

Elemente de limbaje formale. Instrumente pentru reprezentare

- ▶ Gramaticile formale, in particular gramaticile independente de context, sunt uneltele cele mai utilizate pentru a reprezenta clar structura programelor, sub forma arborilor de derivare.
- ▶ Pornind de la gramatici, se pot specifica automate care accepta programe concrete.
- ▶ Automatele pot fi modificate pentru a genera o codificare acceptabila din arborii de derivare.

Outline

Siruri si sisteme de rescriere

Gramatici

Ierarhia lui Chomsky

Arbori de derivare

Siruri si sisteme de rescriere

- ▶ Un limbaj = set de stringuri
- ▶ Definirea formala a limbajului = raspuns formal pt “Care stringuri sunt admise de catre Limbaj?”

Siruri si sisteme de rescriere cont

Alfabet (vocabulary) V - un set de simboluri

ex: $V=1,2,3,4,5,6,7,8,9,0$

String un string peste alfabetul V = secventa(sir finit) de simboluri din alfabetul V

ex: 2018

Stringul vid: ϵ

$$\epsilon\chi = \chi\epsilon = \chi$$

V^* - multimea tuturor stringurilor peste V

$$V^+ = V^* \setminus \{\epsilon\}$$

Limbaj L peste V este orice subset al lui V^*

Propozitii - elementele limbajului

Numarul de propozitii dintr-un limbaj poate fi infinit

Exemple:

$$V = \{a, b, c, \dots a\}; \quad L = \{\text{cuvintele limbii engleze}\}$$

$$V = \{0, 1\}; \quad L = \{\varepsilon, 01, 010, 0101, 01010, 010101, \dots\}$$

Cum putem defini propozitiile unui limbaj? Ne trebuie o reprezentare formală

proces de generare

Derivare

Relatie de derivare \Rightarrow^+ binara, tranzitiva pe V^*

$$L = \{\chi | \zeta \Rightarrow^+ \chi, \zeta \text{ un anumit sir din } V^*\}$$

Sistem formal (V, \Rightarrow^+)

definire derivare prin enumerare?? – NU

Productii Un sir finit de perechi (σ, τ) de siruri din V^*

Definesc relatia de derivare

Generam un string pornind de la alt string

Inchiderea tranzitiva a relatiei finite descrise de catre productii =
relatia de derivare

Derivare cont.

Sistem de rescriere (V, P) , V vocabular, P set finit de productii

$$\sigma \rightarrow \tau, \sigma, \tau \in V^*$$

Derivare directa \Rightarrow Un sir χ este derivabil direct din π : $\pi \Rightarrow \chi$ daca exista sirurile $\sigma, \tau, \mu, \nu \in V^*$ a.i.

$$\sigma \rightarrow \tau \in P,$$

$$\pi = \mu\sigma\nu,$$

$$\chi = \mu\tau\nu.$$

Derivare \Rightarrow^+ Un sir χ este derivabil din sirul π : $\pi \Rightarrow^+ \chi$ daca exista sirurile $\rho_0, \dots, \rho_n \in V^*, n \geq 1$ a.i.

$$\pi = \rho_0, \chi = \rho_n,$$

$$\rho_{i-1} \Rightarrow \rho_i, i = \overline{1, n}.$$

Secventa ρ_0, \dots, ρ_n = derivare de lungime n .

χ reductibil direct la π daca χ derivabil direct din π

Expresii aritmetice: Fie sistemul de rescriere (V, P)

unde

- ▶ $V = \{+, *, (,), i, E, T, F\}$
- ▶ cu productiile P
 1. $E \rightarrow T$
 2. $E \rightarrow E + T$
 3. $T \rightarrow F$
 4. $T \rightarrow T * F$
 5. $F \rightarrow i$
 6. $F \rightarrow (E)$

