Context free grammars - Gramatici independente de context. Automate stiva

March 31, 2024

$\{a^n b^n\}$ e regulat?

- Pt a fi limbaj regulat, ar trebui sa existe un automat finit care sa-l recunoasca.
- ar trebui sa tina minte cati a a citit, dar n nu este limitat
- dupa ce a citit a^m ar trebui sa fie intr-o stare ce specifica o multime de simboluri nonterminale din care sa fie derivate exact b^m. $\Rightarrow$ pt fiecare m ar trebui sa fie o stare distinctica
- deci automatul ar trebui sa aiba evidenta unui numar nelimitat de posibilitati

acest lucru nu se poate face cu un numar finit de stari

- dar, nu tot ce pare a avea nevoie de memorie nelimitata, chiar are:
  - $C = \{w | w$ are un numar egal de 0 si 1$\}$
  - $D = \{w | w$ are un numar egal de aparitii 01 si 10 $\}$

$\{a^n b^n\}$ e regulat?

- Pt a fi limbaj regulat, ar trebui sa existe un automat finit care sa-l recunoasca.
- ar trebui sa tina minte cati a a citit, dar n nu este limitat
- dupa ce a citit a^m ar trebui sa fie intr-o stare ce specifica o multime de simboluri nonterminale din care sa fie derivate exact b^m. $\Rightarrow$ pt fiecare m ar trebui sa fie o stare distinctica
- deci automatul ar trebui sa aiba evidenta unui numar nelimitat de posibilitati

acest lucru nu se poate face cu un numar finit de stari

- dar, nu tot ce pare a avea nevoie de memorie nelimitata, chiar are:
  - $C = \{w | w$ are un numar egal de 0 si 1$\}$
  - $D = \{w | w$ are un numar egal de aparitii 01 si 10 $\}$

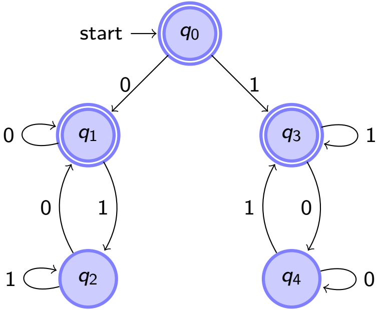
D este limbaj regulat

$D = \{w | w \text{are un numar egal de aparitii 01 si 10 ca substringuri}\}$

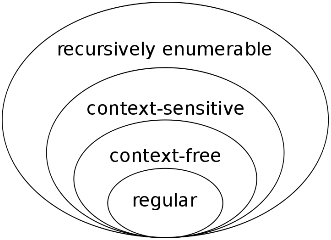
$$D = \{0, 1, \varepsilon$$
$$w \text{ daca incepe cu 0 se termina cu 0}$$
$$w \text{ daca incepe cu 1 se termina cu 1}\}$$

? $101, 1010, 0110$

$$1 + 0 + \varepsilon + 0(0+1)^*0 + 1(0+1)^*1$$



Gramatici independente de context. Context-free grammars



- $G = (T, N, Z, P)$ e independenta de context daca
- fiecare productie are forma

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

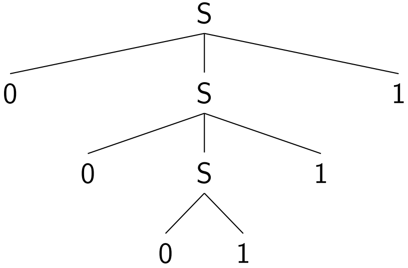
Un limbaj care e definit de o gramatica independenta de context este limbaj independent de context.
Exista CFL care nu sunt Regular languages

$0^n 1^n$

Fie $G = (\{0,1\}, \{S\}, \{S \to 01|0S1\}, S)$

- $S \Rightarrow 0S1 \Rightarrow 00S11 \Rightarrow 000111$

Arbore de parsare pentru 000111:



Limbajul parantezelor

Fie $G = (\{(,)\}, \{S\}, \{S \to SS|(S)|()\}, S)$

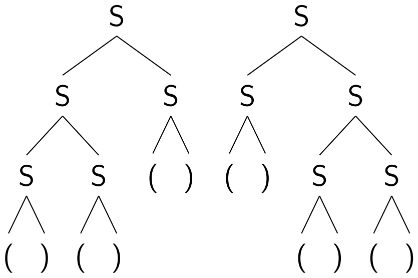
- left-most derivation: $S \Rightarrow \mathbf{S}S \Rightarrow (\mathbf{S})S \Rightarrow (())S \Rightarrow (())()$
  Deci $S \Rightarrow^L (())()$
- right-most derivation: $S \Rightarrow S\mathbf{S} \Rightarrow \mathbf{S}() \Rightarrow (S)() \Rightarrow (())()$
  Deci $S \Rightarrow^R (())()$

Gramatica ambigua - reamintire

O gramatica e ambigua daca exista un string in limbaj care e parsat in doi arbori de derivare.

Pentru $G = (\{(,)\}, \{S\}, \{S \rightarrow SS|(S)|()\}, S)$

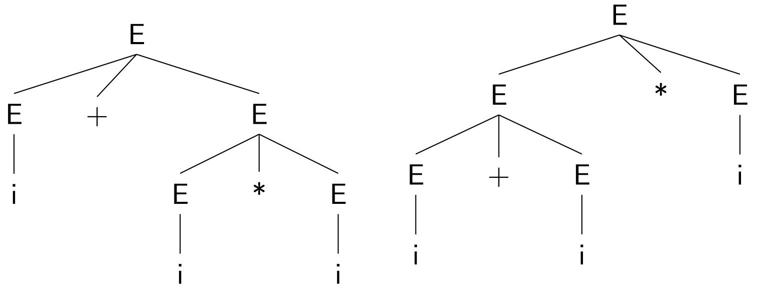
Derivarea $()()()$:



Ambiguitate

Fie $G_4 = (\{+, *, i\}, \{E\}, E, P)$ Doua derivari distincte stanga, doua derivari distincte dreapta

- $E \to E + E$
- $E \to E * E$
- $E \to i$



2+3*4??

CFG pentru Engleza

- $T = \{eats, saw, man, woman, telescope, the, with, at\}$
- $N = \{S, NP, VP, PP, DT, Vi, Vt, NN, IN\}$
- P=

S	$\rightarrow$	NP VP
VP	$\rightarrow$	Vi
VP	$\rightarrow$	Vt NP
VP	$\rightarrow$	VP PP
NP	$\rightarrow$	DT N
NP	$\rightarrow$	NP PP
PP	$\rightarrow$	IN NP

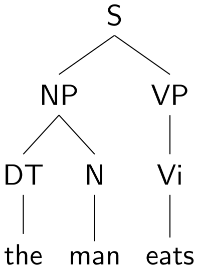
Vi	$\rightarrow$	eats
Vt	$\rightarrow$	saw
N	$\rightarrow$	man
N	$\rightarrow$	woman
N	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	at

S = sentence, VP = verb phrase, NP = noun phrase, PP = prepositional phrase, DT = determiner, Vi = intransitive verb, Vt = transitive verb, N = noun, IN = preposition

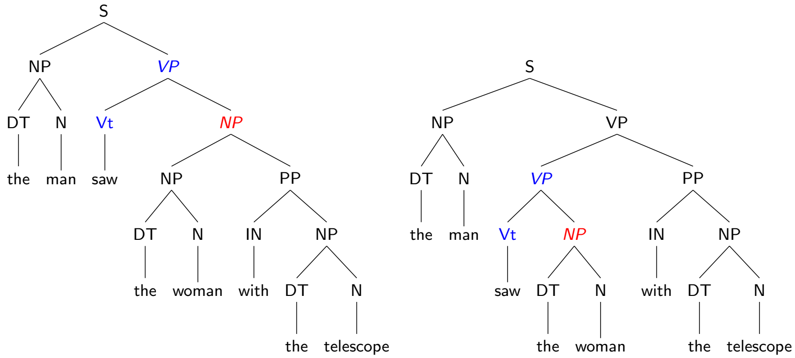
Arbore de derivare

Derivarea stanga:

$S \Rightarrow \mathbf{NP}\ VP \Rightarrow \mathbf{DT}\ N\ VP \Rightarrow the\ \mathbf{N}\ VP \Rightarrow the\ man\ \mathbf{VP}$
$\Rightarrow the\ man\ \mathbf{Vi} \Rightarrow the\ man\ eats$

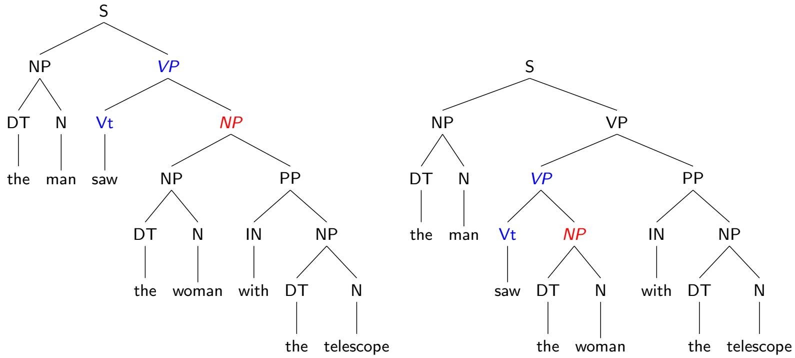


The man saw the woman with the telescope.



Ce a vazut “the man”?

The man saw the woman with the telescope.



Ce a vazut “the man”?
The telescope at the man saw the woman $\in ?\ L(G)$

if then else grammar

- $T = \{if, then, else, E1, E2, S1, S2, S3\},\ N = \{stmt, expr\}$
- P=

$$\begin{array}{lcl} stmt & \rightarrow & \text{if } expr \text{ then } stmt \\ stmt & \rightarrow & \text{if } expr \text{ then } stmt \text{ else } stmt \\ stmt & \rightarrow & \text{S1} \mid \text{S2} \mid \text{S3} \\ expr & \rightarrow & \text{E1} \mid \text{E2} \end{array}$$

if then else grammar

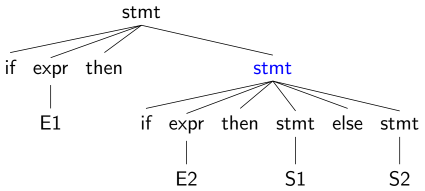
- $T = \{if, then, else, E1, E2, S1, S2, S3\},\ N = \{stmt, expr\}$
- P=

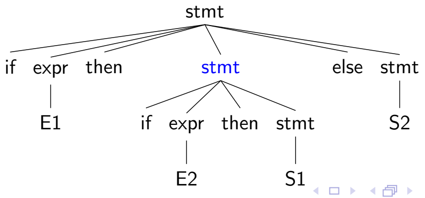
$$
\begin{array}{lcl}
stmt & \rightarrow & \text{if } expr \text{ then } stmt \\
stmt & \rightarrow & \text{if } expr \text{ then } stmt \text{ else } stmt \\
stmt & \rightarrow & \text{S1} \mid \text{S2} \mid \text{S3} \\
expr & \rightarrow & \text{E1} \mid \text{E2}
\end{array}
$$

```
if E1
then     if E2
         then S1
         else S2
```

sau?

```
if E1
then        if E2
            then S1
else S2
```





if then else - rezolvare ambiguitate

- $T = \{if, then, else, E1, E2, S1, S2, S3\}$,
  $N = \{stmt, matched\_stmt, unmatched\_stmt, expr\}$
- P=

$$
\begin{array}{lcl}
stmt & \rightarrow & m\_stmt \\
 & | & um\_stmt \\
m\_stmt & \rightarrow & \text{if } expr \text{ then } m\_stmt \text{ else } um\_stmt \\
 & | & smtm1 \\
um\_stmt & \rightarrow & \text{if } expr \text{ then } stmt \\
 & | & \text{if } expr \text{ then } m\_stmt \text{ else } um\_stmt \\
stmt1 & \rightarrow & \text{S1} \mid \text{S2} \mid \text{S3} \\
expr & \rightarrow & \text{E1} \mid \text{E2}
\end{array}
$$

