ unde $A, B \in N$ şi $u \in T^*$.
- Exemplu:

  Fie $G = (\{A, B\}, \{a, b\}, A, \{A \to aA, A \to B, B \to bB, B \to \epsilon\})$

Gramatici de tip 3

- O gramatică $G = (N, T, S, P)$ este de tip 3 dacă regulile sale au forma: $A \to u$ sau $A \to uB$ unde $A, B \in N$ şi $u \in T^*$.
- Exemplu:

  Fie $G = (\{A, B\}, \{a, b\}, A, \{A \to aA, A \to B, B \to bB, B \to \epsilon\})$

  $L(G) = \{a^n b^m, n, m \geq 0\}$

Exemple

- $G = (\{D\}, \{0, 1, ..., 9\}, D, P)$

  Unde P este:

  $D \rightarrow 0D|1D|2D|\ldots|9D$

  $D \rightarrow 0|1|\ldots|9$

- $G = (\{A, B\}, \{l, d\}, A, P)$ unde P este:

  $A \rightarrow lB,\ B \rightarrow lB|dB|\epsilon$ (l = litera, d = cifra)

Exemple

- $G = (\{D\}, \{0, 1, ..., 9\}, D, P)$

  Unde P este:

  $D \rightarrow 0D|1D|2D| \dots |9D$

  $D \rightarrow 0|1| \dots |9$

- $G = (\{A, B\}, \{l, d\}, A, P)$ unde P este:

  $A \rightarrow lB,\ B \rightarrow lB|dB|\epsilon$ (l = litera, d = cifra)

  $L(G)$: multimea identificatorilor

Forma normală

- O gramatică de tip 3 este in formă normală daca regulile sale sunt de forma $A \rightarrow a$ sau $A \rightarrow aB$, unde $a \in T$, si, eventual $S \rightarrow \epsilon$ (caz in care S nu apare in dreapta regulilor).

- Pentru orice gramatica de tip 3 exista o gramatica echivalenta in forma normala.

Forma normală

- Obtinerea gramaticii in forma normala echivalenta cu o gramatica de tip 3:
  - Se poate arata ca pot fi eliminate regulile de forma $A \to B$ (redenumiri) si cele de forma $A \to \epsilon$ (reguli de stergere), cu exceptia, eventual a regulii $S \to \epsilon$.
  - Orice regula de forma $A \to a_1 a_2 \dots a_n$ se inlocuieste cu $A \to a_1 B_1, B_1 \to a_2 B_2, \dots, B_{n-2} \to a_{n-1} B_{n-1}, B_{n-1} \to a_n$, $n > 1$, $B_1, \dots, B_{n-1}$ fiind neterminali noi.
  - Orice regula de forma $A \to a_1 a_2 \dots a_n B$ se inlocuieste cu $A \to a_1 B_1$, $B_1 \to a_2 B_2, \dots, B_{n-2} \to a_{n-1} B_{n-1}, B_{n-1} \to a_n B$, $n > 1$, $B_1, \dots, B_{n-1}$ fiind neterminali noi
  - Transformarile care se fac nu modifica limbajul generat de gramatica

Limbaje Formale, Automate şi Compilatoare - Curs 1

Fie L, L_1, L_2 limbaje de tip 3 (regulate).

Atunci, urmatoarele limbaje sunt de asemenea de tip 3:

- $L_1 \cup L_2$
- $L_1 \cdot L_2$
- L^*
- $L_1 \cap L_2$
- $L_1 \setminus L_2$

Închiderea la reununiune

Fie L, L_1, L_2 limbaje de tip 3 (regulate).

Fie $G_1 = (N_1, T_1, S_1, P_1)$ si $G_2 = (N_2, T_2, S_2, P_2)$ gramatici de tip 3 cu $L_1 = L(G_1)$, $L_2 = L(G_2)$.

Presupunem $N_1 \cap N2 = \emptyset$

Închiderea la reuniune: se arata ca $L_1 \cup L_2 \in \mathcal{L}_3$:

Gramatica $G = (N_1 \cup N_2 \cup \{S\}, T_1 \cup T_2, S, P_1 \cup P_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\})$ este de tip 3 si genereaza limbajul $L_1 \cup L_2$

Închiderea la operaţia de produs

Fie L_1, L_2 limbaje de tip 3 (regulate).

Fie $G_1 = (N_1, T_1, S_1, P_1)$ si $G_2 = (N_2, T_2, S_2, P_2)$ gramatici de tip 3 cu $L_1 = L(G_1)$, $L_2 = L(G_2)$.

Presupunem $N_1 \cap N2 = \emptyset$

Gramatica $G = (N_1 \cup N_2, T_1 \cup T_2, S_1, P)$ unde P consta din:

- regulile de forma $A \to uB$ din P_1 ($B \in N_1$)
- reguli $A \to uS_2$ pentru orice regula de forma $A \to u$ ($u \in T_1^*$) din P_1
- toate regulile din P_2

este de tip 3 si genereaza limbajul L_1L_2.

Exemplu

$L = \{uc^n, u \in \{a,b\}^+, n \geq 2\}$

$L = L_1 \cdot L_2$, unde: $L_1 = \{a,b\}^+$, $L_2 = \{c^n, n \geq 2\}$

$G1:$

1. $S_1 \rightarrow aS_1$
2. $S_1 \rightarrow bS_1$
3. $S_1 \rightarrow a$
4. $S_1 \rightarrow b$

$G2:$

1. $S_2 \rightarrow cS_2$
2. $S_2 \rightarrow cc$

$G = (\{S_1, S_2\}, \{a,b,c\}, S_1, P)$,

$P:$

1. $S_1 \rightarrow aS_1$
2. $S_1 \rightarrow bS_1$
3. $S_1 \rightarrow aS_2$
4. $S_1 \rightarrow bS_2$
5. $S_2 \rightarrow cS_2$
6. $S_2 \rightarrow cc$

Închiderea la operaţia de iteraţie

Fie L limbaj de tip 3 (regulat).

Fie $G = (N, T, S, P)$ de tip 3 care genereaza L ($L = L(G)$).

Presupunem ca simbolul de start S nu apare in partea dreapta a vreunei reguli.

Gramatica $G' = (N, T, S, P')$ unde P' consta din

- reguli $A \to uB$ din P ($B \in N$)
- reguli $A \to uS$, pentru orice regula $A \to u$ din P ($u \in T^*$), diferită de $S \to \epsilon$
- regula $S \to \epsilon$

este de tip 3 si generează L^*

Exemplu

$L = \{a^{n_1}b^{m_1}a^{n_2}b^{m_2}\dots a^{n_k}b^{m_k}, n_i, m_i \geq 1 \forall i \in \{1, k\}, k \geq 0\}$

$L = \{a^n b^m, n \geq 1, m \geq 1\}^*$

G :

1. $S \to x$
2. $x \to ax$
3. $x \to ay$
4. $y \to by$
5. $y \to b$

G' :

1. $S \to x$
2. $x \to ax$
3. $x \to ay$
4. $y \to by$
5. $y \to bS$
6. $S \to \epsilon$