Derivari cu lungimea lor:

$$E \Rightarrow T \quad 1$$

$$T \Rightarrow T * F \quad 1$$

$$T * F \Rightarrow T * i \quad 1$$

$$E \Rightarrow^* T * i \quad 3$$

$$TiE \Rightarrow^* iii \quad 5$$

$$E \Rightarrow i + i * i \quad 8$$

Outline

Siruri si sisteme de rescriere

Gramatici

Ierarhia lui Chomsky

Arbori de derivare

Gramatici

Definitie generativa a unui limbaj: Un quadruplu (T, N, Z, P) este o gramatica pentru limbajul $L(G)$

$$L(G) = \{\chi \in T^* \mid Z \Rightarrow^+ \chi\}$$

daca

- ▶ T si N disjuncte, formeaza impreuna vocabularul
- ▶ $(T \cup N, P)$ este un sistem de rescriere
- ▶ $Z \in N$

Doua gramatici sunt echivalente daca $L(G_1) = L(G_2)$

- ▶ T - multimea terminalelor
- ▶ N - multimea nonterminalelor sau a variabilelor sintactice
- ▶ Z - nonterminal anume, simbol de start
- ▶ in limbaj - sirurile derivabile din simbolul de start si care constau doar din terminalelor

Gramatica G genereaza limbajul L

Exemplul 1

Fie gramatica $G = (\{0, 1\}, \{S\}, S, P = \{S \rightarrow 0S1, S \rightarrow \varepsilon\})$

Care dintre stringurile de mai jos $\in L(G)$?

- ▶ 01
- ▶ 010
- ▶ 1
- ▶ 01010
- ▶ ε
- ▶ $0^n 1^n$

Exemplul 1

Fie gramatica $G = (\{0, 1\}, \{S\}, S, P = \{S \rightarrow 0S1, S \rightarrow \varepsilon\})$
Care dintre stringurile de mai jos $\in L(G)$?

- ▶ 01
- ▶ 010
- ▶ 1
- ▶ 01010
- ▶ ε
- ▶ $0^n 1^n$

$$S \Rightarrow^* 0^n 1^n$$

$$L(G) = \{0^n 1^n \mid n \geq 0\}$$

Exemplul 2

Fie gramatica $G = (T, N, S, P)$ unde

- ▶ $T = \{a, b\}$
- ▶ $N = \{S, A, B\}$
- ▶ cu productiile P
 1. $S \rightarrow AB$
 2. $A \rightarrow aA$
 3. $A \rightarrow \varepsilon$
 4. $B \rightarrow bB$
 5. $B \rightarrow \varepsilon$

Care stringuri apartin limbajului $L(G)$?

- ▶ ε
- ▶ a
- ▶ b
- ▶ $aaabb$

Exemplul 2

Fie gramatica $G = (T, N, S, P)$ unde

- ▶ $T = \{a, b\}$
- ▶ $N = \{S, A, B\}$
- ▶ cu productiile P
 1. $S \rightarrow AB$
 2. $A \rightarrow aA$
 3. $A \rightarrow \varepsilon$
 4. $B \rightarrow bB$
 5. $B \rightarrow \varepsilon$

Care stringuri apartin limbajului $L(G)$?

- ▶ ε
- ▶ a
- ▶ b
- ▶ $aaabb$

$$\begin{aligned} S &\Rightarrow^1 AB \Rightarrow^2 aAB \Rightarrow^2 aaAB \Rightarrow^2 aaaAB \\ &\Rightarrow^3 aaaB \Rightarrow^4 aaabB \Rightarrow^4 aaabbB \Rightarrow^5 aaabb \end{aligned}$$

Exemplul 3

Fie gramatica $G = (T, N, S, P)$ unde

- ▶ $T = \{a\}$
- ▶ $N = \{S, N, Q, R\}$
- ▶ cu productiile P
 1. $S \rightarrow QNQ$
 2. $QN \rightarrow QR$
 3. $RN \rightarrow NNR$
 4. $RQ \rightarrow NNQ$
 5. $N \rightarrow a$
 6. $Q \rightarrow \varepsilon$

Expresii aritmetice:

$$G_1 = (T, N, E, P)$$

- ▶ $T = \{+, *, (,), i\}$
- ▶ $N = \{E, T, F\}$
- ▶ cu productiile P
 1. $E \rightarrow T$
 2. $E \rightarrow E + T$
 3. $T \rightarrow F$
 4. $T \rightarrow T * F$
 5. $F \rightarrow i$
 6. $F \rightarrow (E)$

derivare pt i; dar pt i*i?