Intre un *then* si un *else* e permis doar *matched_stmt*.
m_stmt=*matched_stmt* (if cu ambele *then* si *else*),
um_stmt=*unmatched_stmt*

Exemple de gramatici: Liste de *stmt*

- Recursivitate dreapta

$$\begin{array}{lcl} \mathit{stmt\_list} & \rightarrow & \mathit{stmt}\ \mathit{stmt\_list} \\ & | & \text{stmt} \\ \text{stmt} & \rightarrow & \text{A}|\text{B}|\text{C} \end{array}$$

  Arbore pt A B C?

- Recursivitate stanga

$$\begin{array}{lcl} \mathit{stmt\_list} & \rightarrow & \mathit{stmt\_list}\ \mathit{stmt} \\ & | & \text{stmt} \\ \text{stmt} & \rightarrow & \text{A}|\text{B}|\text{C} \end{array}$$

Exemple de gramatici: Liste de elemente cu separator/marcaj de final

- Separator intre elemente

  $$\begin{array}{lcl} \mathit{elem\_list} & \rightarrow & \mathit{elem\_list}\ \ \mathit{','}\ \ \mathit{elem} \\ & | & \text{elem} \\ \text{elem} & \rightarrow & \text{A}|\text{B}|\text{C} \end{array}$$

  Arbore pt A, B, C?

- Semn de punctuatie la final

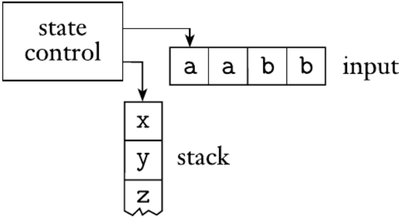
  $$\begin{array}{lcl} \mathit{stmt\_list} & \rightarrow & \mathit{stmt\_list}\ \mathit{stmt}\ \ \mathit{';'} \\ & | & \text{stmt ';'} \\ \text{stmt} & \rightarrow & \text{A}|\text{B}|\text{C} \end{array}$$

  Arbore pt A; B; C;?

Letia and Chifu. 2.3, 2.3.1 Sipser - 2.1,2.2

Automat stiva. Push down automaton (PDA)

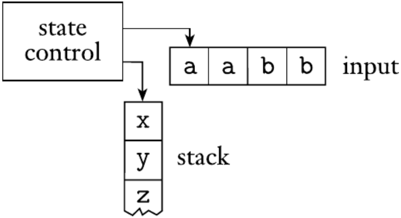
- Niciun automat finit nu poate fi construit pt a recunoaste $a^n b^n$ sau limbajul parantezelor - structuri imbricate
- Se creste puterea automatelor finite prin adaugarea unei stive drept structura aditionala de memorie



- Daca gramaticile regulate sunt o subclasa a gramaticilor independente de context, de ce se dezvolta metode specifice gramaticilor regulate si nu se aplica pt acestea cele de la gramaticile independente?

