

Seminar 7. Sugestii si rezolvări

1 Limbaje si gramatici de toate felurile

1. Definiti (folosind multimi) limbajul secventelor de simb. a care apar in numar par. Dati o gramatica regulara, una independenta de context si una ce nu e independenta de context care genereaza acest limbaj.

2. Sa se dea cate o gramatica care genereaza limbajele:

1. $L = \{ww \mid w \in \{a, b\}^*\}$
2. $L = \{wxw \mid w \in \{a, b\}^+, x \in \{a, b\}^*\}$
3. $L = \{a^n b^n c^n \mid n \in \mathbb{N}^*\}$
4. $L = \{a^n b^n c^n d^n \mid n \in \mathbb{N}^*\}$
5. $L = \{a^{2^n} \mid n \in \mathbb{N}\}$ a apare de 2^n ori
6. $L = \{w \mid w \in \{a, b, c\}^*, nr_a(w) = nr_b(w) = nr_c(w)\}$
7. $L = \{a^n b^n c^m d^m \mid n, m \in \mathbb{N}\}$
8. $L = \{a^n b^m c^m d^n \mid n, m \in \mathbb{N}\}$
9. $L = \{a^n b^m c^k \mid n, m, k \in \mathbb{N}, (n=m) \text{ sau } (m=k)\}$
10. $L = \{a^n b^m c^k \mid n, m, k \in \mathbb{N}, m+n = k\}$
11. $L = \{w \in \{a, b\}^* \mid w \text{ incepe si se termina cu acelasi simbol}\}$

2.1 $L = \{ww \mid w \in \{a, b\}^*\}$

Limbajul nu este independent de context.

O gramatica ce il genereaza este data de urmatoarele reguli de productie:

$S \rightarrow aSA$
 $S \rightarrow bSB$ $\Rightarrow a..b ..$ in oglinda cu $A..B..$
Dupa care inversam secv. $A B$
Obligam A, B sa vina pana la mijloc (unde e D)
trecand numai peste terminale (\sim acestea sunt inversate)
Obligatia vine din faptul ca A si B devin terminale doar langa D

$S \rightarrow D$
 $aB \rightarrow Ba$
 $aA \rightarrow Aa$
 $bA \rightarrow Ab$
 $bB \rightarrow Bb$
 $DA \rightarrow Da$
 $DB \rightarrow Db$
 $D \rightarrow \varepsilon$

2.2 $L = \{wxw \mid w \in \{a, b\}^+, x \in \{a, b\}^*\}$

ideea: adaugam un nou neterminat X . D se transforma in X . X poate introduce orice secventa de a si b . Dupa transformarea lui D in X , A si B nu mai pot fi transformate in terminale. Astfel

$S \rightarrow aSA$
 $S \rightarrow bSB$
 $S \rightarrow D$
 $aB \rightarrow Ba$
 $aA \rightarrow Aa$
 $bA \rightarrow Ab$
 $bB \rightarrow Bb$
 $DA \rightarrow Da$
 $DB \rightarrow Db$
 $D \rightarrow X$
 $X \rightarrow Xa$
 $X \rightarrow Xb$
 $X \rightarrow \varepsilon$

2.3 $L = \{a^n b^n c^n \mid n \in \mathbb{N}^*\}$

$S \rightarrow aSBc$
 $S \rightarrow abc$
 $cB \rightarrow Bc$
 $bB \rightarrow bb$

2.5 $L = \{a^{2^n} \mid n \in \mathbb{N}\}$

a apare de 2^n ori

$S \rightarrow LaR$
 $L \rightarrow LD$
 $DR \rightarrow R$
 $Da \rightarrow aaD$
 $L \rightarrow \varepsilon$
 $R \rightarrow \varepsilon$

A trece de la L(left) pana la R(right) dubland a deja existenti

2.6 $L = \{w \mid w \in \{a, b, c\}^*, nr_a(w) = nr_b(w) = nr_c(w)\}$

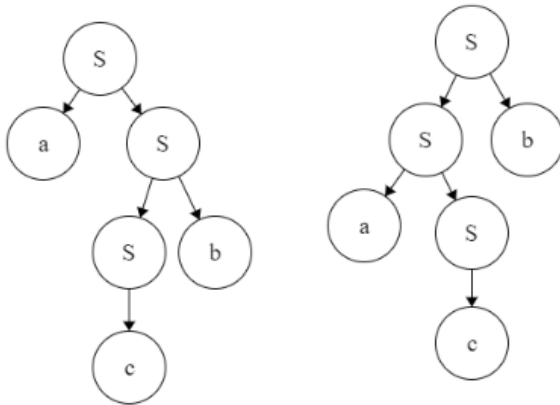
$S \rightarrow ABCS$
 $S \rightarrow \varepsilon$
 $AB \rightarrow BA$
 $BA \rightarrow AB$
 $AC \rightarrow CA$
 $CA \rightarrow AC$
 $BC \rightarrow CB$
 $CB \rightarrow BC$
 $A \rightarrow a$
 $B \rightarrow b$
 $C \rightarrow c$

