Tekintsük a $G = (\{S, A\}, \{a, b\}, P, S)$ környezetfüggetlen nyelvtant, ahol

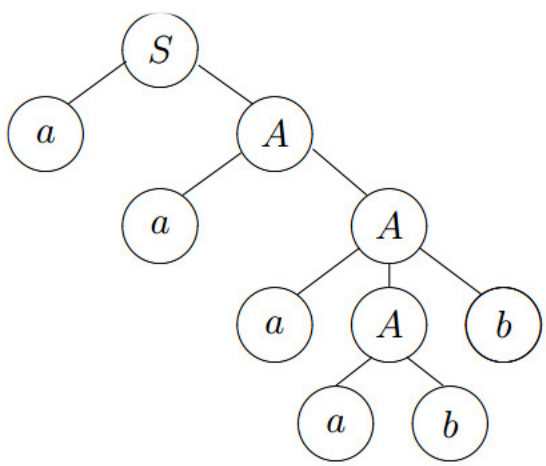
$P = \{S \to aA,\ S \to a,\ S \to \varepsilon,\ A \to aA,\ A \to aAb,\ A \to ab,\ A \to b\}$

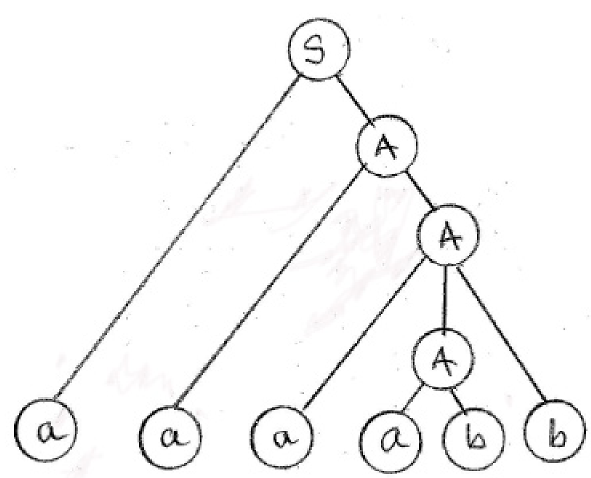
Ez a nyelvtan az $L(G) = \{a^n b^m \mid n \geq m \geq 0\}$ nyelvet generálja.

Például az $a^4b^2 \in L(G)$ szó levezetése a következő:

$$S \Longrightarrow aA \Longrightarrow aaA \Longrightarrow aaaAb \Longrightarrow aaaabb.$$

Ezt a levezetést ábrázolhatjuk levezetési vagy szintaxisfával, amelynek eredménye az $aaaabb$ szó.





Minden levezetéshez hozzárendelhető egy levezetési fa.

Veremautomaták és környezetfüggetlen nyelvek

Pumpáló lemma környezetfüggetlen nyelvekre

Környezetfüggetlen nyelvekre is létezik a reguláris nyelveknél megismert pumpáló lemmához hasonló lemma.

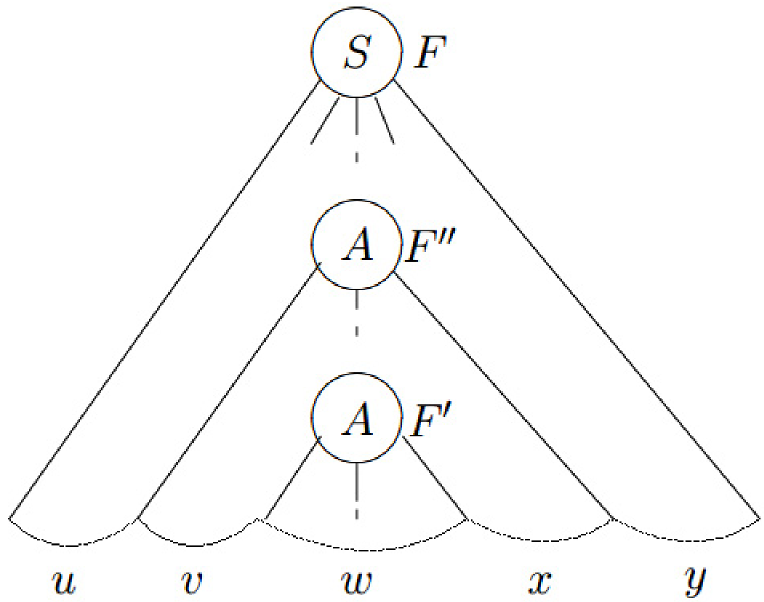
Tétel. Tetszőleges L környezetfüggetlen nyelvhez megadható egy olyan n természetes szám (amely csak az adott nyelvtől függ) úgy, hogy a nyelv minden n-nél hosszabb z szavát fel lehet írni $uvwxy$ alakban, és igazak a következők:
(1) $|w| \geq 1$,
(2) $|vx| \geq 1$,
(3) $|vwx| \leq n$,
(4) uv^iwx^iy is eleme L-nek, minden $i \geq 0$ értékre.

Legyen $G = (N, T, P, S)$ egy olyan, átnevezéseket nem tartalmazó környezetfüggetlen nyelvtan, amely L-et generálja.
$m = |N|$ a nemterminális jelek száma,
ℓ a P-beli szabályok jobb oldalai hosszának maximuma, azaz

$$\ell = \max\left\{|\alpha| \mid \exists A \in N : (A \to \alpha) \in P\right\}.$$

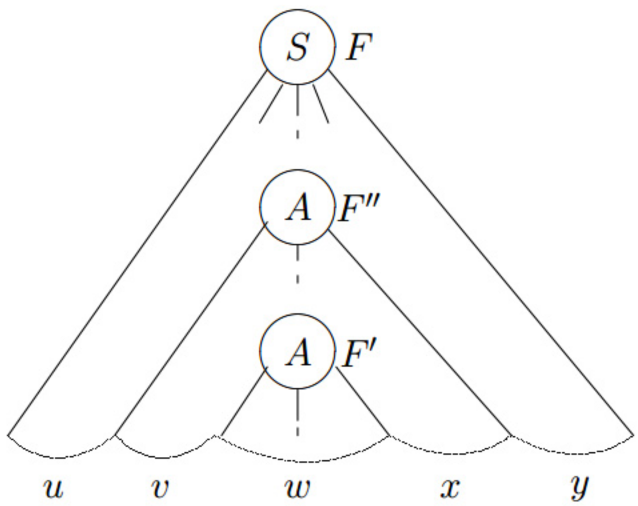
$n = \ell^{m+1}$ és $z \in L(G)$ úgy, hogy $|z| > n$.

Ekkor létezik egy olyan F levezetési fa, amelynek eredménye z.
F magassága (a gyökértől a levelekig vezető utak hosszának a maximuma): h.
Mivel F-ben minden belső csúcsnak legfeljebb ℓ leszármazottja van, ezért F-nek legfeljebb ℓ^h levele van, vagyis $|z| \leq \ell^h$.



Mivel $|z| > \ell^{m+1}$, ezért $\ell^h \geq |z| > \ell^{m+1}$, és ekkor $h > m+1$. Ebből következik, hogy az F levezetési fában van olyan út gyökértől levélig, amelyben $(m+1)$-nél több csúcs van. Tekintsünk egy ilyen utat. Mivel G nemterminálisainak száma m és ezen az úton minden, levéltől különböző csúcs nemterminálissal van címkézve, a skatulya-elv szerint van olyan nemterminális, amelyik legalább kétszer fordul elő ezen az úton.

