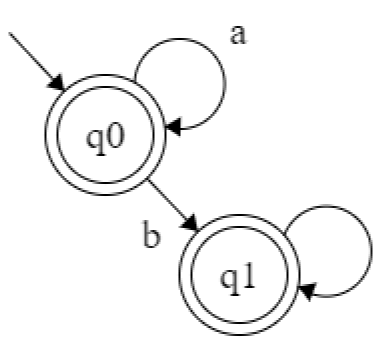
Automate push-down = un Automat Finit + stivă
Stiva –pop & push (nu există peek).
Pt. a vedea ce e în vf. Stivei, se scoate vf. Stivei

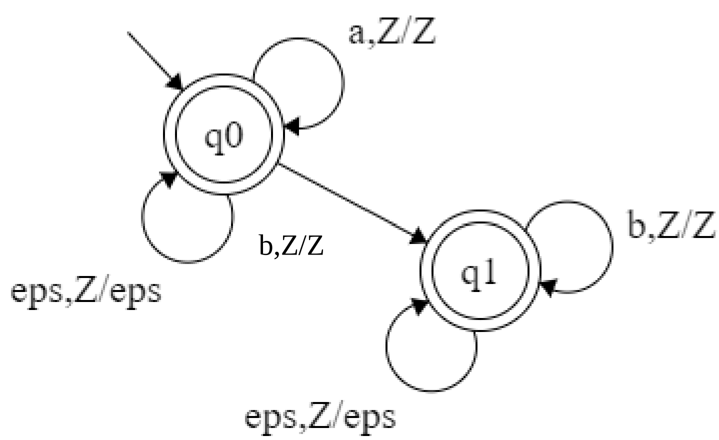
1. Construiti APD care accepta urmatoarele limbaje dupa criteriul stivei vide:

a) $L = \{a^n b^{2n} \mid n>=0\}$
b) $L = \{a^n b^m \mid m,n >=0\}$
c) $L = \{a^n b^m \mid n >= m >= 0\}$
d) $L = \{a^m b^n \mid n >= m >= 0\}$
e) $L = \{w w^{tilda} \mid w \in \{a,b\}^*, w^{tilda}$ este inversul lui $w\}$
f) $L = \{w \mid w \in \{a, b\}^*, nr_a(w) = nr_b(w) \}$
g) $L = \{a^{2n} b^{2n} \mid n>=0\}$

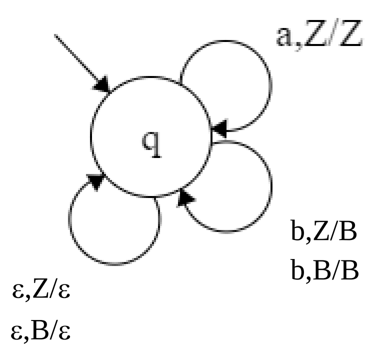
h) $L = \{a^n b^n \mid n>=0\} \cup \{b^n a^n \mid n>=0\}$
i) $L = \{a^n b^n \mid n>=0\} \cup \{ a^n b2^n \mid n>=1\}$
j) $\{w\ \ x \mid w^{tilda}$ is a substring of x, where $x \in \{a, b\}^*$, $w \in \{a, b\}^*$, $|w|>=1 \}$

b) e limbaj regular, deci acceptă un Automat Finit (AF)





putem să adăugăm stiva doat pt. a-l transfortma într-un Automat Push-Down (APD)
Su putem să avem un APD cu o singură stare și în stivă să trecem de la a la b



Verificăm cu ultimul APD dacă secvența aabbb $\in$ L(M)

(q,aabbb,Z) |– (q,abbb,Z) |– (q,bbb,Z) |– (q,bb,B) |– (q,b,B) |– (q,ε,B) |– (q,ε,ε) $\Rightarrow$ aabbb $\in$ L(M) conform criteriului stivei vide

s-a epuizat banda de intrare și s-a golit stiva

1.2 GIC: a fi sau a nu fi ambigua ☺

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

a) S-> aS | Sb | c
b) S -> if b then S else S | if b then S | stmt
c) S -> (S | S) | (S) | 1

1 a)

$S \rightarrow aS \mid Bb \mid c$

$B \rightarrow Bb \mid c$

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. $L = \{a^n b^n c^n \mid n \in \mathbf{N}^*\}$
4. $L = \{a^n b^n c^n d^n \mid n \in \mathbf{N}^*\}$
5. $L = \{a\hat{\ }2^n \mid n \in \mathbf{N}\}$ 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. $L = \{a^n b^n c^m d^m \mid n,m \in \mathbf{N}\}$
8. $L = \{a^n b^m c^m d^n \mid n,m \in \mathbf{N}\}$
9. $L = \{a^n b^m c^k \mid n,m,k \in \mathbf{N}, (n=m) \text{ sau } (m=k)\}$
10. $L = \{a^n b^m c^k \mid n,m,k \in \mathbf{N}, m+n = k\}$
11. $L = \{w \in \{a, b\}^* \mid$ w incepe si se termina cu acelasi simbol $\}$

6. gramatică care nu e independentă de context (tip0 nu e nici monotonă, deoarece S e și în dreapta)

$S \to ABCS$

$S \to \varepsilon$

$AB \to BA$

$BA \to AB$

$AC \to CA$

$CA \to AC$

$BC \to CB$

$CB \to BC$

$A \to a$

$B \to b$

$C \to c$

De tip 1:

$S' \to \varepsilon$

$S' \to S$

$S \to ABCS$

$S \to ABC$

$AB \to BA$

$BA \to AB$

$AC \to CA$

$CA \to AC$

$BC \to CB$

$CB \to BC$

$A \to a$

$B \to b$
$C \to c$

1. Fie limbajul: $L = \{a^n b^n c^n \mid n \text{ - natural}\}$
   Dati o gramatica de atribute care il genereaza.

2. Fie gramatica ambigua:

$S \rightarrow$ if c then S else S | if c then S | stmt

Daca inlocuim: ***if c then*** cu ***a***, ***else*** cu ***b*** si ***stmt*** cu ***i*** avem:

S -> a S

S -> a S b S

S -> i

Pentru una dintre cele 2 gramatici de mai sus:

a) Verificati daca gramatica este LL(1).

b) Incercati sa transformati gramatica in una echivalenta LL(1) aplicand factorizarea la stanga. Verificati daca gramatica obtinuta este LL(1).

c) Discutati, impreuna cu cadrul didactic, cum se poate modifica tabelul de analiza astfel incat sa se elimine conflictele.

d) Folosind analizorul LL(1), verificati daca secventa:

if c then if c then stmt else stmt

(sau echivalenta ei scrisa cu a,b,c,i)

apartine limbajului generat de gramatica.

a) Tabelul de analiză adaptat LL(1)

	a	b	i	$
S	(aSX,1)	err	(i,2)	
X		(bS,3) ~~(ε,4)~~	-	(ε,4)
a	pop			
b		pop		
i			pop	
$				acc

b) if c then if c then stmt else stmt

aaibi $\in$ L(G)?

(aaibi$,S$,ε) |–$^{\text{(push 1)}}$ (aaibi$,aSX$,1) |–$^{\text{(pop)}}$ (aibi$,SX$,1) |–$^{\text{(push 1)}}$ (aibi$,aSXX$,11) |–$^{\text{(pop)}}$ (ibi$,SXX$,11) |–$^{\text{(push 2)}}$ (ibi$,iXX$,112) |–$^{\text{(pop)}}$ (bi$,XX$,112) |–$^{\text{(push 3)}}$ (bi$,bSX$,1123) |–$^{\text{(pop)}}$ (i$,SX$,1123) |–$^{\text{(push 2)}}$ (i$,iX$,11232) |–$^{\text{(pop)}}$ ($,X$,11232) |–$^{\text{(push 4)}}$ ($,$,112324) |– acc $\Rightarrow$ aaibi $\in$ L(G) (Da), șirul producților utilizate pentru obținerea secvnței este 112324

Factorizare la dreapta

$E \rightarrow T\ X$

$X \rightarrow +\ E$

$X \rightarrow \varepsilon$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow (E)$

$F \rightarrow a$

Devine:

$E \rightarrow T\ X$

$X \rightarrow +\ E$

$X \rightarrow \varepsilon$

$T \rightarrow Z\ F$

$Z \rightarrow T *$

$Z \rightarrow \varepsilon$

$F \rightarrow (E)$

$F \rightarrow a$

3. Fie gramatica:

$E \text{ -> } T + E \mid T$

$T \text{ -> } T * F \mid F$

$F \text{ -> } (E) \mid a$

a) Verificati daca gramatica este LL(1).

b) Incercati sa transformati gramatica in una echivalenta LL(1) aplicand factorizarea la stanga. Verificati daca gramatica obtinuta este LL(1).

c) Folosind un analizor descendent verificati daca secventa:

$a + a$

apartine limbajului generat de gramatica.

$E \rightarrow T + E$

$E \rightarrow T$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow (E)$

$F \rightarrow a$

Putem factoriza la stânga dor regulile lui E

Gramatica după factorizarea la stânga:

$E \rightarrow T\ X$

$X \rightarrow + E$

$X \rightarrow \varepsilon$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow (E)$

$F \rightarrow a$

(Vom avea în continuare conflict la T)

	$First_1$	$Follow_1$
E	(,a	$,)
T	(,a	+, $,), *
X	+, ε	$,)
F	(,a	+, $,), *

Tabelul de analiză LL(1):

	+	*	(	)	a	$
E			(TX,1)		(TX,1)	
X	(+E,2)			(ε,3)		(ε,3)
T			(T*F,4) (F,5)		(T*F,4) (F,5)	
F						
+						
*						
(						
)						
a						
$						

E → T X (1)
X → + E (2)
X → ε (3) follow(X)
T → T * F (4)
T → F (5)
F → (E) (6)
F → a (7)

	$First_1$	$Follow_1$
E	(,a	$,)
T	(,a	+, $,), *
X	+, ε	$,)
F	(,a	+, $,), *

Avem conflict, deci Gramatica nu e de tip LL(1)

Dar, deoarece înmulțirea este comutativă, putem modifica regula (4) în
T → F * T

Deci, am avea gramatica:

E → T X (1)
X → + E (2)
X → ε (3)
T → F * T (4)
T → F (5)
F → (E) (6)
F → a (7)

Și să factorizăm din nou:

E → T X (1)
X → + E (2)
X → ε (3)
T → F Y (4)
Y → * T (5)
Y → ε (6)
F → (E) (7)
F → a (8)

	$First_1$	$Follow_1$
E	(,a	$,)
X	+, ε	$,)
T	(,a	+, $,)
Y	*, ε	+, $,)
F	(,a	*, +, $,)

Tabelul de analiză LL(1):

	+	*	(	)	a	$
E			(TX,1)		(TX,1)	
X	(+E,2)			(ε,3)		(ε,3)
T			(FY,4)		(FY,4)	

E → T X (1)
X → + E (2)
X → ε (3)
T → F Y (4)
Y → * T (5)
Y → ε (6)