Automat stiva. Push down automaton (PDA)

- Niciun automat finit nu poate fi construit pt a recunoaste $a^n b^n$ sau limbajul parantezelor - structuri imbricate
- Se creste puterea automatelor finite prin adaugarea unei stive drept structura aditionala de memorie



- Daca gramaticile regulate sunt o subclasa a gramaticilor independente de context, de ce se dezvolta metode specifice gramaticilor regulate si nu se aplica pt acestea cele de la gramaticile independente?
- Datorita complexitatii analizei gramaticilor independente de context: gramaticile regulate sunt mai simplu de analizat

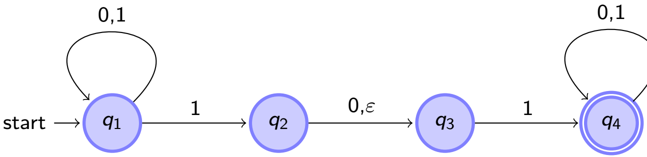
Idee: Push down automata: 00001111

1. citeste simboluri de la intrare
2. la fiecare 0 citit, impinge-l pe stiva
3. la fiecare 1 citit, scoate de pe stiva un 0
4. daca citirea stringului se termina cand stiva se goleste, accepta stringul. Daca stiva devine goala cand mai sunt 1 de citit sau s-a terminat sirul si in stiva inca mai sunt 0-uri, respinge stringul

NFA- reamintire

Un automat finit nedeterminist este $(Q, \Sigma, \delta, q_0, F)$, unde:

1. Q este setul de stari
2. Σ un alfabet finint de intrare
3. $\delta : Q \times \Sigma_\varepsilon \to P(Q)$ este o functie de tranzitie
4. $q_0 \in Q$ este starea de start
5. $F \subseteq Q$ setul de stari finale



$\delta(q_1, 1) = \{q_1, q_2\}$

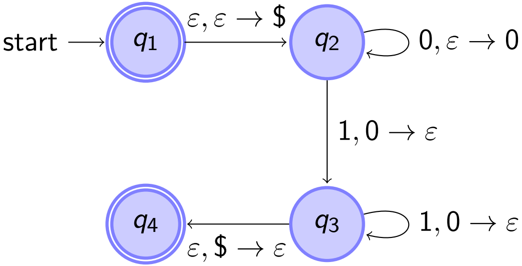
Definitie formala 1 a automatului stiva (Sipser)

Un automat stiva este $(Q, \Sigma, \Gamma, \delta, q_0, F)$, unde Q, Σ, Γ, F sunt seturi finite:

1. Q este setul de stari
2. Σ un alfabet de intrare
3. Γ este alfabetul stivei
4. $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow P(Q \times \Gamma_\varepsilon)$ este o functie de tranzitie
5. $q_0 \in Q$ este starea de start
6. $F \subseteq Q$ setul de stari finale

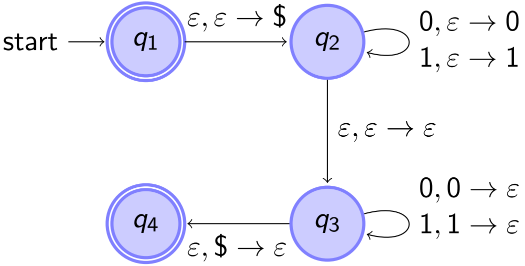
PDA for 0^n1^n

Fie $M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$



PDA for ?

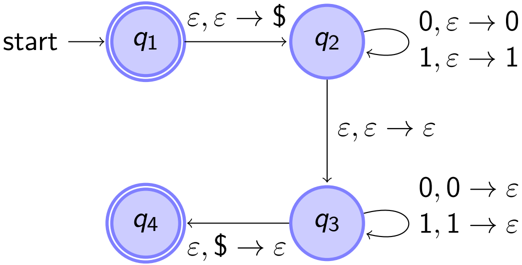
Fie $M_1 = (Q, \{0,1\}, \{\$, 0, 1\}, \delta, q_1, \{q_1, q_4\})$