Tekintsük azt a nemterminálist, amelyik a levéltől a gyökérig haladva legelőször ismétlődik az úton és jelöljük A-val. Jelöljük F'-vel azt részfát, amelynek gyökere A ezen előfordulása. Hasonlóképpen, jelöljük F''-vel azt a részfát, amelynek gyökere az A következő (második) előfordulása az úton. Legyen F' eredménye w. Akkor F'' eredménye vwx, míg F eredménye $uvwxy$ alakban írható (I. ábra). Megmutatjuk, hogy z felbontása kielégíti a lemmában megkövetelt (1)–(4) feltételeket.



• Mivel P-ben nincsenek ε-szabályok (kivéve esetleg az $S \to \varepsilon$ szabályt), ezért $|w| \geq 1$.

• Mivel a levezetési fa minden csúcsából és így F'' gyökeréből is legalább két él fut ki (ugyanis nincsenek átnevezések sem), ezért $|vx| \geq 1$.

• Mivel A az a nemterminális, amelyik a vizsgált úton a levéltől a gyökérig haladva legelőször megismétlődik, ezért F'' magassága legfeljebb $m+1$, amiből következik, hogy $|vwx| \leq \ell^{m+1} = n$.

Ha F-ből eltávolítjuk F'' csúcsait, csak a gyökeret hagyva meg, akkor az így kapott fa eredménye uAy, azaz $S \underset{G}{\overset{*}{\Longrightarrow}} uAy$.

Hasonlóképpen kapjuk F''-ből eltávolítva F'-t, hogy $A \underset{G}{\overset{*}{\Longrightarrow}} vAx$, és végül F' definíciója miatt, hogy $A \underset{G}{\overset{*}{\Longrightarrow}} w$.

Tehát $S \xRightarrow[G]{*} uAy$, $A \xRightarrow[G]{*} vAx$ **és** $A \xRightarrow[G]{*} w$. **Innen** $S \xRightarrow[G]{*} uAy \xRightarrow[G]{*} uwy$ **és** $S \xRightarrow[G]{*} uAy \xRightarrow[G]{*} uvAxy \xRightarrow[G]{*} \ldots \xRightarrow[G]{*} uv^iAx^iy \xRightarrow[G]{*} uv^iwx^iy$ **tetszőleges** $i \geq 1$ **értékre. Tehát tetszőleges** $i \geq 0$**-ra** $S \xRightarrow{*} uv^iwx^iy$, **vagyis tetszőleges** $i \geq 0$**-ra** $uv^iwx^iy \in L(G)$.

A pumpáló lemma következményei

1. köv. $\mathcal{L}_2 \subset \mathcal{L}_1$.

A következmény azt állítja, hogy létezik olyan környezetfüggő nyelv, amely nem környezetfüggetlen, tehát a tartalmazás szigorú. Ennek bizonyításához elég, ha találunk egy olyan környezetfüggő nyelvet, amelyre az előbbi pumpáló lemma nem teljesül. Legyen ez $L = \{a^m b^m c^m \mid m \geq 1\}$.

Ez a nyelv környezetfüggő, pl. a köv. nyelvtanok mindegyike generálja:

$G_1 = (N_1, T, P_1, S_1)$, **ahol**
$N_1 = \{S_1, X, Y\}$,
$T = \{a, b, c\}$,
$P_1 = \{S_1 \to abc,\ S_1 \to aXbc,\ Xb \to bX,\ Xc \to Ybcc,\ bY \to Yb,$
$aY \to aaX,\ aY \to aa\}$.

$G_2 = (N_2, T, P_2, S_2)$, **ahol**
$N_2 = \{S_2, A, B, C\}$,
$T = \{a, b, c\}$,
$P_2 = \{S_2 \to aS_2BC,\ S_2 \to aBC,\ CB \to BC,\ aB \to ab,\ bB \to bb,$
$bC \to bc,\ cC \to cc\}$.

Legyen n a pumpáló lemmában az L-hez tartozó természetes szám, és tekintsük a $z = a^n b^n c^n$ szót. Mivel $|z| = 3n > n$, z felírható $z = uvwxy$ alakban úgy, hogy teljesülnek a pumpáló lemmában szereplő (1)–(4) feltételek. Megmutatjuk, hogy ez ellentmondáshoz vezet.

Először megmutatjuk, hogy a v és x szavak mindegyike legfeljebb egyféle betűt tartalmaz. Valóban, ha v és x valamelyike egynél többféle betűt tartalmaz, akkor az $uvvwxxy$ szóban a betűk sorrendje nem az a, b, c sorrend, tehát $uvvwxxy \notin L(G)$, ami ellentmond a pumpáló lemmában szereplő (4) feltételnek.

Ha viszont v és x mindegyike legfeljebb egyféle betűt tartalmaz, akkor az uwy szóban valamelyik betű többször fordul elő, mint a másik kettő, tehát $uwy \notin L(G)$. Ez is ellentmond a pumpáló lemmában szereplő (4) feltételnek, tehát L nem környezetfüggetlen.

2. köv. **A környezetfüggetlen nyelvek osztálya nem zárt a metszetre.**

Legyen $N = \{S, A, B\}$, $T = \{a, b, c\}$ és

$G_1 = (N, T, P_1, S)$ ahol P_1:
$S \to AB$,
$A \to aAb \mid ab$,
$B \to cB \mid c$,

$G_2 = (N, T, P_2, S)$, ahol P_2:
$S \to AB$,
$A \to Aa \mid a$,
$B \to bBc \mid bc$.

$L(G_1) = \{a^n b^n c^m \mid n \geq 1, m \geq 1\}$, $L(G_2) = \{a^n b^m c^m \mid n \geq 1, m \geq 1\}$ nyelvek környezetfüggetlenek.

De $L(G_1) \cap L(G_2) = \{a^n b^n c^n \mid n \geq 1\}$ nem környezetfüggetlen.

Környezetfüggetlen nyelvtanok normálalakjai

Tetszőleges nyelvtanok esetében a normálalakot úgy értelmeztük, hogy a szabályok bal oldalán csak nemterminálisok szerepelnek. Környezetfüggetlen nyelvtanok esetében a normálalak a szabályok jobb oldalára adott bizonyos megkötéseket jelent. A következőkben a környezetfüggetlen nyelvtanoknak két normálalakját vizsgáljuk meg, a Chomsky-, illetve a Greibach-féle normálalakokat.

Chomsky-féle normálalak

Noam Chomsky sz. 1928. dec. 7.

Egy $G = (N, T, P, S)$ környezetfüggetlen nyelvtant Chomsky-normálalakú nevezünk, ha minden szabálya $A \to a$ vagy $A \to BC$ alakú, ahol $A, B, C \in N$, $a \in T$.

Példa. A $G = (\{S, A, B, C\}, \{a, b\}, \{S \to AB,\ S \to CB,\ C \to AS,\ A \to a,\ B \to b\}, S)$ nyelvtan Chomsky-normálalakú és $L(G) = \{a^n b^n \mid n \geq 1\}$.

Minden ε-mentes környezetfüggetlen nyelvtanhoz megadható egy vele ekvivalens Chomsky-normálalakú nyelvtan. Megadunk egy algoritmust, amely a $G = (N, T, P, S)$ ε-mentes környezetfüggetlen nyelvtant átalakítja a $G' = (N', T, P', S)$ Chomsky-normálalakúvá.