Y	(ε,6)	(*T,5)		(ε,6)		(ε,6)
F			((E),7)		(a,8)	
+	pop					
*		pop				
(			pop			
)				pop		
a					pop	
$						acc

Nu sunt conflicte, deci gramatica este de tip LL(1)

(a+a\$,E\$,ε) $\vdash^{\text{push 1}}$ (a+a\$,TX\$,1) $\vdash^{\text{push 4}}$(a+a\$,FYX\$,14) $\vdash^{\text{push 8}}$(a+a\$,aYX\$,148) $\vdash^{\text{pop}}$ (+a\$,YX\$,148) $\vdash^{\text{push 6}}$(+a\$,X\$,1486) $\vdash^{\text{push 2}}$(+a\$,+E\$,14862) $\vdash^{\text{pop}}$(a\$,E\$,14862) $\vdash^{\text{push 1}}$ (a\$,TX\$,148621) $\vdash^{\text{push 4}}$(a\$,FYX\$,1486214) $\vdash^{\text{push 8}}$(a\$,aYX\$,14862148) $\vdash^{\text{pop}}$ (\$,YX\$,14862148) $\vdash^{\text{push 6}}$(\$,X\$,148621486) $\vdash^{\text{push 3}}$(\$,\$,1486214863) $\vdash^{\text{acc}}$ acc

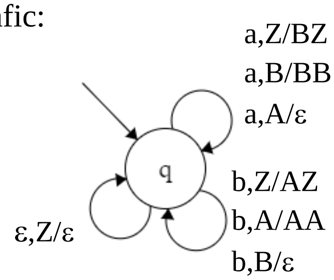
$\Rightarrow$ a+a $\in$ L(G)

f) L = {w | w ∈ {a, b}*, $nr_a(w) = nr_b(w)$ } APD cu GIC APD cu criteriul starii finale

Criteriul stivei vide

		a	b	ε
q	Z	(q,BZ)	(q,AZ)	(q, ε)
	A	(q, ε)	(q,AA)	
	B	(q,BB)	(q, ε)	

Grafic:



Verificăm dacă abba este acceptat de automat:

(starea curentă (inițială), secvența de verificat (banda de intrare, cu vf. spre stânga), stiva de lucru cu v.f. spre stânga, inițial avem simbolul din vârful stivei))

(q,abba,Z) $\vdash^{\text{rămânem în q, l-am folosit pe a, punem BZ în vf. stivei}}$ (q,bba,BZ) $\vdash$ (q,ba,Z) $\vdash$ (q,a,AZ) $\vdash$ (q,ε,Z) $\vdash$ (q,ε,ε) – s-a golit stiva (și banda de intrare), deci secvența este acceptată.

Transformați gramatica în automat push-down

S → aSb | bSa | SS | ε

Cu criteriul stivei vide, din gramatică

		a	b	ε
q	S			$(q,aSb)^{(1)}$, $(q,bSa)^{(2)}$, $(q,SS)^{(3)}$, $(q, \varepsilon)^{(4)}$
	a	$(q, \varepsilon)^{(5)}$		
	b		$(q, \varepsilon)^{(6)}$	

(q,abba,S) $\vdash^{(3)}$ (q,abba,SS) $\vdash^{(1)}$ (q,abba,aSbS) $\vdash^{(5)}$ (q,bba,SbS) $\vdash^{(4)}$ (q,bba,bS) $\vdash^{(6)}$ (q,ba,S) $\vdash^{(2)}$ (q,ba,bSa) $\vdash^{(6)}$ (q,a,Sa) $\vdash^{(4)}$ (q,a,a) $\vdash^{(5)}$ (q,ε,ε) , deci abba ∈ L(M)

Sau o altă gramatică:

S → aSbS | bSaS | ε

2. Fie gramatica:

S -> if c then S endif

S -> if c then S else S endif

S -> stmt

Daca inlocuim: ***if c then*** cu ***a***, ***else*** cu ***b*** , ***endif*** cu ***c*** , si ***stmt*** cu ***i*** avem:

S -> a S c

S -> a S b S c

S -> i

Pentru una dintre cele 2 gramatici de mai sus:

a) Verificati daca gramatica este LR(0).

b) Verificati daca este SLR.

c) Este LR(1)?

$S \to aSc$

$S \to aSbSc$

$S \to i$

Îmbogățim gramatica:

$S' \to S$

$S \to aSc$

$S \to aSbSc$

$S \to i$

a) Constucția colecției canonice LR(0):Tabela de analiz[LR(0)