2 Ambiguitate in gramatici independente de context

1. Sa se arate ca gramaticile urmatoare sint ambigue si sa se gaseasca o gramatica echivalenta neambigua.

- a) $S \rightarrow aS \mid Sb \mid c$
- b) $S \rightarrow \text{if } b \text{ then } S \text{ else } S \mid \text{if } b \text{ then } S \mid \text{stmt}$
- c) $S \rightarrow (S \mid S) \mid (S) \mid 1$

a) Observam ca putem da 2 arbori de derivare diferiti pentru secventa: acb



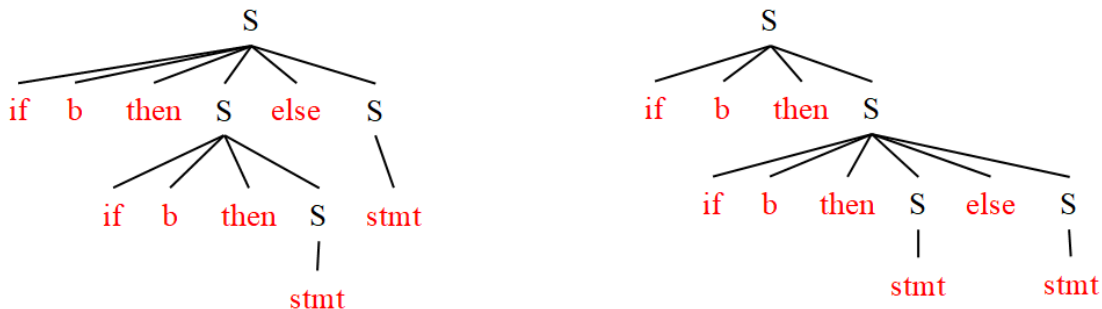
De aici rezulta ca gramatica este ambigua.

Pentru a elimina ambiguitatea (adica a avea 2 alegeri la un moment dat, care ne duc la rezultat), vom crea reguli de productie care impun ca, pentru a ajunge la rezultat, prima data sa fie generate toate simbolurile a necesare, dupa care vor putea fi introduse toate simbolurile b necesare.

Gramatică neambiguă echivalentă este data prin regulile de productie:

- $S \rightarrow c$
- $S \rightarrow aS$
- $S \rightarrow cB$
- $B \rightarrow b$
- $B \rightarrow bB$

b) Observam ca putem da 2 arbori de derivare diferiti pentru secventa:
if b then if b then stmt else stmt



De aici rezulta ca gramatica este ambigua.

In domeniul limbajelor de programare, aceasta problema se numeste problema atasarii lui else (dangling else). Astfel, cand avem structuri imbricate de tip if-then-else, specificarile limbajelor vin cu o precizare suplimentara, si anume ca else tine de cel mai interior then posibil. Vom construi o gramatica echivalenta neambigua fortand "else" sa tina de cel mai interior then posibil; astfel, vom permite doar structuri if-then-else complete intre oricare then - else posibili.

Gramatică neambiguă echivalentă este data prin regulile de productie:

$$\begin{aligned} S &\rightarrow \text{if } b \text{ then } S' \text{ else } S \\ S &\rightarrow \text{if } b \text{ then } S \\ S &\rightarrow \text{stmt} \\ S' &\rightarrow \text{if } b \text{ then } S' \text{ else } S' \\ S' &\rightarrow \text{stmt} \end{aligned}$$

c)

Problema este similara cu cea de la pct. a

Vom da doar o gramatica echivalenta neambigua:

$$\begin{aligned} S &\rightarrow (S \\ S &\rightarrow 1 \\ S &\rightarrow 1A \\ A &\rightarrow A) \\ A &\rightarrow) \end{aligned}$$