CHOMSKY-ALAK(G)

1 $N' \leftarrow N$

2 küszöböljük ki a szabályokban az átnevezéseket, és legyen P' az új szabályhalmaz (lásd az ÁTNEVEZÉS-KIZÁRÁS algoritmust)

3 a P' minden olyan szabályában, amelynek jobb oldala legalább két szimbólumból áll, minden a terminális jelet helyettesítsünk egy új A nemterminálissal, amelyet vegyünk fel N'-be, és vegyük fel a szabályok közé az $A \to a$ új szabályt

4 minden $B \to A_1A_2 \dots A_k$ alakú szabályt, ahol $k \geq 3$ és $A_1, A_2, \dots, A_k \in N$, helyettesítsünk a következőkkel:

$$\begin{array}{ll} B & \to A_1C_1, \\ C_1 & \to A_2C_2, \\ \dots & \\ C_{k-3} & \to A_{k-2}C_{k-2}, \\ C_{k-2} & \to A_{k-1}A_k, \end{array}$$

ahol $C_1, C_2, \dots, C_{k-2}$ új nemterminális szimbólumok, amelyeket felveszünk N'-be.

5 **return** G'

Példa. Legyen $G = (\{S, D\}, \{a, b, c\}, \{S \to aSc,\ S \to D,\ D \to bD,\ D \to b\}, S)$. Könnyű belátni, hogy $L(G) = \{a^nb^mc^n \mid n \geq 0, m \geq 1\}$. A Chomsky-féle normálalakú nyelvtanná való alakítás lépései a következők:

1. lépés: $N' = \{S, D\}$

2. lépés: Az $S \to D$ átnevezés kiküszöbölése után, a szabályok a következők:

$S \to aSc \mid bD \mid b$
$D \to bD \mid b$

3. lépés: Mivel a szabályokban három terminális szerepel, három új nemterminálist vezetünk be (legyenek ezek A, B, C). Ekkor a szabályok:

$S \to ASC \mid BD \mid b$
$D \to BD \mid b$
$A \to a$
$B \to b$
$C \to c$

4. lépés: Mivel egyetlen szabály kivételével, amelynek jobb oldala három szimbólumból áll, minden más szabály jobb oldala legfeljebb kettő hosszúságú, egyetlen új nemterminálist kell bevezetnünk, legyen ez E. Ezért

$N' = \{S, A, B, C, D, E\}$, **a** P' **szabályai pedig:**
$S \to AE \mid BD \mid b$
$D \to BD \mid b$
$A \to a$
$B \to b$
$C \to c$
$E \to SC$

Greibach-féle normálalak

Sheila Greibach sz. 1939. okt. 6.

Egy $G = (N, T, P, S)$ környezetfüggetlen nyelvtant Greibach-normálalakúnak nevezünk, ha minden szabálya $A \to aw$ alakú, ahol $A \in N$, $a \in T$, $w \in N^*$.

Példa. A $G = (\{S, B\}, \{a, b\}, \{S \to aB,\ S \to aSB,\ B \to b\}, S)$ nyelvtan Greibach-normálalakú és $L(G) = \{a^n b^n \mid n \geq 1\}$.

Minden ε-mentes környezetfüggetlen nyelvtanhoz megadható egy vele ekvivalens Greibach-normálalakú nyelvtan. Megadunk egy algoritmust, amely a Chomsky-normálalakban levő $G = (N, T, P, S)$ környezetfüggetlen nyelvtant átalakítja a $G' = (N', T, P', S)$ Greibach-normálalakúvá.

Először rögzítjük a nemterminális jelek egy $A_1, A_2, \ldots, A_n$ sorrendjét úgy, hogy A_1 legyen a kezdőszimbólum. Az algoritmusban a következő jelöléseket használjuk: $x \in N'^+$, $\alpha \in TN'^* \cup N'^+$.

```
Greibach-alak(G)
N' ← N
P' ← P
for i ← 2 to n                                        ▷A_i → A_j x, j < i esete.
    do for j ← 1 to i − 1
          do minden A_i → A_j x és minden A_j → α alakú szabályra
             (ahol α nem kezdődik A_j-vel)
               vegyük fel P'-be az A_i → αx szabályt,
             töröljük P'-ből az A_i → A_j x szabályokat
    if létezik A_i → A_i x alakú szabály              ▷A_i → A_i x esete.
      then vegyük fel N'-be az új B_i nemterminálist,
           minden A_i → A_i x alakú szabályra vegyük fel P'-be a
             B_i → xB_i és B_i → x szabályokat,
             töröljük P'-ből az A_i → A_i x szabályt,
           minden A_i → α alakú szabályra (ahol α nem kezdődik A_i-vel)
             vegyük fel P'-be az A_i → αB_i szabályt
for i ← n − 1 downto 1                                ▷A_i → A_j x, j > i esete.
    do for j ← i + 1 to n
        do minden A_i → A_j x és minden A_j → α alakú szabályra
                vegyük fel P'-be az A_i → αx szabályt és
                töröljük P'-ből az A_i → A_j x szabályokat,
for i ← 1 to n                                        ▷B_i → A_j x esete.
    do for j ← 1 to n
          do minden B_i → A_j x és minden A_j → α alakú szabályra
               vegyük fel P'-be a B_i → αx szabályt és
               töröljük P'-ből a B_i → A_j x szabályokat
return G'
```

Az algoritmus az első lépésben az $A_i \to A_j x,\ j < i$ alakú szabályokat átalakítja úgy, hogy azok $A_i \to A_j x,\ j \geq i$ vagy $A_i \to \alpha$

alakúak legyenek, ahol ez utóbbi már Greibach-normálalakú. A második lépésben, új nemterminális bevezetésével, kiküszöböli az $A_i \to A_i x$ alakú szabályokat, majd helyettesítésekkel eléri, hogy az $A_i \to A_j x,\ j > i$ és $B_i \to A_j x$ alakú szabályok is Greibach-normálalakúak legyenek.

Példa. Alakítsuk át a következő Chomsky-normálalakú szabályokat Greibach-normálalakúvá:

$A_1 \to A_2A_3 \mid A_2A_4$
$A_2 \to A_2A_3 \mid a$
$A_3 \to A_2A_4 \mid b$
$A_4 \to c$

Az algoritmus lépései:

3–5: Az $A_3 \to A_2A_4$ szabályt kell átalakítani. Erre csak az $A_2 \to a$ szabály alkalmas. Ezért felvesszük a szabályok közé az $A_3 \to aA_4$ szabályt és töröljük az $A_3 \to A_2A_4$ szabályt.
Tehát a szabályok:

$A_1 \to A_2A_3 \mid A_2A_4$
$A_2 \to A_2A_3 \mid a$
$A_3 \to aA_4 \mid b$
$A_4 \to c$

6-7: Az $A_2 \to A_2A_3$ kiküszöbölése a következő szabályokkal történik:

$B_2 \to A_3B_2$
$B_2 \to A_3$
$A_2 \to aB_2$

Tehát, a 6–7. lépések után, a szabályok:

$A_1 \to A_2A_3 \mid A_2A_4$
$A_2 \to aB_2 \mid a$
$A_3 \to aA_4 \mid b$
$A_4 \to c$
$B_2 \to A_3B_2 \mid A_3$

