ELMÉLETI INFORMATIKA

I. rész

Formális nyelvek és automaták

Szintaktikai elemzés

6. előadás

Szintaktikai elemzés

A felhasználó által egy magasabb szintű programozási nyelven megírt program forráskódja és a számítógép által végrehajtható gépi kód jelentős mértékben különböznek egymástól. Ezért szükséges, hogy ezt a magasabb szintű programozási nyelven megírt kódból egy alacsonyabb szintű programot készítsünk. Ezt az átalakítást végző fordítóprogramot **compiler**nek nevezzük.

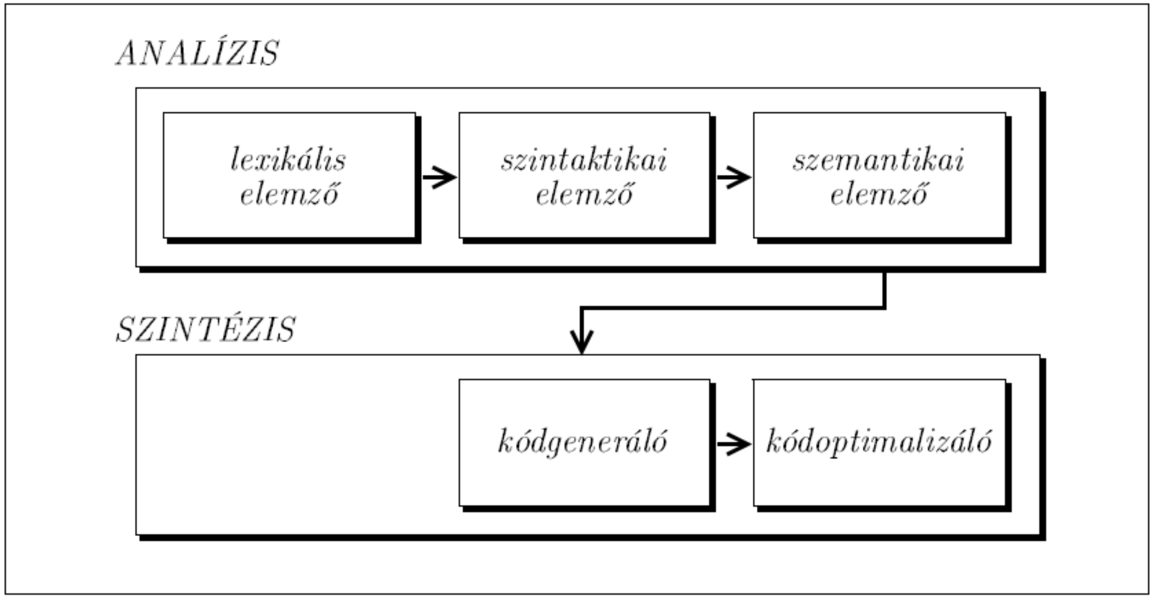
A **compiler** inputja a forrásnyelven megírt program (forrásprogram),
outputja pedig a tárgynyelvű program (tárgyprogram).

A **compiler**nek két jelentős feladatot kell megoldania:

- elemeznie kell az inputként kapott forráskódot,
- szintetizálnia kell a tárgykódot.