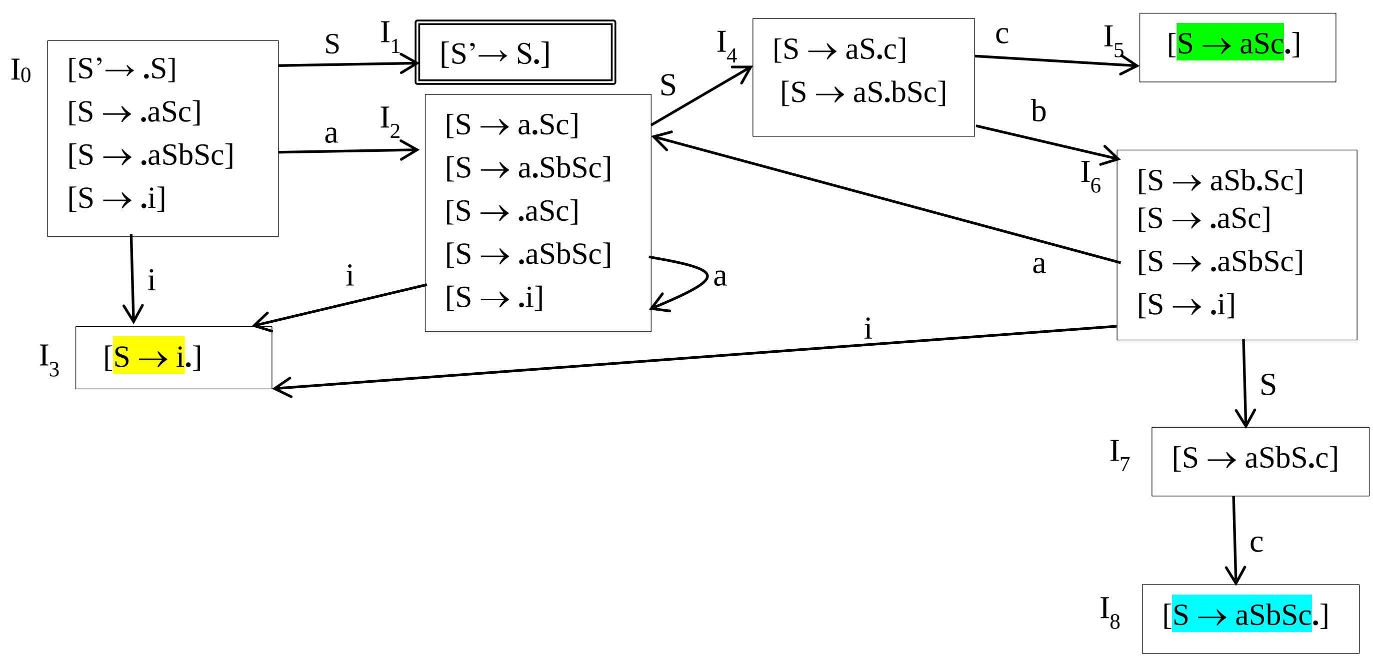


Tabela de analiză LR(0)

	Acțiune	go to				
		S	a	b	c	i
I_0	s	I_1	I_2			I_3

$S' \to S$ (0)

$S \to aSc$ (1)

$S \to aSbSc$ (2)

$S \to i$ (3)

I_1	acc					
I_2	s	I_4	I_2			I_3
I_3	3					
I_4	s			I_6	I_5	
I_5	1					
I_6	s	I_7	I_2			I_3
I_7	s				I_8	
I_8	2					

Nu avem conflicte, deci, Gramatica e de tip LR(0) $\Rightarrow$ este și SLR și LR(1)

S’ → S

S → aSc

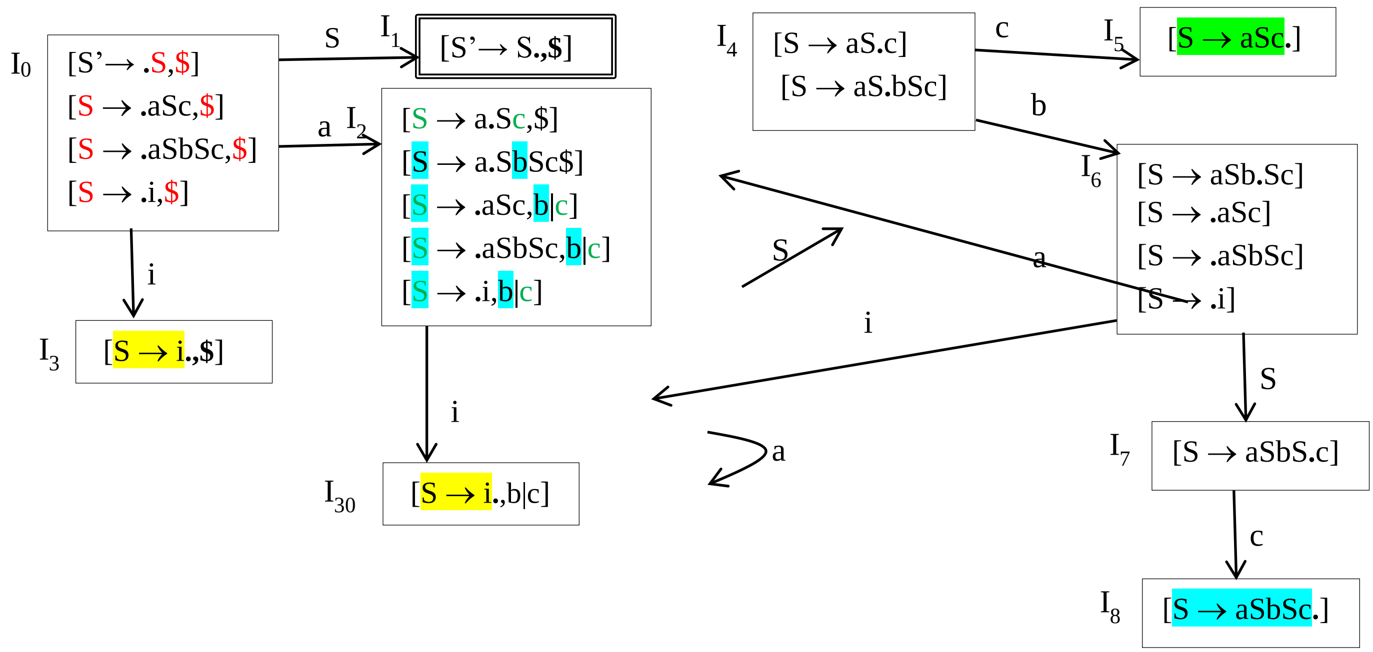
S → aSbSc

S → i

Nu enecesar, deoarece nu avem 2 neterminali unul după celălalt.

