

Notiuni de baza

- alfabet
- secventa
 - lungime
 - secv. vida
- concatenare
 - notatii
- subsecventa
 - prefix
 - sufix
- multimi speciale
- proprietati Σ^*
- limbaj
 - cuvant
- tipuri de limbaje
- specificarea unui limbaj
- operatii cu limbaje

Alfabet

- Def:

Alfabet = o multime finită si nevidă de elemente numite simboluri

- Notatie: Σ

Secventa

Def.

- **secventa peste Σ**
o succesiune finita de simboluri din Σ
- **subsecventa**
o succesiune de simboluri consecutive dintr-o secventa

Lungimea unei secvente

- def:
nr. de simboluri din care este formata acea secventa
- notatie
 $|\dots|$
Ex: $|abc| = 3$
- **Secv. vida** = secv. de lungime 0
- notatie: ε (unele surse λ)

Concatenare

Dacă

$$\mathbf{x} = \mathbf{a}_1 \mathbf{a}_2 \dots \mathbf{a}_n$$

$$\mathbf{y} = \mathbf{b}_1 \mathbf{b}_2 \dots \mathbf{b}_m$$

atunci $\mathbf{z} = \mathbf{a}_1 \dots \mathbf{a}_n \mathbf{b}_1 \dots \mathbf{b}_m$

reprezintă **concatenarea** secvențelor \mathbf{x} și \mathbf{y}

și se notează $\mathbf{z} = \mathbf{xy}$

Notatii: $aa \dots aa = a^n$

(de n ori)

Exemplu: $a^2 = aa$

Secventa

Def.

- **secventa peste Σ**
o succesiune finita de simboluri din Σ
- **subsecventa**
o succesiune de simboluri consecutive dintr-o secventa

w_2 – subsecventa a lui w_1

daca \exists secventele u, v a.i. $w_1 = u w_2 v$

Prefix, sufix

Fie x , y , z sunt secvente peste alfabetul Σ

- x este un **prefix** al secventei xy
- y un **sufix** al secventei xy
- **Prefix**: o subsecventa care
 - fie este vida
 - fie incepe cu primul simbol al secventei date
- **Sufix**: o subsecventa care
 - fie este vida
 - fie se termina cu ultimul simbol al secventei date

Multimi speciale

- $\Sigma^n = \{w \mid w \text{ -- secventa peste } \Sigma, |w| = n\}$
- $\Sigma^* = \{w \mid w \text{ -- secventa peste } \Sigma, 0 \leq |w| \}$
- $\Sigma^+ = \{w \mid w \text{ -- secventa peste } \Sigma, 0 < |w| \}$

- $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$

Operatia *

Denumiri:

- Operatia: *, steaua lui Kleene
- Inchiderea lui Kleene

Σ^* - inchiderea alfabetului

- multimea tuturor secventelor ce se pot obtine folosind secvente din Σ

(Similar pentru limbaje)

Σ^* - proprietati

- $\forall w_1, w_2 \in \Sigma^*$ avem: $w_1 w_2 \in \Sigma^*$
- $\forall w_1, w_2, w_3 \in \Sigma^*$ avem: $(w_1 w_2) w_3 = w_1 (w_2 w_3)$
- $\forall w \in \Sigma^*$, $\varepsilon w = w \varepsilon = w$

(Σ^*, \cdot) - monoid

Limбай, cuvаnt

- def: (limбай)

L – limбай peste alfabetul Σ

daca $L \subseteq \Sigma^*$

- def: (cuvаnt)

Cuvаnt al unui limбай – un element al limбайului

Metode de specificare a unui limbaj

- enumerand elementele
- evidentierea unor proprietati ale elementelor
 - folosind multimi si descrieri matematice
 - ...
- folosind gramatici, automate, expresii regulate
- ...

