<u>Algoritmi de decizie</u> pentru limbaje independente de context (CFG).

Teoremă: Este decidabil dacă limbajul lui $L(G)$ pentru G dat este a) vid, b) finit, c) infinit.

Dem:

Fie $G=(N,T,S,P)$ o gramatică independentă de context

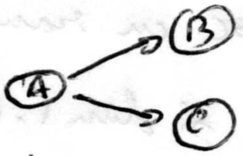
pentru a) în Forma Normală Chomsky se identifică neterminalele folositoare (accesibile din S și terminale). Dacă S este terminal atunci $L(G) \neq \emptyset$.

pentru b) și c) transformăm G în gramatica G' în forma normală Chomsky cu $L(G')=L(G)-\{\lambda\}$. $G'=(N',T,S,P')$.

Construim un graf orientat pentru G':

nodurile sunt neterminalele din G': N'

pentru $A,B \in N'$, avem muchia $(A,B) \Leftrightarrow \exists\ A \to BC$ sau $A \to CB$ în P'

exemplu: $A \to BC \in P' \Rightarrow$ 

Demonstrăm că $L(G')$ e finit (b) $\Leftrightarrow$ graful nu are cicluri.

Obs: G' în FNG $\Rightarrow$ nu avem simboluri nefolositoare

Dem graf cu cicluri $\Rightarrow L(G')$ nu e finit

Luăm un ciclu $A_0, A_1, A_2, \dots, A_n, A_0$, deci în gramatică avem

$$A_0 \Rightarrow \alpha_1 A_1 \beta_1 \Rightarrow \alpha_2 A_2 \beta_2 \Rightarrow \dots \Rightarrow \alpha_n A_n \beta_n \Rightarrow \alpha_{n+1} A_0 \beta_{n+1}$$

unde $\alpha_i, \beta_i \in N'^*$ cu propr. că $|\alpha_i \beta_i| = i$ (din forma normală Chomsky)

G' nu are simboluri nefolositoare $\Rightarrow \alpha_{n+1} \overset{*}{\Rightarrow} w_1$, $\beta_{n+1} \overset{*}{\Rightarrow} w_2$, $w_1, w_2 \in T^*$ și $\exists\, y,z \in T^*$ a.î. $S \overset{*}{\Rightarrow} y A_0 z$

și $\exists\, v \in T^*$ a.î. $A_0 \overset{*}{\Rightarrow} v$

Deci
$$\left.\begin{array}{l} S \overset{*}{\Rightarrow} y A_0 z \\ A_0 \overset{*}{\Rightarrow} w_1 A_0 w_2 \\ A_0 \Rightarrow v \end{array}\right\} \Rightarrow \forall i \geq 1$$

$$S \overset{*}{\Rightarrow} y A_0 z \overset{*}{\Rightarrow} y w_1 A_0 w_2 z \overset{*}{\Rightarrow} y w_1^2 A_0 w_2^2 z \overset{*}{\Rightarrow} y w_1^i A_0 w_2^i z \overset{*}{\Rightarrow} y w_1^i v w_2^i z \in L(G') = L(G) - \{\lambda\}.$$

Dem că dacă graful nu are cicluri $\Rightarrow L(G')$ e finit.

P.p. că graful nu are cicluri, definim $rank(A)$ = cel mai lung drum în graf care pleacă din A ($A \in N'$).

E bine definit pentru că nu avem drumuri infinite (fără cicluri).

Obs: dacă $A \to BC \in P' \Rightarrow rank(A) > rank(B)$ și
$rank(A) > rank(C)$.

Dem prin inducție că din A nu putem deriva șiruri mai lungi de 2^r unde $r = rank(A)$.

$r=0 \Rightarrow rank(A)=0 \Rightarrow outdegree(A)=0$, CNF ne spune că A e folositor deci avem producții $A \to a$, $a \in T \Rightarrow$ orice șir derivat are lungimea $1 = 2^0$.

P.p. că $r>0$ și proprietatea adevărată pentru neterminalele de rank $< r$.

Fie $A \in N'$ cu $rank(A)=r$. Fie $w \in T^*$ a.î. $A \Rightarrow^* w$.

A se rescrie în w cu anumite producții.