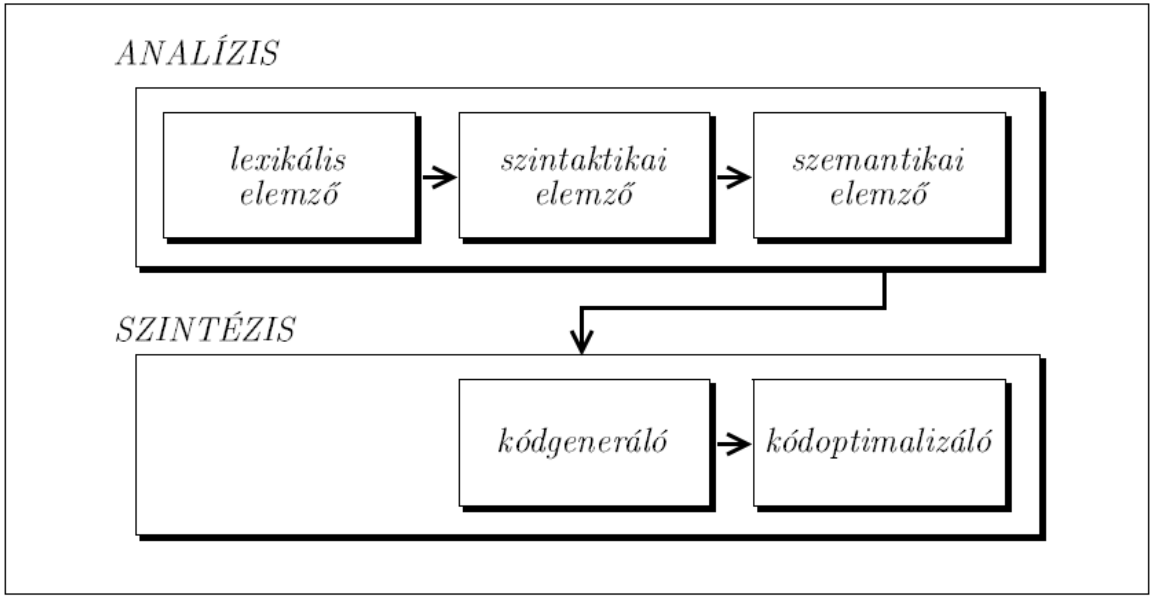
Az analízis során a forráskód karaktersorozata részekre bontódik, míg a szintézis során az egyes részeknek megfelelő tárgykódokból épül fel a fordítandó program teljes tárgykódja.

Szintaktikai elemzés



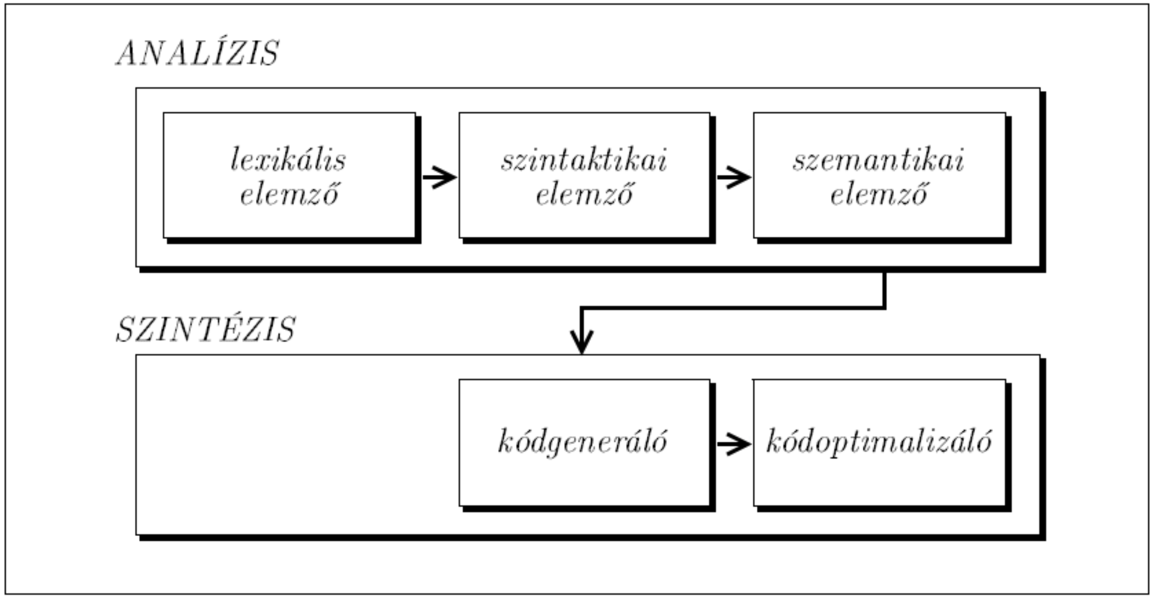
A **lexikális elemző** a forráskód karaktersorozatában meghatározza az egyes szimbolikus egységeket, a kulcsszavakat, változókat, konstansokat és operátorokat. Kiszűri a forráskódbeli szóközöket és kommenteket, mivel ezek nem játszanak szerepet a tárgykód elkészítésében. A lexikális elemző a karaktersorozatból szimbólum-sorozatot készít, s ez az output lesz a szintaktikai elemző inputja.