8–10: Az A_1 baloldalú szabályoknál végzünk helyettesítéseket. Az eredmény:

$A_1 \to aA_3 \mid aB_2A_3 \mid aA_4 \mid aB_2A_4$
11–13: Hasonlóképpen járunk el a B_2 baloldalú szabályokkal:

$B_2 \to aA_4B_2 \mid aA_3A_4B_2 \mid aA_4 \mid aA_3A_4$

Miután kitöröltük a 8–13. lépésekben a helyettesített szabályokat, a következőket kapjuk, amelyek már mind Greibach-alakú szabályok:

$A_1 \to aA_3 \mid aB_2A_3 \mid aA_4 \mid aB_2A_4$
$A_2 \to aB_2 \mid a$
$A_3 \to aA_4 \mid b$
$A_4 \to c$
$B_2 \to aA_4B_2 \mid aA_3A_4B_2 \mid aA_4 \mid aA_3A_4$

Példa. Nézzünk meg egy másik példát. Megadunk egy nyelvtant a következő nyelv generálására.

$$L = \left\{a^nb^kc^{n+k} \mid n \geq 0, k \geq 0, n+k > 0\right\}.$$

Bebizonyítható, hogy a következő nyelvtan generálja L-et.

$$G = \left\{\{S,R\}, \{a,b,c\}, \{S \to aSc,\ S \to ac,\ S \to R, R \to bRc,\ R \to bc\}, S\right\}$$

Először kiküszöböljük az átnevezéseket (itt most csupán egy van), azután megadunk egy vele ekvivalens Chomsky-alakú nyelvtant, majd egy ezzel ekvivalens Greibach-alakút. Az $S \to R$ átnevezés kiküszöbölése után a következő szabályokat kapjuk:

$S \to aSc \mid ac \mid bRc \mid bc$
$R \to bRc \mid bc$.

Bevezetjük az $A \to a, B \to b, C \to c$ szabályokat, majd a terminálisokat helyettesítjük minden szabály jobb oldalán a megfelelő változóval:

$S \to ASC \mid AC \mid BRC \mid BC,$
$R \to BRC \mid BC,$
$A \to a,\ B \to b,\ C \to c.$

Két új változó (D, E) bevezetése után:

$S \to AD \mid AC \mid BE \mid BC,$
$D \to SC,$
$E \to RC,$
$R \to BE \mid BC,$
$A \to a,\ B \to b,\ C \to c.$

Ez már Chomsky-féle normálalak. Induljunk ki ebből a nyelvtanból, miután átírjuk a változókat A_i alakúra, hogy könnyebben alkalmazhassuk az algoritmust. Tehát, a következő átnevezés után

S helyett A_1, A helyett A_2, B helyett A_3, C helyett A_4,
D helyett A_5, E helyett A_6, R helyett A_7,

nyelvtanunk a következő szabályokat tartalmazza:

$A_1 \to A_2A_5 \mid A_2A_4 \mid A_3A_6 \mid A_3A_4,$
$A_2 \to a,\ A_3 \to b,\ A_4 \to c,$
$A_5 \to A_1A_4,$
$A_6 \to A_7A_4,$
$A_7 \to A_3A_6 \mid A_3A_4.$

Az algoritmus 3–5. lépéseinek alkalmazásakor a következő új szabályok jelennek meg:

$A_5 \to A_2A_5A_4 \mid A_2A_4A_4 \mid A_3A_6A_4 \mid A_3A_4A_4$ **majd**
$A_5 \to aA_5A_4 \mid aA_4A_4 \mid bA_6A_4 \mid bA_4A_4$
$A_7 \to A_3A_6 \mid A_3A_4,$ **majd**
$A_7 \to bA_6 \mid bA_4.$

Tehát:

$A_1 \to A_2A_5 \mid A_2A_4 \mid A_3A_6 \mid A_3A_4,$
$A_2 \to a,\ A_3 \to b,\ A_4 \to c,$
$A_5 \to aA_5A_4 \mid aA_4A_4 \mid bA_6A_4 \mid bA_4A_4$
$A_6 \to A_7A_4,$
$A_7 \to bA_6 \mid bA_4.$

A 6–7. lépéseket átugorjuk, hisz nincs balrekurzív szabály. A 8–10. lépésekben a megfelelő helyettesítések után:

$A_1 \to aA_5 \mid aA_4 \mid bA_6 \mid bA_4,$
$A_2 \to a,$
$A_3 \to b,$
$A_4 \to c,$
$A_5 \to aA_5A_4 \mid aA_4A_4 \mid bA_6A_4 \mid bA_4A_4$
$A_6 \to bA_6A_4 \mid bA_4A_4,$
$A_7 \to bA_6 \mid bA_4.$

Sajátos nyelvtanok és nyelvek

- $LL(k)$,
- $LR(k)$,
- elsőbbségi (precedencia) nyelvtanok

$LL(k)$ nyelvtanok

Ahhoz, hogy bebizonyítsuk, hogy egy szó benne van egy környezetfüggetlen nyelvben, meg kell adnunk az adott szónak egy levezetését (derivációját). Ez általában nem könnyü feladat, hisz egy-egy alkalommal több helyettesítés közül is választhatunk, és nem könnyű eldönteni, hogy melyiket válasszuk. Ha ez a választás mindig megtörténhet egyértelműen úgy, hogy az elemzendő szóban a még meg nem vizsgált rész k ($k \geq 1$) betűjét előre megnézzük, akkor azt mondjuk, hogy az illető nyelvtan $LL(k)$ típusú, az elemzés pedig $LL(k)$ elemzés.

Az $LL(0)$ eset nem érdekes, mert ez a nyelvtan csupán egyetlen szót generál.

Az $LL(k)$ rövidítés a következő szövegből származik: Left to right scan, producing Leftmost derivation with k symbol lookahead.

Mielőtt még definiálnánk az $LL(k)$ nyelvtant, vezessük be a következő jelölést egy adott $G = (N, T, P, S)$ nyelvtanra és egy $\alpha \in (N \cup T)^*$ szóra.

$$FIRST_k^G(\alpha) = \{x \mid \alpha \underset{G}{\overset{*}{\Longrightarrow}} xy,\ xy \in T^*,\ |x| = k\} \cup \{x \mid \alpha \underset{G}{\overset{*}{\Longrightarrow}} x,\ x \in T^*,\ |x| < k\}.$$

azaz, $FIRST_k^G(\alpha)$ az α-ból G-ben levezethető terminális szavak k hosszúságú kezdőszeleteinek a halmazát jelenti. Ha a terminális szó rövidebb k-nál, akkor azonos a k hosszúságú kezdőszeletével.

Ha $\alpha \in T^*$, akkor $\alpha = xy$ vagy $\alpha = x$, tehát $FIRST_k^G(\alpha)$ az α terminális szó k hosszúságú kezdőszeletét tartalmazza csupán.

A G felső indexet esetleg elhagyhatjuk.

Példa.