3. Reprezentari pentru gramatici independente de context

Fie gramatica:

$$G = (\{E, T, F\}, \{a, +, *, (,)\}, P, E)$$

P:

$$\begin{aligned} E &\rightarrow E + T \\ E &\rightarrow T \\ T &\rightarrow T * F \\ T &\rightarrow F \\ F &\rightarrow (E) \\ F &\rightarrow a \end{aligned}$$

Pentru gramatica data, ilustrati modurile de reprezentare:

- Folosind liste liniare (de exemplu: vectoriala)
- reprezentarea cu ajutorul unor liste inlantuite ramificate
 - Liste dublu inlantuite
 - Liste triplu inlantuite

Se dă gramatica independentă de context $G = \{S, A, B\}, \{a, b, c\}, P, S)$

unde: $P = \{S \rightarrow AB, A \rightarrow aAb \mid ab, B \rightarrow cB \mid c\}$.

Să se reprezinte sub formă de vectori, listă ramificată și tablou gramatica dată.

1) vectorial;

Vom construi vectorii:

N:

S	A	B	\$
---	---	---	----

Σ

a	b	c	\$
---	---	---	----

P:

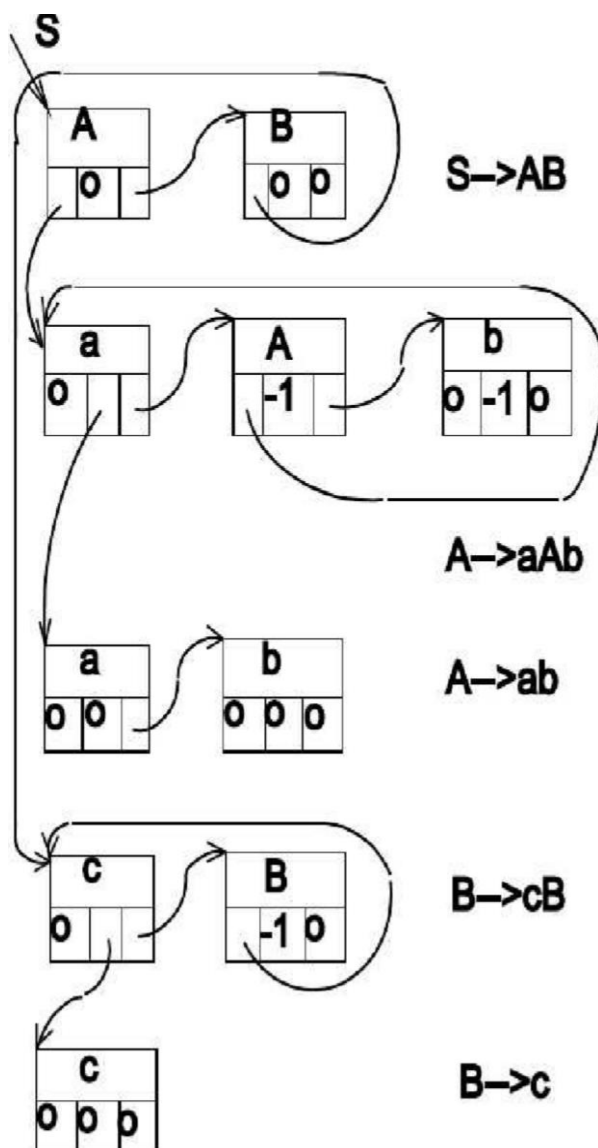
S	A	B	#	A	a	A	b		a	b	#	B	c	B		c	#	\$
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---	--	---	---	----

Am marcat sfârșitul șirului de caractere ce definesc cei trei vectori cu \$, iar sfârșitul regulilor de producție cu același membru stâng, cu #.

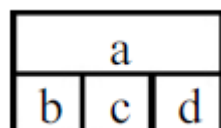
2) listă ramificată înlănțuită

Pentru fiecare membru drept al fiecărei producții se construiește o sublistă liniară.

Pentru gramatica dată în problemă avem următoarea reprezentare:



Nodurile listei sunt de forma:



unde câmpurile au următoarele semnificații:

a - simbolul terminal sau neterminal al gramaticii (numai din membrul drept al fiecărei producții);

b - pointer sau întreg:

b = 0 dacă $a \in \Sigma$;

b = pointer spre capul primei subliste atașate regulii de producție cu membrul stâng a.

c - pointer sau întreg;

Considerăm următoarele propoziții:

p - a este primul simbol al membrului doi dintr-o producție;