Dacă prima derivare este $A \to a$, $a \in T \Rightarrow |w|=1$

Dacă primul pas este $A \to BC$ cu $rank(B) < r$ și $rank(C) < r \Rightarrow$

$$w = w_1 w_2, \quad \begin{matrix} B \Rightarrow^* w_1 \\ C \Rightarrow^* w_2 \end{matrix} \quad \text{și din i.i.} \quad \left.\begin{matrix} |w_1| \le 2^{r-1} \\ |w_2| \le 2^{r-1} \end{matrix}\right| \Rightarrow$$

$$\Rightarrow |w| = |w_1| + |w_2| \le 2^{r-1} + 2^{r-1} = 2^r$$

În gramatica G luăm S (neterminalul de rang maxim).

$rank(S) = r_0 \Rightarrow \forall w \in T^*$ derivat din S ($S \Rightarrow^* w$) avem că $|w| \le 2^{r_0} \Rightarrow L(G')$ e finit. q.e.d.

Teoremă: Pt. cuvântul w și gramatica $G=(N,T,S,P)$ este decidabil dacă $w \in L(G)$.

Dem: algoritmul Cocke-Younger-Kasami săpt. viitoare.
$O(n^3)$.

Problema Corespondenței lui Post (PCP).

Se dau două liste de cuvinte

$A = x_1, x_2, \dots, x_k$

$B = y_1, y_2, \dots y_k$

$x_i, y_i \in \Sigma^* \ \forall i = \overline{1 \dots k}$

$\Rightarrow$ Spunem că avem o soluție dacă există o secvență de numere $i_1, i_2, \dots i_n$ cu $n \geq 1$ și $i_j \in \{1, 2, \dots, k\}$ pt $\forall j$ a. î.

$$x_{i_1} x_{i_2} x_{i_3} \dots x_{i_n} = y_{i_1} y_{i_2} y_{i_3} \dots y_{i_n}$$

Exemplu:

	1	2	3
$A =$	1	10111	10
$B =$	111	10	0

o soluție pentru A și B este 2, 1, 1, 3

A: <u>10111</u> <u>1</u> <u>1</u> <u>10</u>

B: <u>10</u> <u>111</u> <u>111</u> <u>0</u>

Exemplu:

A: 10, 011, 101

B 101 11 011

nu are soluție.

Teoremă: PCP nu este decidabilă.

Reformulare: nu există un algoritm care să decidă PCP are sau nu soluție (chiar și pentru cazul binar: $\Sigma = \{a, b\}$).

Folosim PCP pentru a arăta că anumite proprietăți ale CFG nu sunt decidabile.

Teoremă: Nu se poate decide algoritmic dacă o gramatică independentă de context este ambiguă

Dem:

Fie $A = x_1, x_2, \dots x_k$ și $B = y_1, y_2, \dots y_k$ o instanță a PCP.

Fie $\$, 0, 1$ trei noi simboluri ($\$, 0, 1 \notin \Sigma$).

Fie $L_A = \{ x_{i_1} x_{i_2} \dots x_{i_n} \$ 0 1^{i_1} 0 1^{i_2} \dots 0 1^{i_n} \mid 1 \leq i_1, i_2 \dots i_n \leq k \}$

$L_B = \{ y_{i_1} y_{i_2} \dots y_{i_n} \$ 0 1^{i_1} 0 1^{i_2} \dots 0 1^{i_n} \mid 1 \leq i_1, i_2 \dots i_n \leq k \}$

Fie gramatica $G^{*} = (\{S, S_A, S_B\}, \Sigma \cup \{\$, 0, 1\}, S, P)$

unde $P = \{ S \to S_A \mid S_B$

$S_A \to x_i S_A 0 1^i \mid \$ \quad \forall i \in \{1 \dots k\}$

$S_B \to y_i S_B 0 1^i \mid \$ \quad \forall i \in \{1 \dots k\}$

$\}$

Evident G este independentă de context și $L(G) = L_A \cup L_B$

Instanța $PCP(A,B)$ are soluție $\Leftrightarrow \exists i_1, i_2, \dots i_n$ a.î.

$$x_{i_1} x_{i_2} \dots x_{i_n} = y_{i_1} y_{i_2} \dots y_{i_n}$$

deci în G producem $x_{i_1} x_{i_2} \dots x_{i_n} \$ 0 1^{i_n} 0 1^{i_{n-1}} \dots 0 1^{i_1}$ pe partea S_A