$G = (\{S\}, \{a, b\}, \{S \to ab,\ S \to aSb\}, S)$

$$FIRST_1(aSb) = \{a\}$$
$$FIRST_2(aSb) = \{aa\}$$
$$FIRST_3(aSb) = \{aaa, aab\}$$
$$FIRST_4(aSb) = \{aabb, aaab, aaaa\}$$
$$FIRST_5(aSb) = \{aabb, aaabb, aaaab, aaaaa\}$$

A $G = (N, T, P, S)$ környezetfüggetlen nyelvtant $LL(k)$ nyelvtannak ($k \geq 0$) nevezzük, ha bármely

$$S \overset{*}{\underset{G}{\Longrightarrow}} uA\alpha \underset{G}{\Longrightarrow} u\beta_1\alpha \overset{*}{\underset{G}{\Longrightarrow}} ux$$

$$S \overset{*}{\underset{G}{\Longrightarrow}} uA\alpha \underset{G}{\Longrightarrow} u\beta_2\alpha \overset{*}{\underset{G}{\Longrightarrow}} uy$$

legbaloldalibb levezetéspárra (ahol $\alpha, \beta_1, \beta_2 \in (N \cup T)^*$, $u, x, y \in T^*$) a $FIRST_k(x) = FIRST_k(y)$ egyenlőségből következik, hogy $\beta_1 = \beta_2$.

Példa. **Vizsgáljuk meg a** $G = (\{S\}, \{a, b\}, \{S \to ab,\ \ S \to aSb\}, S)$ **nyelvtant!**

A következő levezetéspárból következik, hogy G **nem** $LL(1)$ **nyelvtan.**

$$S \Longrightarrow aSb \stackrel{*}{\Longrightarrow} \underbrace{aa}_{u}\underbrace{S}_{A}\underbrace{bb}_{\alpha} \Longrightarrow \underbrace{aa}_{u}\underbrace{ab}_{\beta_1}\underbrace{bb}_{\alpha} \stackrel{*}{\Longrightarrow} \underbrace{aa}_{u}\underbrace{abbb}_{x}$$

$$S \Longrightarrow aSb \stackrel{*}{\Longrightarrow} \underbrace{aa}_{u}\underbrace{S}_{A}\underbrace{bb}_{\alpha} \Longrightarrow \underbrace{aa}_{u}\underbrace{aSb}_{\beta_2}\underbrace{bb}_{\alpha} \stackrel{*}{\Longrightarrow} \underbrace{aa}_{u}\underbrace{aabbbb}_{y}$$

És annak ellenére, hogy $FIRST_1(x) = FIRST_1(y) = \{a\}$**, nem következik a** $\beta_1 = \beta_2$ **egyenlőség, mivel** $\beta_1 = ab,\ \beta_2 = aSb$**.**

Tétel. **Egy** $G = (N, T, P, S)$ **környezetfüggetlen nyelvtan akkor és csakis akkor** $LL(k)$ **típusú (**$k \geq 1$**), ha bármely**
$S \stackrel{*}{\Longrightarrow} uA\gamma$ **legbaloldalibb levezetésre** $\big($**ahol** $u \in T^*$, $\gamma \in (N \cup T)^*\big)$ **és**
$A \to \alpha$**,** $A \to \beta$ **két különböző szabályra**

$$FIRST_k(\alpha\gamma)\ \cap\ FIRST_k(\beta\gamma) = \emptyset.$$

Példa. **Előbbi példánk,** $G = (\{S\}, \{a, b\}, \{S \to ab,\ \ S \to aSb\}, S)$ **esetében a következő levezetések lehetségesek:**
$S \stackrel{*}{\Longrightarrow} a^nSb^n$**,** $n \geq 1$ **és**
$S \stackrel{*}{\Longrightarrow} ab$
valamint léteznek az $S \to aSb$ **és** $S \to ab$ **szabályok.**

Ekkor
$FIRST_1(aSbb^n) = \{a\}$**,** $FIRST_1(abb^n) = \{a\}$**; metszetük nem üres.**
Tehát G **nem** $LL(1)$ **típusú.**

$FIRST_2(aSbb^n) = \{aa\}$, $FIRST_2(abb^n) = \{ab\}$; metszetük üres. Tehát a G nyelvtan $LL(2)$ típusú.

Könnyű belátni, éppen a fenti tétel alapján, hogy ha egy nyelvtan $LL(k)$, akkor egyben $LL(k+1)$ is. Például az előbbi nyelvtan esetében, amely $LL(2)$ típusú:

$FIRST_3(aSbb^n) = \{aaa, aab\}$, $FIRST_3(abb^n) = \{aba\}$, metszetük üres. Tehát a G nyelvtan $LL(3)$ típusú is.

Következmény. Ha egy környezetfüggetlen nyelv $LL(k)$ típusú, akkor $LL(k+1)$ típusú is.

Fontos eset az $LL(1)$. Ekkor annak vizsgálata, hogy egy nyelvtan $LL(1)$ típusú nagyon egyszerű, mivel nyelvtanaink ε-mentesek. (Nincs $A \to \varepsilon$ szabály, kivéve ha $S \to \varepsilon$, de ekkor S szabály jobb oldalán nem szerepelhet.)

Tétel. Egy környezetfüggetlen nyelvtan akkor és csakis akkor $LL(1)$ típusú, ha minden $A \to \alpha$, $A \to \beta$ különböző szabálypárra
$FIRST_1(\alpha) \cap FIRST_1(\beta) = \emptyset$.

Példa. 1) A $G_1 = \{S, A, B\}, \{a, b\}, P, S)$, ahol

$$P: \quad \begin{aligned} &S \to aAB \mid bBA \\ &A \to a \mid bS \\ &B \to b \mid aS \end{aligned}$$

$LL(1)$ típusú nyelvtan, mivel:
$FIRST_1(aAB) = \{a\}$, $FIRST_1(bBA) = \{b\}$
$FIRST_1(a) = \{a\}$, $FIRST_1(bS) = \{b\}$
$FIRST_1(b) = \{b\}$, $FIRST_1(aS) = \{a\}$ és a megfelelő metszetek üresek.

2) a $G_2 = (\{S, A\}, \{0, 1\}, \{S \to 0A \mid 0,\ A \to 0A \mid 0A1 \mid 01 \mid 1\}, S)$ nyelvtan nem $LL(1)$, mivel például az $A \to 0A,\ A \to 0A1$ szabályokra:

$FIRST_1(0A) = \{0\},\ \ FIRST_1(0A1) = \{0\}$, tehát metszetük nem üres.

A következő tétel fontos az elemzés szempontjából, és az értelmezésből következik.

Tétel. Minden $LL(k)$ nyelvtan egyértelmű.

Egy A változót (nemterminális jelet) **balrekurzívnak** nevezünk, ha létezik

$A \overset{*}{\Longrightarrow} A\alpha \ \left(\alpha \in (N \cup T)^*\right)$

levezetés.

Ha egy környezetfüggetlen nyelvtanban van olyan változó, amely nem szerepel egyetlen terminális szó levezetésében sem, akkor az a nyelvtan **redundáns**.

Tétel. Ha egy G nem redundáns környezetfüggetlen nyelvtanban van balrekurzív változó, akkor G semmilyen k-ra sem $LL(k)$.

Példa. Legyen adva a következő nyelvtan:

$G = (\{S\}, \{a, b\}, \{S \to Sa, S \to b\}, S)$.

Ekkor $S \overset{*}{\Longrightarrow} Sa^n$. Meg kell vizsgálnunk a következő halmazokat:

$FIRST_k(Saa^n) = \{ba^{k-1}\}$ és $FIRST_k(ba^n) = \{ba^{k-1}\}$.