	$First_1$
S‘	a,i
S	a,i

Construcția colecției canonice LR(1) neterminată:



1. Fie gramatica:

S → AA
A → aA
A → b

b) Verificati daca gramatica este **LR(1)**. d) Folosind un analizor de tip LR(K), verificați dacă secvența ”abab” apartine limbajului generat de gramatica. Analizorul va fi ales in functie de răspunsul la intrebarile de mai sus.

b) Îmbogățim gramatica:

S’→ S (0)
S → AA (1)
A → aA (2)
A → b (3)

	$First_1$
S’	a,b
S	a,b
A	a,b

Colecția canonică LR(1):

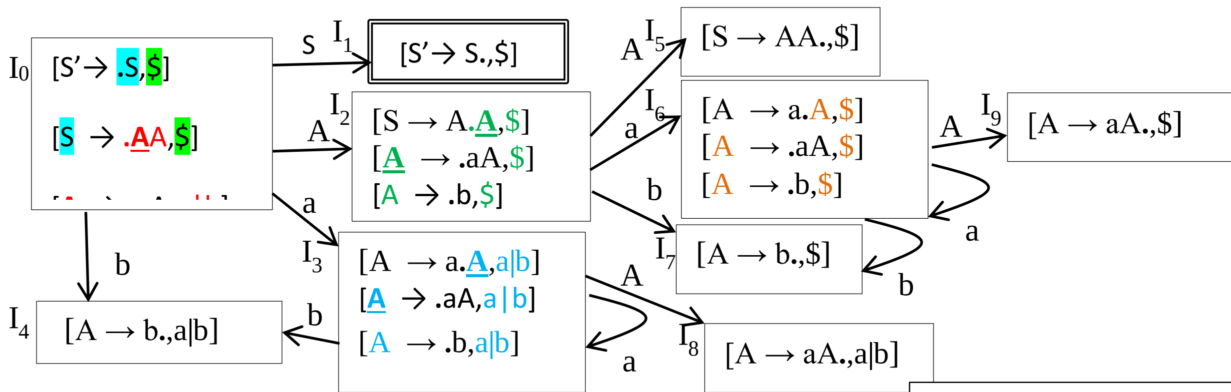


Tabela de analiză LR(1)- similar cu tabela SLR

	S	A	a	b		$
I_0	s_1	s_2	s_3	s_4		
I_1						acc
I_2		s_5	s_6	s_7		
I_3		s_8	s_3	s_4		
I_4			r_3	r_3		
I_5						r_1
I_6		s_9	s_6	s_7		
I_7						r_3
I_8			r_2	r_2		
I_9						r_2

S → AA (1)
A → aA (2)
A → b (3)

($\boxed{0,a}$bab$,ε) |– $^{(s_3)}$ ($0a3,bab$,ε) |– $^{(s_4)}$ ($0a3<u>b</u>$\boxed{4,a}b,ε) |– $^{(r3)}$ ($0<u>a3A8</u>,ab$,3) |– $^{(r_2)}$ ($0A2,ab$,23) |– $^{(s_6)}$($0A2a6,b$,23) |– $^{(s_7)}$($0A2a6b7,$,23) |– $^{(r_3)}$($0A2a6A9,$,323) |– $^{(r_2)}$($0A2A5,$,2323) |– $^{(r_1)}$($0S1,$,12323) |– $^{(acc)}$acc =>

abab∈L(G) și șirul producțiilor utilizate este 1, 2, 3, 2, 3

Nu avem conflicte, deci gramatica e de tip LR(1)

Analiza-la fel cu SLR, se utilizează predicția – vf. benzii de intrare (linie + coloană)

($0,abab$,ε) |– $^{(s_3)}$ ($0a3,bab$,ε) |– $^{(s_4)}$ ($0a3<u>b4</u>,ab$,ε) |– $^{(r3)}$ ($0<u>a3A8</u>,ab$,3) |– $^{(r_2)}$ ($0A2,ab$,23) |– $^{(s_6)}$($0A2a6,b$,23) |– $^{(s_7)}$($0A2a6b7,$,23) |– $^{(r_3)}$($0A2a6A9,$,323) |– $^{(r_2)}$($0A2A5,$,2323) |– $^{(r_1)}$($0S1,$,12323) |– $^{(acc)}$acc =>

abab∈L(G) și șirul producțiilor utilizate este 1, 2, 3, 2, 3

Cum dem. că o gramatică e independentă de context? – cu definiția (vezi cursul)
Ai doar $N \to asD$
Nu și $aN \to asD$

$S \to aSb \mid bSa \mid SS \mid \varepsilon$
Epsilon independentă:
$S' \to \varepsilon \mid S$
$S \to aSb \mid bSa \mid SS \mid ab \mid ba$

5 Gramatica de precedență simplă. Exemplu

Mai există (și vor fi studiate la seminar) și alte tipuri de analiză sintactică ascendentă. Dintre acestea, vom vedea doar cum se lucrează cu (/un exemplu de) gramatici de predecență slabă.

- Analiza ascendentă
- Depistează limita dreaptă și a a celei stângi pentru a face o reducere.