$y_{i_1} y_{i_2} \dots y_{i_n} \$ 0 1^{i_n} 0 1^{i_{n-1}} \dots 0 1^{i_1}$ pe partea S_B

Deci avem un cuvânt cu două derivări stângi diferite (reguli liniare)

$$S \Rightarrow S_A \Rightarrow x_{i_1} S_A 0 1^{i_1} \Rightarrow x_{i_1} x_{i_2} S_A 0 1^{i_2} 0 1^{i_1} \Rightarrow^{*} x_{i_1} x_{i_2} \dots x_{i_n} \$ 0 1^{i_n} \dots 0 1^{i_1}$$

analog pe partea S_B, deci G este ambiguă dacă $PCP(A,B)$ are soluție

Reciproc: arătam că dacă G e ambiguă, $PCP(A,B)$ are soluție.

Deoarece cuvintele $0 1^i$ de la dreapta lui $\$$ ne spun exact ce producții s-au folosit este evident că din S_A (sau S_B) avem o derivare unică. G este ambiguă, deci ambiguitatea provine din alegerea S_A sau S_B la primul pas. Deci există cuvântul w_{AB} a.î. $S_A \Rightarrow^{*} w_{AB}$ și

$S_B \Rightarrow^{*} w_{AB}$.

Descompunem w_{AB} și aflăm $i_1, i_2, \dots i_n$ a.î. $x_{i_1} x_{i_2} \dots x_{i_n} = y_{i_1} y_{i_2} \dots y_{i_n}$

Deci am găsit o soluție pentru $PCP(A,B)$. q.e.d.

Probleme nedecidabile:

G_1 și G_2 gramatici independente de context, R expresie regulată.
Următoarele întrebări sunt nedecidabile:

a) G_1 este ambiguă?

b) $L(G_1) \cap L(G_2) \neq \emptyset$?

c) $L(G_1) \neq L(G_2)$?

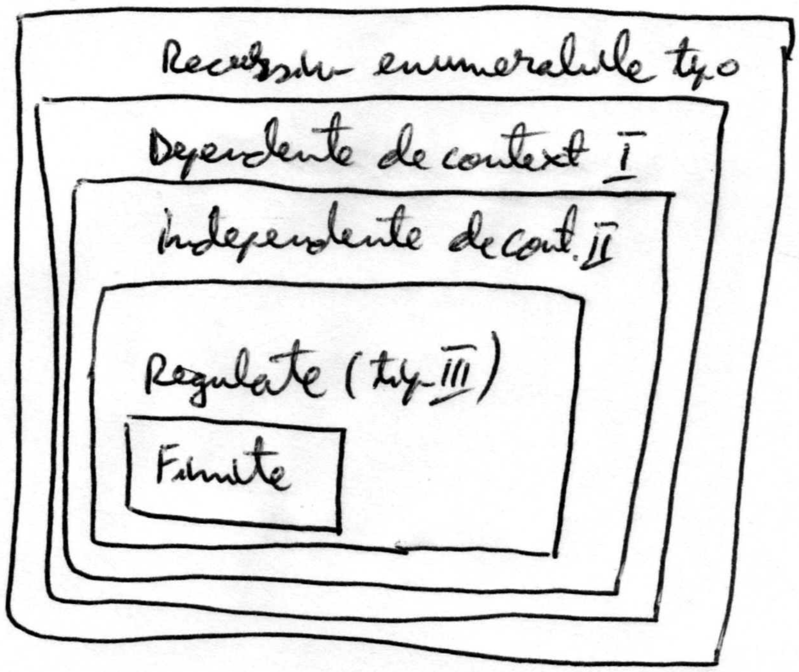
d) $L(G_1) \neq L(R)$?

e) $L(G_1) \neq T^*$?

f) $L(G_2) - L(G_1) \neq \emptyset$?

g) $L(R) - L(G_1) \neq \emptyset$?

ierarhia lui Chomsky



Limbajul	gramatica	automatul/mașina
Reg. tip III	regulată	DFA, NFA, λ-NFA, RE
CF tip II	CFG	PDA (nedeterminist)
CS tip I	CSG	TM linear bounded
RE tip 0	arbitrară	TM