Látszik, hogy egybeesnek tetszőleges $k \geq 1$ értékre, tehát a metszetük nem lehet üres. A nyelvtan nem $LL(0)$ sem, így G nem $LL(k)$ egyetlen k-ra sem.

A $FIRST_k(\alpha)$ halmazok meghatározása

Először meghatározzuk a $FIRST_k(A)$ halmazokat, ahol A változó.

Ehhez generáljuk az összes $A \stackrel{*}{\Longrightarrow} u\alpha$ alakú levezetéseket, ahol $u \in T^*$ és $|u| \geq k$.

Ha $a \in T$, akkor $FIRST_k(a) = \{a\}$.

Tetszőleges $\alpha = X_1X_2 \ldots X_n$ és $X_i \in N \cup T$ ($i = 1, 2, \ldots, n$) esetében:

$$FIRST_k(X_1X_2 \ldots X_n) =$$

$$FIRST_k\Big(FIRST_k(X_1)FIRST_k(X_2) \ldots FIRST_k(X_n)\Big),$$

ahol a jobb oldali zárójelben szóhalmazok szorzata (konkatenációja) szerepel.

A $FIRST_1(X)$, $X \in N$ meghatározására létezik egy jól algoritmizálható módszer, amely relációk tranzitív lezártjának a kiszámításán alapszik.

Értelmezzük a $G = (N, T, P, S)$ nyelvtan esetében a következő bináris relációt:

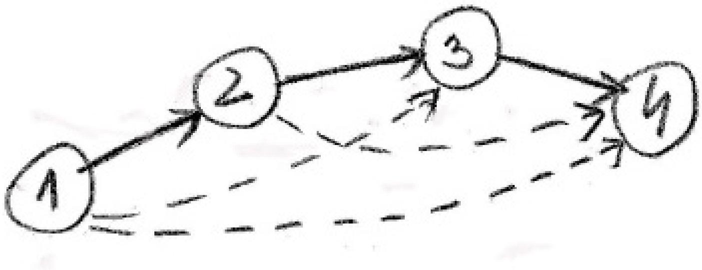
$$\rho \subseteq N \times (N \cup T), \quad X\rho Y, \text{ ha létezik az } X \to Y\alpha \text{ szabály.}$$

Ennek a relációnak a mátrixára alkalmazzuk a Warshall-algoritmust a tranzitív lezárt kiszámítására. A mátrix $A = (a_{ij})_{i=\overline{1,m}, j=\overline{1,n}}$, ahol

$$a_{ij} = \begin{cases} 1 & \text{ha } X_i \rho Y_j \ (X_i \in N, \ Y_j \in N \cup T) \\ 0 & \text{különben} \end{cases}$$

A Warshall-algoritmus esetünkben a következőképpen írható le. Az algoritmus bemenete a reláció A mátrixa, ahol $j = 1, 2, \ldots, m$-re $Y_j = X_j \ \Big(|N| = m,\ |N \cup T| = n\Big)$, eredménye pedig a tranzitív lezárt R mátrixa.

tranzitív lezárt:



$\textsc{Warshall}(A)$

$R := A$
for $k = 1$ **to** m
    do for $i = 1$ **to** m
        do for $j = 1$ **to** n
            do if $r_{ik} = 1$ **és** $r_{kj} = 1$
                then $r_{ij} := 1$
return R

Ekkor

$$FIRST_1(X_i) = \{Y_j \mid r_{ij} = 1,\ j = m+1, \ldots, n\}, \quad i = 1, 2, \ldots m$$

Példa.

Legyen $G = (\{E, T, F\}, \{a, +, *, (,)\}, P, E)$, ahol a szabályok a következők:

$E \to E + T \mid T$
$T \to T * F \mid F$
$F \to (E) \mid a$

Ekkor az A mátrix a következő:

	E	T	F	$+$	$*$	(	)	a
E	1	1	0	0	0	0	0	0
T	0	1	1	0	0	0	0	0
F	0	0	0	0	0	1	0	1

Az eredménymátrix pedig, amelyet a Warshall-algoritmussal kapunk meg:

	E	T	F	$+$	$*$	(	)	a
E	1	1	1	0	0	1	0	1
T	0	1	1	0	0	1	0	1
F	0	0	0	0	0	1	0	1

Innen pedig azt kapjuk, hogy:

$$FIRST_1(E) = \{(, a\}, \quad FIRST_1(T) = \{(, a\} \quad FIRST_1(F) = \{(, a\}$$

Ez a nyelvtan nem $LL(1)$ típusú, hisz például az $E \to E + T$ és $E \to T$ szabályokra

$$FIRST_1(E + T) = FIRST_1(T) = \{(, a\}.$$

Bizonyítás nélkül közöljük a következő eredményeket:

1) Tetszőleges k-ra létezik olyan algoritmus, amely eldönti egy nyelvtanról, hogy $LL(k)$ típusú-e.

2) Tetszőleges pozitív k-ra létezik olyan $LL(k+1)$ nyelvtan, amelyik nem ekvivalens egyetlen $LL(k)$ nyelvtannal sem.

$LR(k)$ nyelvtanok

Hasonlóan az $LL(k)$ nyelvtanokhoz, értelmezhetjük az $LR(k)$ nyelvtanokat, de itt legbaloldalibb levezetések helyett legjobboldalibbakat használunk. Ezért van nevében az L (leftmost) helyett R (rightmost).

A legjobboldalibb levezetéseknél mindig a szóban lévő utolsó (legjobboldalibb) változót helyettesítjük.

Az $LR(k)$ nyelvtanok esetében a szavak felismerését „fordított" levezetéssel (redukálással) valósítjuk meg. Azaz a szóban megkeressük egy szabály jobb oldalát, majd helyettesítjük a bal oldali változóval, azaz redukáljuk a bal oldali változóra. Ezt mindadig végezzük, amíg el nem érünk a nyelvtan kezdőszimbólumához. Vegyük például a

$$G = \big(\{A, B\}, \{a, +, *\}, \{A \to A + B \mid B,\ B \to B * B \mid a\}, A\big)$$

nyelvtant. Vizsgáljuk meg, hogy az $a + a * a$ szó eleme-e $L(G)$-nek. A következő redukálásokat végezzük:

$a + a * a$	
$B + a * a$,	mivel létezik $B \to a$ szabály,
$A + a * a$,	mivel létezik $A \to B$ szabály,
$A + B * a$,	mivel létezik $B \to a$ szabály,
$A + B * B$,	mivel létezik $B \to a$ szabály,
$A + B$,	mivel létezik $B \to B * B$ szabály,
A,	mivel létezik $A \to A + B$ szabály.

Mivel megkaptuk a nyelvtan kezdőszimbólumát, a szó benne van a G által generált nyelvben. Természetesen, nem mindig könnyű eldönteni, hogy hol melyik jobb oldalt helyettesítsük, ha több lehetőség is van.

Ha például, a negyedik sorban $A + B$-t helyettesítjük A-val, nem jutunk helyes eredményre.

Az $LR(k)$ nyelvtanok pont ebben segítenek, mégpedig úgy, hogy k betű előre megnézésével el lehet dönteni, hogy melyik szabályt kell alkalmaznunk. A fenti G nyelvtan $LR(1)$ típusú. Például a negyedik sorban, ha a B betű után megnézzük a következőt (itt $*$), eldönthetjük, hogy tovább kell lépnünk, ha $+$ jel lett volna, akkor lehetett volna alkalmazni redukálásra az $A \to A + B$ szabályt.