Expresii aritmetice: Gramatici echivalente

Doua gramatici sunt **echivalente** daca $L(G_1) = L(G_2)$

$G_1 = (T, N, E, P)$

- ▶ $T = \{+, *, (,), i\}$
- ▶ $N = \{E, T, F\}$
- ▶ cu productiile P
 1. $E \rightarrow T$
 2. $E \rightarrow E + T$
 3. $T \rightarrow F$
 4. $T \rightarrow T * F$
 5. $F \rightarrow i$
 6. $F \rightarrow (E)$

$G_2 = (T, N, E, P)$

- ▶ $T = \{+, *, (,), i\}$
- ▶ $N = \{E, E', T, T', F\}$
- ▶ cu productiile P
 1. $E \rightarrow T$
 2. $E \rightarrow Te'$
 3. $E' \rightarrow +T$
 4. $E' \rightarrow +TE'$
 5. $T \rightarrow F$
 6. $T \rightarrow FT'$
 7. $T' \rightarrow *F$
 8. $T' \rightarrow *FT'$
 9. $F \rightarrow i$
 10. $F \rightarrow (E)$

Outline

Siruri si sisteme de rescriere

Gramatici

Ierarhia lui Chomsky

Arbori de derivare

Ierarhia lui Chomsky

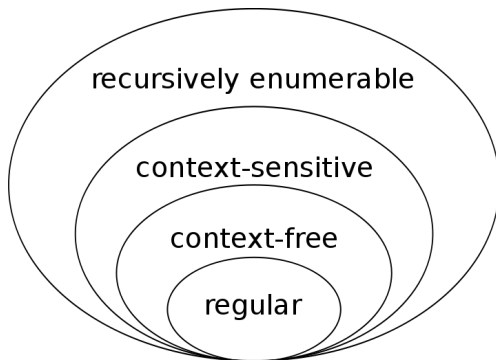
Conceptul de **Clasificarea gramaticilor** - 1950, **Noam Chomsky** - parintele lingvisticii moderne, unul dintre fondatorii stiintelor cognitive

- Modalitate de a descrie complexitatea structurala a unor propozitii particulare din limbajul natural

- ▶ Limbajele clasificate in functie de gramatica care le genereaza: **constrangeri asupra productiilor gramaticii definesc diferite clase de gramatici/limbaje**

Ierarhia lui Chomsky

- ▶ Gramatica de tip 0
- ▶ Gramatica de tip 1 - **dependenta de context**
- ▶ Gramatica de tip 2 - **independenta de context**
- ▶ Gramatica de tip 3 - **regulata**



Gramatica de Tip 0

$$G = (T, N, Z, P)$$

- ▶ cele mai generale gramatici
- ▶ fiecare productie are forma

$$\sigma \rightarrow \tau, \sigma \in V^+, \tau \in V^*$$

Gramatica din exemplul 3 este de Tip 0 (si nu si de tip 1,2,3)

$$RN \rightarrow NNR$$

$$RQ \rightarrow NNQ$$

Gramatica de Tip 1 - Dependente de context

$$G = (T, N, Z, P)$$

- fiecare productie are forma

$$\mu X \nu \rightarrow \mu \chi \nu, \quad \mu, \nu \in V^*, X \in N, \chi \in V^+$$

Context-sensitive (dependenta de context) - contextul lui X

Gramatica de Tip 1 - Dependente de context

$$G = (T, N, Z, P)$$

- fiecare productie are forma

$$\mu X \nu \rightarrow \mu \chi \nu, \quad \mu, \nu \in V^*, X \in N, \chi \in V^+$$

Context-sensitive (dependenta de context) - contextul lui X

Gramatica de Tip 2 - Independentă de context

$$G = (T, N, Z, P)$$

- ▶ fiecare producție are forma

$$X \rightarrow \chi, X \in N, \chi \in V^*$$

Gramatica de Tip 2 - Independentă de context

$$G = (T, N, Z, P)$$

- ▶ fiecare producție are forma

$$X \rightarrow \chi, X \in N, \chi \in V^*$$

- ▶ Context-free grammars - suficient de puternice pentru a descrie sintaxa limbajelor de programare
- ▶ permit construirea unor algoritmi eficienți de parsare:
determină dacă un string este sau nu generat din gramatica

gramatica din Exemplul 1 este Context-free grammar.