Se folosesc relațiile $<\bullet$, $\bullet>$, $=\bullet$ (relații de precedență)

Relații de precedență Wirth-Weber

$R_{<\bullet}\subset(N\cup\Sigma\cup\{\$\})\times(N\cup\Sigma\cup\{\$\})$

$R_{=\bullet}\subset(N\cup\Sigma\cup\{\$\})\times(N\cup\Sigma\cup\{\$\})$

$R_{\bullet>}\subset(N\cup\Sigma\cup\{\$\})\times(\Sigma\cup\{\$\})$

$X=\bullet Y$: $A\rightarrow\alpha XY\gamma \in P$

$X<\bullet Y$: $A\rightarrow\alpha XB\gamma \in P$, $B=>^{+}Y\gamma$

$X\bullet>a$: $A\rightarrow\alpha BY\gamma \in P$, $B=>^{+}\gamma X$, $Y=>^{*}a\delta$

$\$<\bullet X$: $S=>^{+}X\alpha$

$X\bullet>\$$: $S=>^{+}\alpha X$

Definiție:

Gramatica de precedență simplă este o gramatică independentă de context proprie

- unic invertibilă:

nu există 2 reguli de producție cu același membru drept

- între oricare 2 simboluri există cel mult o relație de precedență

Analizorul de precedență simplă:

- constuiește tabelul de precedență a operatorilor
- analizează o secvență de terminale

modelul stivei ~LR

$<\bullet$, și, $=\bullet$ - deplasare

$\bullet>$ - reducere $Y<\bullet X_1=\bullet\ldots=\bullet X_i\bullet>Z$

$A\rightarrow X_1\ldots X_i$

Exemplu:

$S \rightarrow aSSb$ (1)

$S \rightarrow c$ (2)

Cuvântul accb $\in L(G)$?

	S	a	b	c	$
S	=•	<•	=•	<•	
a	=•	<•		<•	
b		•>	•>	•>	•>
c		•>	•>	•>	•>
$		<•		<•	

$S \rightarrow aSSb$ (1)

$S \rightarrow c$ (2)

$R_{<\bullet} \subset (N \cup \Sigma \cup \{\$\}) \times (N \cup \Sigma \cup \{\$\})$

$R_{=\bullet} \subset (N \cup \Sigma \cup \{\$\}) \times (N \cup \Sigma \cup \{\$\})$

$R_{\bullet>} \subset (N \cup \Sigma \cup \{\$\}) \times (\Sigma \cup \{\$\})$

$X =\bullet Y$: $A \rightarrow \alpha XY\gamma \in P$

X și Y sunt consecutive în membrul drept al unei reguli de prod. a gramaticii

$S \rightarrow aSSb$, $S \rightarrow aSSb$, $S \rightarrow aSSb$

$X <\bullet Y$: $A \rightarrow \alpha XB\gamma \in P$, $B \Rightarrow^{+} Y\gamma$

X (a/S) se află în fața unui neterminal (S), care se derivează în cel puțin un pas în ceva ce începe cu Y (a/c)

$S \rightarrow$ **a**SSb

$S \Rightarrow^{(1)} aSSb$

$S \Rightarrow^{(2)} c$

$S \rightarrow$ a**S**Sb

$X \bullet> a$: $A \rightarrow \alpha BY\gamma \in P$, $B \Rightarrow^{+} \gamma X$, $Y \Rightarrow^{*} a\delta$

X (**b**/**c**) este ultimul din derivarea cu cel puin un pas a unui neterminal (S), și se află în fața terminalului a (b) sau a unuineterminal (S) care se derivează în ceva ce începe cu terminalul a (a/c)

$S \rightarrow$ a**SS**b

$S \Rightarrow^{(1)}$ aSS**b** Ultimul

$S \Rightarrow^{(2)}$ **c** de aici

$S \Rightarrow^{(1)} aSSb$ E mai mare decât

$S \Rightarrow^{(2)} c$ primul de aici

$S \rightarrow$ aS**S**b E mai mare decât primul de aici

$S \Rightarrow^{(1)} aSSb \Rightarrow^{(2)} acSb \Rightarrow^{(2)} accb$

$\$ <\bullet X$: $S \Rightarrow^{+} X\alpha$

	\$ e mai mic decât un X care e primul în derivarea de cel puțin un pas a lui S S =>$^{(1)}$ aSSb S =>$^{(2)}$ c X•>\$: S=>$^{+}$αX X e mai mare decât \$ dacă e ultimul în derivarea de cel puțin un pas a lui S S =>$^{(1)}$ aSSb S =>$^{(2)}$ c

	S	a	b	c	\$
S	=•	<•	=•	<•	
a	=•	<•		<•	
b		•>	•>	•>	•>
c		•>	•>	•>	•>
\$		<•		<•	

$S \rightarrow aSSb$ (1)

$S \rightarrow c$ (2)

Gramatica este de precedență simplă

(\$,accb\$,ε) |– $^{(<\bullet \text{ deplasare})}$ (\$<•a,ccb\$,ε) |– $^{(<\bullet \text{ deplasare})}$ (\$<•a<•c,cb\$,ε) |– $^{(\bullet> \text{ reducere } 2)}$ (\$<•a=•S,cb\$,2) |– $^{(<\bullet \text{ deplasare})}$ (\$<•a=•S<•c,b\$,2) |– $^{(\bullet> \text{ reducere } 2)}$ (\$<•a=•S=•S,b\$,22) |– $^{(=\bullet \text{ deplasare})}$ (\$<•a=•S=•S=•b,\$,22) |– $^{(\bullet> \text{ reducere } 1)}$ (\$S,\$,122) |– acc ⇒ **accb** ∈ L(G) și șirul producțiilor este 1, 2 și 2

$S \Rightarrow^{(1)} aS\underline{S}b \Rightarrow^{(2)} aS\underline{c}b \Rightarrow^{(2)} accb$

2.Fie L –limbajul regular corespunzator expresiei regulare:aa*b*.(2p)
Fie w = abb. Puteti identifica doua descompuneri w=xyz a.i. $xy^iz \in L$?(Justificati!)