A fenti redukálássorozat („fordított” levezetés) megfelel egy legjobboldali levezetésnek.

$$A \Longrightarrow A + B \Longrightarrow A + B * B \Longrightarrow A + B * a \Longrightarrow A + a * a \Longrightarrow$$
$$\Longrightarrow B + a * a \Longrightarrow a + a * a.$$

Csak olyan nyelvtanokkal foglalkozunk, amelyekben egyetlen szabály kezdődik a kezdőszimbólummal, és ez nem szerepel egyetlen szabály jobb oldalán se.

Egy $G = (N, T, P, S)$ környezetfüggetlen nyelvtant $LR(k)$ nyelvtannak ($k \geq 0$) nevezünk, ha tetszőleges

$$S \stackrel{*}{\Longrightarrow} \alpha A x \Longrightarrow \alpha\beta x$$
$$S \stackrel{*}{\Longrightarrow} \gamma B y \Longrightarrow \gamma\delta y = \alpha\beta z$$

legjobboldalibb levezetéspárra abból, hogy $FIRST_k(x) = FIRST_k(z)$ következik, hogy $\alpha = \gamma,\ \beta = \delta,\ A = B$ (és következésképpen $y = z$) (ahol $\alpha, \beta, \gamma \in (N \cup T)^*,\ x, y, z \in T^*,\ A, B \in N$).

Példák.

1) A $G_1 = \big(\{S,A\},\{a\},\{S \to A,\ A \to aAb \mid aa \mid a\},S\big)$ **nyelvtan nem** $LR(k)$ **egyetlen** k**-ra sem, mert**

$$S \Longrightarrow A \overset{*}{\Longrightarrow} \underbrace{a^k}_{\alpha}\underbrace{A}_{A}\underbrace{b^k}_{x} \Longrightarrow a^{k+2}b^k = \underbrace{a^k}_{\alpha}\underbrace{aa}_{\beta}\underbrace{b^k}_{x}$$

$$S \Longrightarrow A \overset{*}{\Longrightarrow} \underbrace{a^{k+1}}_{\gamma}\underbrace{A}_{B}\underbrace{b^{k+1}}_{y} \Longrightarrow a^{k+2}b^{k+1} = \underbrace{a^{k+1}}_{\gamma}\underbrace{a}_{\delta}\underbrace{b^{k+1}}_{y} = \underbrace{a^k}_{\alpha}\underbrace{aa}_{\beta}\underbrace{b^{k+1}}_{z}$$

és $FIRST_k(x) = FIRST_k(z)$**, de ennek ellenére** $\alpha = a^k \neq a^{k+1} = \gamma$**, tehát** G_1 **semmilyen** k**-ra sem lehet** $LR(k)$**.**

2) $G_2 = \big(\{S,E,T\},\{a,+,*\},\{S \to E,\ E \to E{+}T \mid T,\ T \to T{*}T \mid a\},S\big)$
Ez a nyelvtan nem $LR(0)$**, mert**

$$S \overset{*}{\Longrightarrow} \underbrace{\varepsilon}_{\alpha}\underbrace{S}_{A}\underbrace{\varepsilon}_{x} \Longrightarrow \underbrace{\varepsilon}_{\alpha}\underbrace{E}_{\beta}\underbrace{\varepsilon}_{x}$$

$$S \overset{*}{\Longrightarrow} \underbrace{\varepsilon}_{\gamma}\underbrace{E}_{B}\underbrace{\varepsilon}_{y} \Longrightarrow \underbrace{\varepsilon}_{\gamma}\underbrace{E+T}_{\delta}\underbrace{\varepsilon}_{y} = \underbrace{\varepsilon}_{\alpha}\underbrace{E}_{\beta}\underbrace{+T}_{z}$$

és $FIRST_0(x) = FIRST_0(z) = \emptyset$**, ennek ellenére** $y \neq z$**.**

A G_2 **nyelvtan ellenben** $LR(1)$**. Egy betű előreolvasásával mindig el lehet dönteni, hogy melyik szabályt kell alkalmazni.**

Egy nyelv $LR(k)$ $\big($**vagy** $LL(k)\big)$ **típusú, ha létezik olyan** $LR(k)$ $\big($**vagy** $LL(k)\big)$ **nyelvtan, amelyik generálja.**

Bizonyítás nélkül közöljük a következő eredményeket.

1) Létezik olyan algoritmus, amely tetszőleges k-ra egy nyelvtanról eldönti, hogy $LR(k)$ típusú-e.

2) Minden $LR(k)$ nyelvtanhoz megadható egy vele ekvivalens $LR(1)$ nyelvtan.

3) Minden $LL(k)$ nyelvtan egyben $LR(k)$ is, de fordítva ez nem igaz. Például egy $LL(0)$ nyelvtan csupán egyetlen szót generál, míg egy $LR(0)$ akár végtelen nyelvet is generálhat.

4) Az $LR(1)$ nyelvek osztálya megegyezik a determinisztikus veremautomaták által felismert nyelvek (ún. determinisztikus környezetfüggetlen nyelvek) osztályával.

5) Az $LR(0)$ nyelvek osztálya megegyezik a kezdőszeletmentes determinisztikus környezetfüggetlen nyelvek osztályával.

Egy L nyelv akkor kezdőszeletmentes, ha bármely $u \in L$ szóra igaz az, hogy u egyetlen valódi kezdőszelete (prefixe) sem eleme L-nek.

Elsőbbségi (precedencia) nyelvtanok

Az ábécé betűi között relációkat határozunk meg, és ezek segítségével lehet eldönteni, hogy milyen redukálásokat kell a vizsgálandó szóban végezni egy alulról felfelé történő elemzésben.

Az egyszerűség kedvéért bevezetünk még egy új határoló jelet, a $-t, amely közrefogja a vizsgálandó szót.

A következő relációkat használjuk $\lessdot$ (kisebb), $\gtrdot$ (nagyobb) és $\doteq$ (egyenlő). Ezek nem feltétlenül tranzitív, illetve ekvivalencia relációk, és két jel között többféle reláció is létezhet. A relációkat a következőképpen értelmezzük.

$\$ \lessdot X$ ha létezik $S \stackrel{+}{\Longrightarrow} X\beta$ alakú levezetés,
$X \lessdot Y$ ha létezik $A \to \alpha X B \beta$ alakú szabály, és $B \stackrel{+}{\Longrightarrow} Y\gamma$,
$X \gtrdot a$ ha létezik $A \to \alpha B Y \beta$ alakú szabály, és $B \stackrel{+}{\Longrightarrow} \gamma X$, $Y \stackrel{*}{\Longrightarrow} a\delta$
$X \doteq Y$ ha létezik $A \to \alpha X Y \beta$ alakú szabály,
$X \gtrdot \$$ ha létezik $S \stackrel{+}{\Longrightarrow} \alpha X$ alakú levezetés,

ahol $X, Y \in N \cup T$, $a \in T$, $A, B \in N$, $\alpha, \beta, \gamma, \delta \in (N \cup T)^*$.