$$S \rightarrow 0S1 \text{ DA}$$

$$QN \rightarrow QR \text{ și } RN \rightarrow NNR \text{ NU}$$

Gramatica de Tip 3 - Regulate

$$G = (T, N, Z, P)$$

- fiecare productie are forma

$$X \rightarrow t, \quad X \in N, \quad t \in T \cup \{\varepsilon\}$$

sau

$$X \rightarrow tY, \quad X, Y \in N, \quad t \in T$$

Gramatica de Tip 3 - Regulate

$$G = (T, N, Z, P)$$

- ▶ fiecare productie are forma

$$X \rightarrow t, \quad X \in N, \quad t \in T \cup \{\varepsilon\}$$

sau

$$X \rightarrow tY, \quad X, Y \in N, \quad t \in T$$

- ▶ Regular grammars: de obicei folosite pt a defini structura lexicala a limbajelor de programare

Ierarhia lui Chomsky - rezumat

tip 0

$$\sigma \rightarrow \tau$$

$$\sigma \in V^+, \tau \in V^*$$

*dependenta de
context*

$$\mu X \nu \rightarrow \mu \chi \nu$$

$$\mu, \nu \in V^*, X \in N, \chi \in V^+$$

*independenta de
context*

$$X \rightarrow \chi$$

$$X \in N, \chi \in V^*$$

regulata

$$X \rightarrow t,$$

$$X \rightarrow tY,$$

$$X \in N, t \in T \cup \{\varepsilon\} \text{ sau}$$

$$X, Y \in N, t \in T$$

Ierharhia Chomsky - concluzii

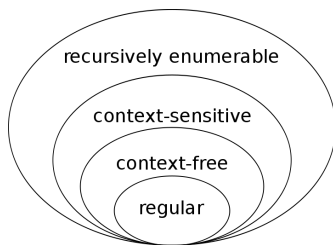
- ▶ productia $S \rightarrow \varepsilon$ productii ε
Admise in gramaticile independente de context, regulate - limbajele se pot descrie printr-o gramatica si fara productia ε
- ▶ Every Regular Language is Context-Free, every Context-Free Language is Context-Sensitive and every Context-Sensitive Language is a Type 0 Language.
- ▶ fiecare simbol din vocabular apare in derivarea cel putin a unei propozitii; (nu exista simboluri inutile)

Ierarhia lui Chomsky - concluzii

- ▶ Gramatica de tip 0
- ▶ Gramatica de tip 1 - **dependenta de context**
- ▶ Gramatica de tip 2 - **independenta de context**
- ▶ Gramatica de tip 3 - **regulata**

Pt compilatoare: gramaticile regulate si independente de context

- ▶ Simboluri fundamentale ale limbajului (identificatori, constante..): gramatici regulate
- ▶ Structura programului: gramatici independente de context



Outline

Siruri si sisteme de rescriere

Gramatici

Ierarhia lui Chomsky

Arbori de derivare

Derivare si arbori - G1 - expresii aritmetice

$P = (E \rightarrow T, E \rightarrow E + T, T \rightarrow F, T \rightarrow T * F, F \rightarrow i, F \rightarrow (E))$

Derivari pentru $i+i*i$

Derivare stanga	Arbitrar	Derivare dreapta
E	E	E
$E + T$	$E + T$	$E + T$
$T + T$	$E + T * F$	$E + T * F$
$F + T$	$T + T * F$	$E + T * i$
$i + T$	$T + F * T$	$E + F * i$
$i + T * F$	$T + F * i$	$E + i * i$
$i + F * F$	$F + F * i$	$T + i * i$
$i + i * F$	$i + F * i$	$F + i * i$
$i + i * i$	$i + i * i$	$i + i * i$

Derivare si arbori - G1 - expresii aritmetice

$P = (E \rightarrow T, E \rightarrow E + T, T \rightarrow F, T \rightarrow T * F, F \rightarrow i, F \rightarrow (E))$

Derivari pentru $i+i*i$

Derivare stanga	Arbitrar	Derivare dreapta
E	E	E
$E + T$	$E + T$	$E + T$
$T + T$	$E + T * F$	$E + T * F$
$F + T$	$T + T * F$	$E + T * i$
$i + T$	$T + F * T$	$E + F * i$
$i + T * F$	$T + F * i$	$E + i * i$
$i + F * F$	$F + F * i$	$T + i * i$
$i + i * F$	$i + F * i$	$F + i * i$
$i + i * i$	$i + i * i$	$i + i * i$

La ce se refera Structura conferita de gramatica unui sir?