L=$\{a^n b^m \mid n>0, m\geq 0\}$

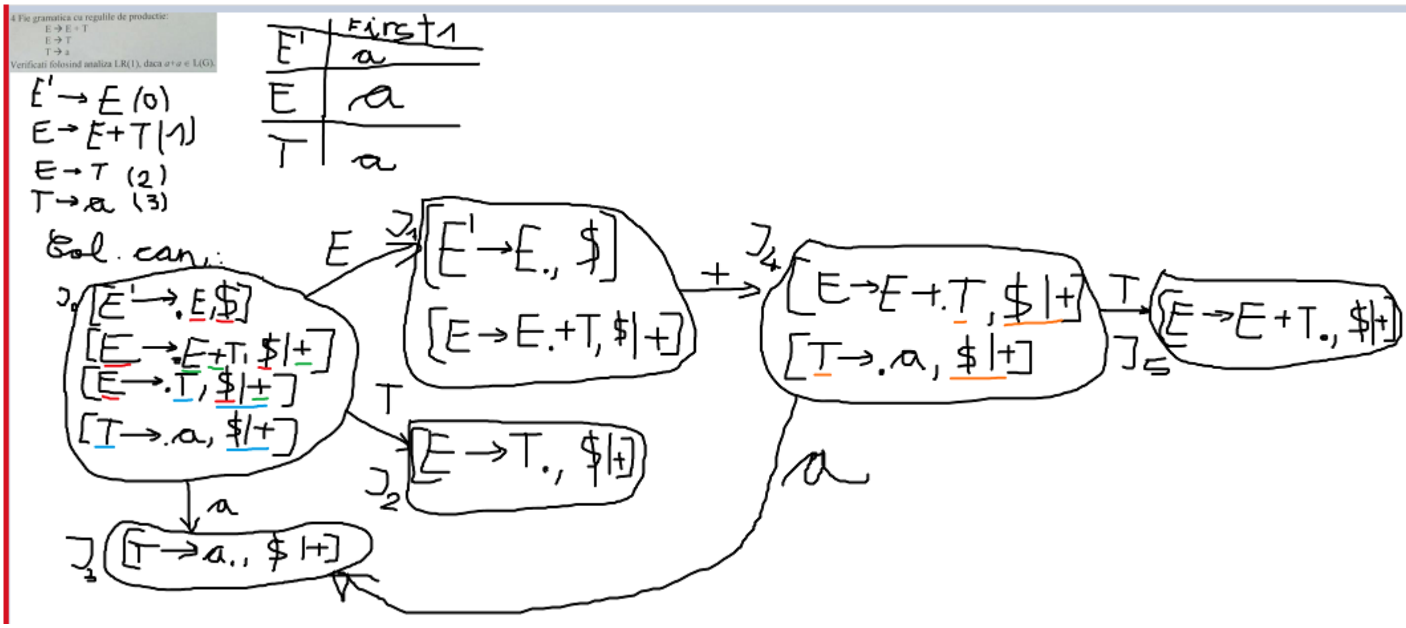
~~Descompunerea incorectă este: x=ε, y=a, z=bb, justificare:~~ $\cancel{xy^iz = a^ibb=a^ib^2}$ ~~care e de forma~~ $\cancel{a^nb^m}$ ~~cu n>0, m≥0, deci e din L~~

Descompunerea 1: x=a, y=b, z=b justificare: $xy^iz = ab^ib=ab^{i+1}$ care e de forma a^nb^m cu n>0, m≥0, deci e din L

Descompunerea 2: x=ab, y=b, z=ε justificare: $xy^iz = abb^i=ab^{i+1}$ care e de forma a^nb^m cu n>0, m≥0, deci e din L

Descompunerea 3: x=a, y=bb, z=ε justificare: $xy^iz = a(bb)^i=ab^{2i}$ care e de forma a^nb^m cu n>0, m≥0, deci e din L

LR(1)



	E	T	+	a	$
I_0	s_1	s_2		s_3	
I_1			s_4		acc
I_2			r_2		r_2
I_3			r_3		r_3
I_4		s_5		s_3	
I_5			r_1		r_1

$E \to E+T$	(1)
$E \to T$	(2)
$T \to a$	(3)

Verificăm dacă $a+a \in L(G)$:

$(\$0, a+a\$, \varepsilon) \vdash^{(\text{shift } 3)} (\$0a3, +a\$, \varepsilon) \vdash^{(\text{reducere } 3)} (\$0T2, +a\$, 3) \vdash^{(\text{reducere } 2)} (\$0E1, +a\$, 23) \vdash^{(\text{shift } 4)}$
$(\$0E1+4, a\$, 23) \vdash^{(\text{shift } 3)} (\$0E1+4a3, \$, 23) \vdash^{(\text{reducere } 3)} (\$0E1+4T5, \$, 323) \vdash^{(\text{reducere } 1)} (\$0E1, \$, 1323)$
$\vdash$acc $\Rightarrow$ $a+a \in L(G)$, șirul producțiilor utilizate pentru obținerea lui a+a este 1,3,2,3:

$E \Rightarrow^{(1)} E+\underline{T} \Rightarrow^{(3)} \underline{E}+a \Rightarrow^{(2)} \underline{T}+a \Rightarrow^{(3)} a+a$

Eliminarea redenumirilor pentru gramatica:

$E \to E+T$
~~$E \to T$~~
$T \to a$
$N_E = \{E, T\}$
$N_T = \{T\}$

După eliminare:
$E \to E+T$
$T \to a$
$E \to a$

E→T+E | T

T→F*T | F

F→E|a

Este recursivă la stg.