Cateva tipuri de limbaje

Multime & descriere matematica a proprietatilor

- teoretice

$L = \{a^n \mid n \in \mathbb{N}\}$ limbaj peste $\Sigma = \{a\}$

$L = \{a^n b^n \mid n \in \mathbb{N}\}$ limbaj peste $\Sigma = \{a, b\}$

- informatice

limbajul identificatorilor

$\Sigma = \{a, \dots, z, A, \dots, Z, _, 0, \dots, 9\}$

$L = \{a'w' \mid a' \in \{a, \dots, z, A, \dots, Z, _ \}, w' \in \Sigma^*\}$

Enumerarea elementelor

- matematice

ex: limbajul reprezentarii zecimale a numerelor naturale

Operatii cu limbaje (1)

Fie: L_1 – limbaj peste Σ_1 L_2 – limbaj peste Σ_2

(operatii cu multimi)

- $L_1 \cup L_2$ limbaj peste Σ **ales corespunzator**;

de exemplu: $\Sigma = \Sigma_1 \cup \Sigma_2$

- $L_1 \cap L_2$ limbaj peste Σ ($\Sigma = \Sigma_1 \cap \Sigma_2$)

- $L_1 - L_2$ limbaj peste Σ ($\Sigma = \Sigma_1$)

- $L_1 L_2$ limbaj peste Σ ($\Sigma = \Sigma_1 \cup \Sigma_2$)

(operatii bazate pe concatenare)

- câțul la dreapta: $L_1 / L_2 = \{ w \in \Sigma^* \mid \exists y \in L_2: wy \in L_1 \}$
- câțul la stânga: $L_1 \setminus L_2 = \{ w \in \Sigma^* \mid \exists y \in L_2: yw \in L_1 \}$

Operatii cu limbaje (2)

- L limbaj peste un alfabet Σ
- complementara: $\overline{L} = \{x \in \Sigma^* \mid x \notin L\}$

- închiderea reflexivă și tranzitivă:

$$L^* = \bigcup_{n \geq 0} L^n \quad \text{unde } L^n = LL^{n-1}, \quad L^0 = \{\epsilon\};$$

- închiderea tranzitivă:

$$L^+ = \bigcup_{n \geq 1} L^n \quad \text{sau } L^+ = LL^*, \quad L^* = L^+ \cup \{\epsilon\}$$

Gramatica independenta de context

O gramatica independenta de context este un cvadruplu

$$G = (N, \Sigma, P, S)$$

- N este un alfabet de simboluri *neterminale*
- Σ este un alfabet de simboluri *terminale*
- $N \cap \Sigma = \emptyset$
- $P \subseteq N \times (N \cup \Sigma)^*$
 P multime finită (multimea regulilor de productie)
- $S \in N$ (simbolul de start - simbolul initial)

Notatie:

$(\alpha, \beta) \in P$ se noteaza: $\alpha \rightarrow \beta$

(α se înlocuieste cu β)

Folosita pentru a
specifica un limbaj

Doua tipuri de gramatici

- Gramatica independenta de context:
reg. productie sunt de forma $A \rightarrow \alpha$, $A \in N$, $\alpha \in (N \cup \Sigma)^*$
- Gramatica regulara:
reg. prod. sunt de forma
 - $A \rightarrow aB$
 - $A \rightarrow b$ unde $A, B \in N$ si $a, b \in \Sigma$caz special: $S \rightarrow \epsilon$ poate $\in P$ In acest caz S nu apare în membrul drept al nici unei reguli de productie.

Backus Naur Form

- primitive (terminale)
- variabile metalingvistice : intre paranteze unghiulare
- conective metalingvistice: ::= , |

$\langle \text{intreg_fara_seamn} \rangle ::= \langle \text{cifra} \rangle \mid$
 $\qquad \qquad \qquad \langle \text{cifra} \rangle \langle \text{intreg_fara_seamn} \rangle$

$\langle \text{cifra} \rangle ::= 0 \mid 1 \mid \dots \mid 9$

Extended Backus Naur Form

- primitive (terminale): între “ghilimele”
- variabile metalingvistice
- conective metalingvistice: $=$, $|$,
[optional], { de oricate ori }
- punctul este folosit ca sfarsit de regula

intreg_fara_semn = cifra { cifra } .

cifra = “0” | “1” | ... | “9” .