- ▶ ? Secventa pasilor de derivare
- ▶ ? Relatia ce arata din ce subsir este derivat un anumit nonterminal

$i*i$ este derivat tot timpul din T

Derivare si arbori - G1 - expresii aritmetice

$P = (E \rightarrow T, E \rightarrow E + T, T \rightarrow F, T \rightarrow T * F, F \rightarrow i, F \rightarrow (E))$

Derivari pentru $i+i*i$

Derivare stanga	Arbitrar	Derivare dreapta
E	E	E
$E + T$	$E + T$	$E + T$
$T + T$	$E + T * F$	$E + T * F$
$F + T$	$T + T * F$	$E + T * i$
$i + T$	$T + F * T$	$E + F * i$
$i + T * F$	$T + F * i$	$E + i * i$
$i + F * F$	$F + F * i$	$T + i * i$
$i + i * F$	$i + F * i$	$F + i * i$
$i + i * i$	$i + i * i$	$i + i * i$

La ce se refera Structura conferita de gramatica unui sir?

- **NU** Secventa pasilor de derivare
- **DA** Relatia ce arata din ce subsir este derivat un anumit nonterminal

$i*i$ este derivat tot timpul din T

Derivare cont.

- ▶ $T \Rightarrow^+ i * i$ - unitate semantica: operatorul $*$ se aplica operanzilor i
- ▶ gramatica - **structura semantica relevanta** fiecarei propozitii din limbaj
- ▶ Daca $E \rightarrow E + T, T \rightarrow T * F$ schimbam in $E \rightarrow E * T, T \rightarrow T + F$
multimea de siruri va fi aceeaasi cu G_1 , dar structura propozitiilor va fi alta: adunarile mai prioritare decat inmultirile

Derivari cont.

Fie gramatica $G = (T, N, Z, P)$.

- ▶ Sirul $\chi \in V^+$ este o fraza pentru X a lui $\mu\chi\nu$ daca si numai daca

$$Z \Rightarrow^* \mu X \nu \Rightarrow^* \mu \chi \nu$$

unde $\mu, \nu \in V^*, X \in N$

- ▶ Sirul $\chi \in V^+$ este o fraza simpla a lui $\mu\chi\nu$ daca si numai daca

$$Z \Rightarrow^* \mu X \nu \Rightarrow \mu \chi \nu$$

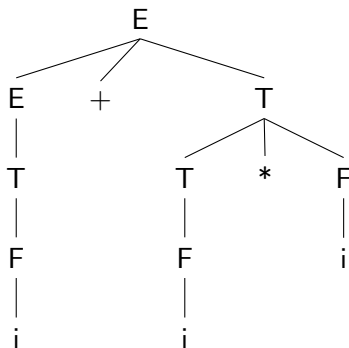
Subsirurile derivate din nonterminale singulare se numesc fraze.

Obs: fraza nu consta numai din terminale:

$E \Rightarrow^* E + T \Rightarrow^* E + F * F$. Deci $F * F$ derivat din T

Set de fraze - Arborele de derivare

- ▶ Toate cele trei derivari pt gramatica expresiilor aritmetice sunt echivalente: confera acelasi set de fraze.
- ▶ Arborele de derivare - reprezentarea intregului set de derivari echivalente; structura frazala



- ▶ din arbore de parsare:
orice sir din orice derivare
a unei propozitii -
taietura = nr minim de
noduri care intrerup
calea de la radacina catre
frunze
- ▶ exemplu: T, +, T, *, F,
E, +, T

Arbori de derivare

- ▶ parse tree - metoda vizuala de a descrie orice derivare dintr-o gramatica independenta de context (Context-free grammar CFG)
- ▶ fiecare nod are un label
- ▶ radacina este simbolul de start al gramaticii
- ▶ daca un nod n , etichetat cu A are cel putin un descendent, A este in N
- ▶ daca nodurile n_1, n_2, \dots, n_k sunt descendentii unui nod n , cu etichetele A_1, A_2, \dots, A_k atunci