Eliminăm recursivitatea la stânga:

E→T+E | T

T→F*T | F

~~F→a~~

~~F→T+E | T~~

~~F→ F*T+E | F+E | F*T | F~~

F→ a | aF’

F’→ *T+E | +E | *T | *T+E F’ | +E F’ | *T F’

Eliminarea recursivității la stânga pentru:

A → BC √

~~B → AC~~

C → c √

~~B → BCC~~

B → B’

B’ → CC|CCB’

Fie urmatoarea instructiune Pascal:

if a>b then max:=a else max:=b

a) Dati o gramatica independent de context (simplificata) care descrie (cel putin) sintaxa instructiunilor din exemplul dat

b) Traduceti in cod intermediar cu 3 adrese, reprez. cvadruple

c) Fie atributul **cod** cu semnificatia: codul intermediar cu 3 adrese (reprezentare cvadruple). Dati gramatica de atribute si regulile de evaluare ale atributului **cod**.

d) Evaluati atributele pentru exemplul dat

a)

<instr_if>→ if <cond> then <asign> else <asign>

<cond> → ID > ID

<asign>→ID := ID

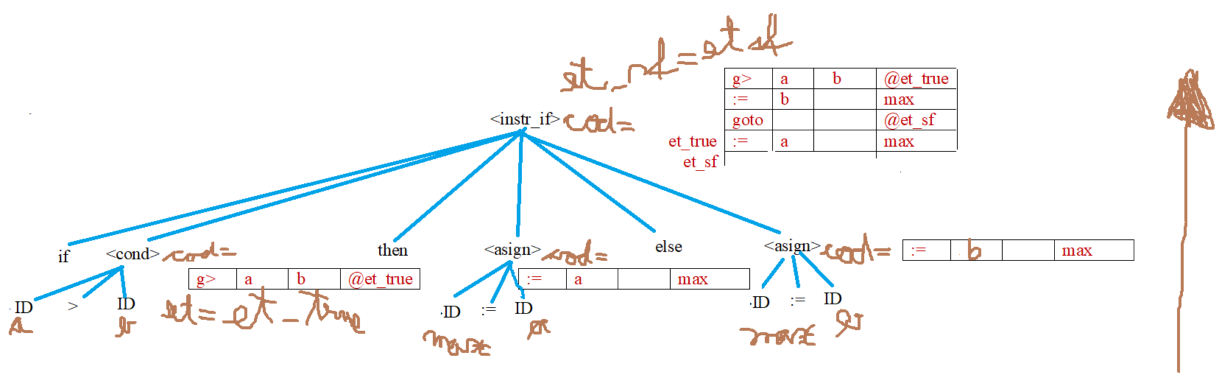
b) if a>b then max:=a else max:=b

	op.	**arg1**	**arg2**	**rez**
	...	...	...	...
	g>	a	b	@et_true
	:=	b		max
	goto			@et_sf
et_true	:=	a		max
et_sf	...	...	...	...

c)

<table>
<tr><td><instr_if>→ if <cond> then <asign> else <asign>’</td><td><instr_if>.et_sf← new Label()
<instr_if>.cod←<cond>.cod||<asign>’.cod||<table><tr><td>goto</td><td></td><td></td><td><instr_if>.et_sf</td></tr></table>||<cond>.et||<asign>.cod||<instr_if>.et_sf</td></tr>
<tr><td><cond> → ID > ID’</td><td><cond>.et←new Label()
<cond>.cod←<table><tr><td>g></td><td>ID</td><td>ID’</td><td>@<cond>.et</td></tr></table></td></tr>
<tr><td><asign>→ID := ID’</td><td><asign>.cod←<table><tr><td>:=</td><td>ID’</td><td></td><td>ID</td></tr></table></td></tr>
</table>

d)



GIC în APD cu algoritm, sse ob;ine APD cu criterial stivei vide:

L={a^nb^m| n,m>=0}

S→aS|Sb|ε

		a	b	ε
q	S			(q, aS) (q, Sb) (q, ε)
	a	(q, ε)		
	b		(q, ε)	

sau

S→AB

A→aA|ε

B→bB|ε

		a	b	ε
q	S			(q, AB)
	A			(q, aA) (q, ε)
	B			(q, bB) (q, ε)
	a	(q, ε)		
	b		(q, ε)	

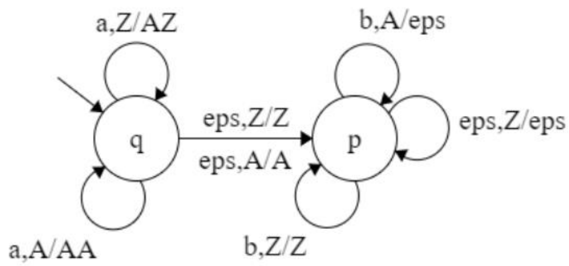
Sau...

$L=\{a^n b^m \mid m>=n>=0\}$

$S \rightarrow aSb|Sb|\varepsilon$

		a	b	ε
q	S			(q, aSb) (q, Sb) (q, ε)
	a	(q, ε)		
	b		(q, ε)	

APD – graf:



APD tabel intuitiv:

		a	b	ε
q	Z	(q,AZ)		(p,Z)
	A	(q,AA)		(p,A)
p	Z		(p, Z)	(p, ε)
	A		(p, ε)	

Varianta 2:

		a	b	ε
q	Z	(q,AZ)		(p,Z)
	A	(q,AA)	(p, ε)	
p	Z		(p, Z)	(p, ε)
	A		(p, ε)	

Varianta 3:

		a	b	ε
q	Z	(q,AZ)		(p,Z)
	A	(q,AA)	(p, ε)	
p	Z		(p, Z) (p, ε)	
	A		(p, ε)	