PDA for $\{ww^R | w \in \{0,1\}^*\}$

w^R = w scris invers

Fie $M_1 = (Q, \{0,1\}, \{\$,0,1\}, \delta, q_1, \{q_1, q_4\})$



la fiecare pas, ghiceste daca a ajuns la mijlocul stringului sau nu

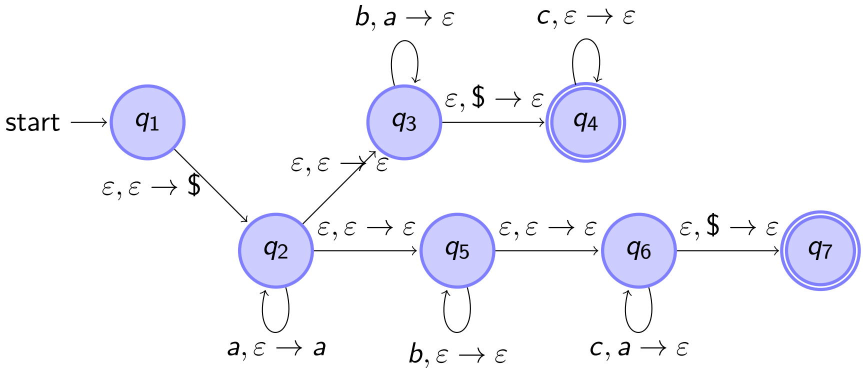
Gramatica palindrom par

- Palindrom: (T, N, P, A), $P =$

$$\{A \to 0A0|1A1$$
$$A \to \varepsilon\}$$

PDA for $\{a^i b^j c^k | i, j, k \geq 0, i = j \text{ sau } i = k\}$

Fie $M_1 = (Q, \Sigma, \Gamma, \delta, q_1, F)$



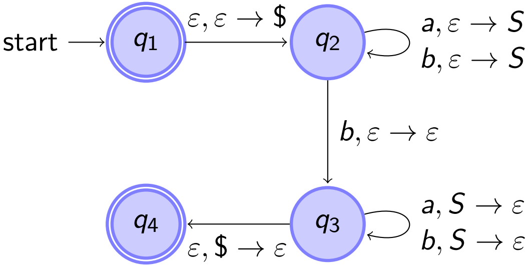
ghiceste daca e acelasi numar de a si b sau a si c

$\{vbw | v, w \in \{a, b\}^*, |v| = |w|\}$

$$M = (Q, \{a, b\}, \{\$, S\}, \delta, q_1, \{q_4\})$$

$\{vbw \mid v, w \in \{a,b\}^*, |v| = |w|\}$

$$M = (Q, \{a,b\}, \{\$, S\}, \delta, q_1, \{q_4\})$$



Exemplu Gramatica independenta de context

- Palindrom: (T, N, P, A), $P =$

$$\{A \to 0A0|1A1$$
$$A \to \varepsilon\}$$

- Acelasi numar de 0 si 1: $(\{0, 1\}, \{A\}, P, A)$, $P =$

$$\{A \to 0A1A|1A0A$$
$$A \to \varepsilon\}$$

Automat finit - reamintire

Automat finit (finite automaton, finite state acceptor):

$$A = (T, Q, R, q_0, F)$$

- Q set nevid - setul starilor interne
- $(T \cup Q, R)$ sistem de rescriere; $T \cap Q = \emptyset$
- $q_0 \in Q$ - starea initiala
- $F \subseteq Q$ - stari finale
- fiecare element din R are forma $qt \to q'$, $q, q' \in Q, t \in T$

$$L(A) = \{\tau \in T^* | q_0\tau \Rightarrow^* q, q \in F\}$$

Automat stiva - definitie sistem de rescriere

Automat stiva

$$A = (T, Q, R, q_0, F, S, s_0)$$

, unde:

- Q set nevid - setul starilor interne
- $(T \cup Q \cup S, R)$ sistem de rescriere; $T \cap Q = \emptyset$
- $q_0 \in Q$ - starea initiala
- $s_0 \in S \cup \{\varepsilon\}$ - simboluri stiva, s_0 continutul initial al stivei
- $F \subseteq Q$ - stari finale
- fiecare element din R are forma $\sigma q t \tau \to \sigma' q' \tau$,
  $\sigma, \sigma' \in S^*,\ q, q' \in Q, t \in T \cup \varepsilon, \tau \in T^*$

Daca automatul e la configuratia $s_1 ... s_n q \tau$ intr-o derivare, automatul e in starea q, τ este partea necitita din input, $s_1, ..., s_n$ este continutul pe stiva, s_n in varf.