A $\lessdot$, $\doteq$ és $\gtrdot$ relációk meghatározására vezessük be a következő relációkat, ahol $A \in N$, $X \in N \cup T$, $\alpha, \beta \in (N \cup T)^*$:

$A \, \varrho_l \, X$ ha létezik $A \to X\beta$ szabály,
$A \, \varrho_r \, X$ ha létezik $A \to \alpha X$ szabály.

Ezen relációkat az $L = (l_{ij})$ és $R = (r_{ij})$ mátrixok segítségével ábrázoljuk.

Használjuk a következő jelöléseket:

$N = \{X_1, X_2, \ldots, X_m\}$,

$N \cup T = \{Y_1, Y_2, \ldots Y_n\}$ (ahol $Y_i = X_i$, ha $i = 1, 2, \ldots, m$).

$$l_{ij} = \begin{cases} 1, & \textbf{ha } X_i\, \varrho_l\, Y_j, \\ 0, & \textbf{különben.} \end{cases} \qquad i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$$

$$r_{ij} = \begin{cases} 1, & \textbf{ha } X_i\, \varrho_r\, Y_j, \\ 0, & \textbf{különben.} \end{cases} \qquad i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$$

A továbbiakban a számításokat következő példán mutatjuk be.

Legyen $G = (\{S, A, B\}, \{+, *, a\}, P, S)$, ahol a P szabályai:

$S \to A$
$A \to A + B \mid B$
$B \to C * B \mid C$
$C \to a$

L mátrix:

	S	A	B	C	$+$	$*$	a
S	0	1	0	0	0	0	0
A	0	1	1	0	0	0	0
B	0	0	0	1	0	0	0
C	0	0	0	0	0	0	1

R mátrix:

	S	A	B	C	$+$	$*$	a
S	0	1	0	0	0	0	0
A	0	0	1	0	0	0	0
B	0	0	1	1	0	0	0
C	0	0	0	0	0	0	1

A $\lessdot$, $\doteq$ és $\gtrdot$ relációk meghatározásában szükségünk lesz ezen relációk tranzitív lezártjára. A tranzitív lezárt könnyen megkapható a már ismertetett Warshall-algortimussal. A szokásos jelölés a tranzitív lezárt mátrixára: L^+ és R^+.

L^+ mátrix:

	S	A	B	C	$+$	$*$	a
S	0	1	1	1	0	0	1
A	0	1	1	1	0	0	1
B	0	0	0	1	0	0	1
C	0	0	0	0	0	0	1

R^+ mátrix:

	S	A	B	C	$+$	$*$	a
S	0	1	1	1	0	0	1
A	0	0	1	1	0	0	1
B	0	0	1	1	0	0	1
C	0	0	0	0	0	0	1

Ezen mátrixok segítségével, a relációk értelmezése alapján elkészíthetjük az elsőbbségi (precedencia) táblázatot.

ELőször beírjuk az összes $\doteq$ relációt, megvizsgálva minden szabály jobb oldalát. Az egymást követő betűk között lesz $\doteq$ reláció.

Majd a $\$ \lessdot X$ meghatározására megnézzük az L^+ mátrixban az S-nek megfelelő sort. Az A, B, C és a oszlopában van egy-egy 1-es. Ezért az elsőbbségi mátrix \$-nak megfelelő sorába beírunk egy-egy $\lessdot$ jelet az A, B, C és a oszlopokba.

Hasonlóképen járunk el az $X \gtrdot \$$ meghatározásban, csak itt az R^+ mátrix segítségével dolgozunk.

Az $X \lessdot Y$ típusú relációknál megkeressük a szabályok jobb oldalán a szomszédos betűket, például $+B$. Megnézzük az L^+ mátrix B-nek megfelelő sorát, itt a C és a oszlopokban van egy-egy 1-es, ezért az elsőbbségi táblázat $+$ jelnek megfelelő sorába beírunk egy-egy $\lessdot$ jelet a C és a oszlopokba.

Az eredmény a következő táblázatban látható:

	S	A	B	C	$+$	$*$	a	$\$$
S								
A					$\doteq$			$\gtrdot$
B					$\gtrdot$			$\gtrdot$
C					$\gtrdot$	$\doteq$		$\gtrdot$
$+$			$\doteq$	$\lessdot$			$\lessdot$	
$*$			$\doteq$	$\lessdot$			$\lessdot$	
a					$\gtrdot$	$\gtrdot$		$\gtrdot$
$\$$		$\lessdot$	$\lessdot$	$\lessdot$			$\lessdot$	

Szerencsénkre a fenti táblázatban minden betűpár között legfönnebb egy reláció van. Az üres mező jelzi mindhárom reláció hiányát. Az ilyen nyelvtanok alkalmasak arra, hogy hatékonyan felismerhessük a szvaikat egy alulról felfelé történő elemzésssel.

Egy környezetfüggetlen nyelvtant egyszerű elsőbbségi (precedencia) nyelvtannak hívunk, ha
• ε-mentes és nincs olyan változója, amelyre $A \stackrel{+}{\Longrightarrow} A$,
• a $N \cup T \cup \{\$\}$ halmaz bármely két eleme között a $\lessdot$, $\doteq$ és $\gtrdot$ relációk közül legfeljebb egy áll fenn,
• a helyettesítési szabályok jobb oldalai különböznek egymástól.

Egy egyszerű elsőbbségi nyelvtan segítségével könnyen lehet elemezni, hogy egy szó eleme a nyelvtan által generált nyelvnek vagy sem.

Megvizsgáljuk a szó szomszédos betűi közötti relációkat, minden esetben (ha a szó felismerhető a nyelvtan által) van olyan része, amely $\lessdot$ jellel kezdődik, utána esetleg $\doteq$ jelek következnek, majd a végén $\gtrdot$ van (ez a nyél). A nyél mindig egy szabály jobb oldala,

és ez redukálható a szabály bal oldalára. Ezt folytatva, ha a szó eleme a nyelvnek, el lehet jutni a kezdőszimbólumhoz.

Nézzük ezt meg a következő példán! Az $a + a * a$ szót vizsgáljuk.

$\$ \lessdot a \gtrdot + \; a \; * \; a \; \$$	**redukálás** $C \to a$
$\$ \lessdot C \gtrdot + \; a \; * \; a \; \$$	**redukálás** $B \to C$
$\$ \lessdot B \gtrdot + \; a \; * \; a \; \$$	**redukálás** $A \to B$
$\$ \lessdot A \doteq + \lessdot a \gtrdot * \; a \; \$$	**redukálás** $C \to a$
$\$ \lessdot A \doteq + \lessdot C \doteq * \lessdot a \gtrdot \$$	**redukálás** $C \to a$
$\$ \lessdot A \doteq + \lessdot C \doteq * \lessdot C \gtrdot \$$	**redukálás** $B \to C$
$\$ \lessdot A \doteq + \lessdot C \doteq * \doteq B \gtrdot \$$	**redukálás** $B \to C * B$
$\$ \lessdot A \doteq + \doteq B \gtrdot \$$	**redukálás** $A \to A + B$
$\$ \lessdot A \gtrdot \$$	**redukálás** $S \to A$

$\$ \quad S \quad \$$

Ez megfelel a következő (legjobboldalibb) levezetésnek:

$S \Rightarrow A \Rightarrow A + B \Rightarrow A + C * B \Rightarrow A + C * C \Rightarrow A + C * a \Rightarrow A + a * a$
$\Rightarrow B + a * a \Rightarrow C + a * a \Rightarrow a + a * a$