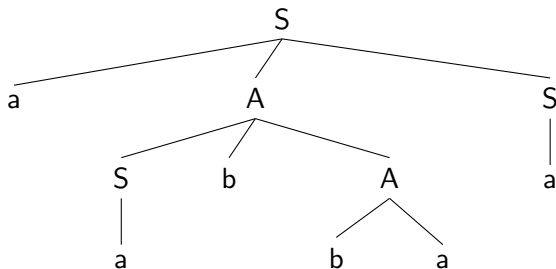
$$A \rightarrow A_1, A_2, \dots, A_k$$

este o productie in P

Exemplu

Fie $G = (\{a, b\}, \{S, A\}, S, P)$

- ▶ $S \rightarrow aAS$
- ▶ $S \rightarrow a$
- ▶ $A \rightarrow SbA$
- ▶ $A \rightarrow ba$
- ▶ $A \rightarrow SS$



Exemplu

Fie $G = (\{a, b\}, \{S, A\}, S, P)$

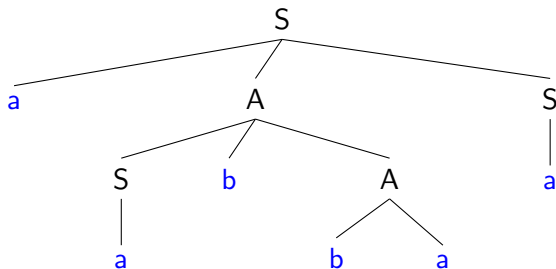
► $S \rightarrow aAS$

► $S \rightarrow a$

► $A \rightarrow SbA$

► $A \rightarrow ba$

► $A \rightarrow SS$



$S \Rightarrow^* aabbbaa$

$S \rightarrow aAS \rightarrow aSbAS \rightarrow aabAS \rightarrow aabbbaS \rightarrow aabbbaa$

Stanga la dreapta frunzele: propozitie (rezultatul arborelui de derivare)

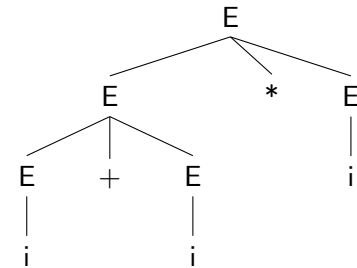
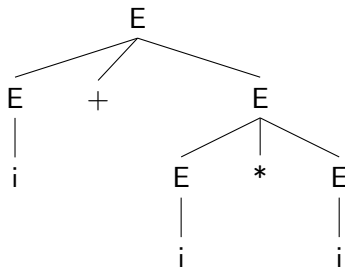
Ambiguity

Let $G_4 = (\{+, *, i\}, \{E\}, E, P)$

► $E \rightarrow E + E$

► $E \rightarrow E * E$

► $E \rightarrow i$



$2+3*4??$

Ambiguitate

“Look at the dog with one eye”

- ▶ O propozitie este ambigua daca derivarile sale pot fi descrise prin cel putin doi arbori de derivare distincti.
- ▶ O gramatica este ambigua daca in limbajul generat exista cel putin o propozitie ambigua
- ▶ O gramatica este ambigua daca genereaza mai mult de o derivare cea mai din stanga pentru vreo propozitie

Rezumat

Siruri si sisteme de rescriere

Gramatici

Ierarhia lui Chomsky

Arbori de derivare

Exemplu

Fie $G = (\{the, a, reads, walks, kid, robot\}, \{S, NounPhrase, Predicate, Article, Noun, Verb\}, S, P)$

- ▶ $S \rightarrow NounPhrase Predicate$
- ▶ $NounPhrase \rightarrow Article Noun$
- ▶ $Predicate \rightarrow Verb$
- ▶ $Article \rightarrow the$
- ▶ $Article \rightarrow a$
- ▶ $Verb \rightarrow reads$
- ▶ $Verb \rightarrow walks$
- ▶ $Noun \rightarrow kid$
- ▶ $Noun \rightarrow robot$