Limbaj acceptat

Daca automatul e la configuratia $s_1 ... s_n q\tau$ intr-o derivare, automatul e in starea q, τ este partea necitita din input, $s_1, ..., s_n$ este continutul pe stiva, s_n in varf.

$$L(A) = \{\tau | s_0 q_0 \tau \# \Rightarrow^* q\#, q \in F, \tau \in T^*\}$$

0^n1^n

$M_1 = (\{0,1\}, \{q_2, q_3\}, R, q_2, \{q_3\}, \{0,1\}, \varepsilon),\ R = \{$

1. $\varepsilon q_2 0 \to 0 q_2$
2. $0 q_2 1 \to \varepsilon q_3$
3. $0 q_3 1 \to \varepsilon q_3\}$

$$??\varepsilon q_2 0011 \Rightarrow^* q_3$$

Pe stiva pot fi alte simboluri decat cele din alfabetul de intrare.

CFG - PDA

Pentru fiecare gramatica independenta de context G exista un automat stiva A a.i. L(A)=L(G).

exemplu 2

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

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

Automatul stiva construit pentru analiza descendenta:

- $T = \{+, *, (,), i\}$, $Q = \{q\}$, $q_0 = q, F = \{q\}, S = \{+, -, *, (,), i, E, T, F\}, s_0 = E$
- cu productiile R
  1. $Eq \rightarrow Tq, Eq \rightarrow T + Eq,$
  2. $Tq \rightarrow Fq, Tq \rightarrow F * Tq,$
  3. $Fq \rightarrow iq, Fq \rightarrow)E(q,$
  4. $+q+ \rightarrow q, *q* \rightarrow q,\ (q(\rightarrow q,)q) \rightarrow q, iqi \rightarrow q\}$

Derivarea gasita: i+i*i

stiva	stare	intrare	derivarea cea mai din stanga
E	q	$i+i*i$	E
T+E	q	$i+i*i$	E+T
T+T	q	$i+i*i$	T+T
T+F	q	$i+i*i$	F+T
T+i	q	$i+i*i$	i+T
T+	q	$+i*i$	
T	q	$i*i$	
F*T	q	$i*i$	i+T*F
F*F	q	$i*i$	i+F*F
F*i	q	$i*i$	i+i*F
F*	q	$*i$	
F	q	i	
i	q	i	i+i*i
	q		

Exemplu

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

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

Automatul stiva:

- $T = \{+, *, (,), i\}$, $Q = \{q\}$, $q_0 = q, F = \{q\}, S = \{+, -, *, (,), i, E, T, F\}, s_0 = E$
- cu productiile R
  1. $Tq \rightarrow Eq, E + Tq \rightarrow Eq,$
  2. $Fq \rightarrow Tq, T * Fq \rightarrow Tq,$
  3. $iq \rightarrow Fq, (E)q \rightarrow Fq,$
  4. $q+ \rightarrow +q, q* \rightarrow *q,\ q(\rightarrow (q, q) \rightarrow q), qi \rightarrow iq\}$
  5. $Eq \rightarrow q\}$

Derivarea gasita: i+i*i

stiva	stare	intrare	derivarea cea mai din dreapta
	q	$i + i * i$	i+i*i
i	q	$+i * i$	
F	q	$+i * i$	F+i*i
T	q	$+i * i$	T+i*i
E	q	$+i * i$	E+i*i
E+	q	$i * i$	
E+i	q	$*i$	
E+F	q	$*i$	E+F*i
E+T	q	$*i$	E+T*i
E+T*	q	i	
E+T*i	q	i	
E+T*F	q		
E+T	q		E+T*F
E	q		E+T
	q		E