Szintaktikai elemzés



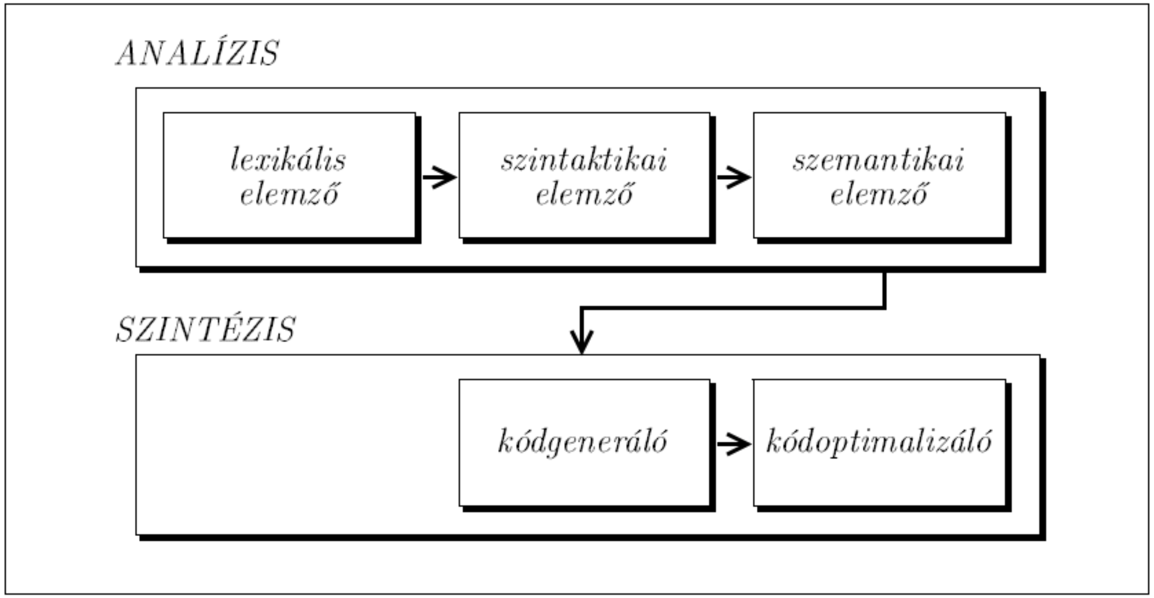
A **szintaktikai elemző** feladata annak vizsgálata, hogy az input szimbólumsorozatban az egyes szimbólumok a megfelelő helyen vannak-e, a szimbólumok sorrendje megfelel-e az adott programnyelv szintaktikai szabályainak. Outputja az elemzett program ***derivációs fája*** (más néven szintaxisfája).

Szintaktikai elemzés



A **szemantikai elemző** bizonyos szemantikai jellegű tulajdonságokat vizsgál. Pl. amikor egy változóhoz egy konstanst adunk hozzá, vizsgálja, hogy az adott azonosítóval rendelkező változó deklarálva van-e, továbbá a konstansnak van-e értéke, és a típusuk azonos-e. Outputja a derivációs fa formájában megadott elemzett program, s ez lesz a szintetizálást végző programok inputja.

Szintaktikai elemzés



A **szintézis** első lépése a kódgenerálás, amelyet a **kódgeneráló** program végez el. Ennek eredménye a *tárgykód*, amely leggyakrabban az adott számítógép assembly nyelvű vagy gépi kódú programja. A szintézis második lépése a **kódoptimalizáló** által végrehajtott kódoptimalizálás, melynek eredménye egy hatékonyabb tárgyprogram.

Szintaktikai elemzés alapfeladata

Adjunk meg olyan algoritmust, amely tetszőleges $G = (N, \Sigma, P, S)$ környezetfüggetlen nyelvtan és $w \in \Sigma^*$ szó esetén eldönti, hogy $w \in L(G)$ teljesül-e.

Az elemzési algoritmusok két típusával fogunk foglalkozni:

- **Felülről lefelé** (Top-Down) **haladó elemzési algoritmusok** – az ilyen algoritmus az S kezdő nemterminális szimbólumból kiindulva próbál meg felépíteni egy olyan derivációs fát, amelynek határa w. Amennyiben ez sikerül, az output IGEN, ellenkező esetben az output NEM.
- **Alulról felfelé** (Bottom-Up) **haladó elemzési algoritmusok** – az ilyen algoritmus a w szóból kiindulva próbál meg felépíteni egy olyan derivációs fát, amelynek gyökere S. Amennyiben ez sikerül, az output IGEN, ellenkező esetben az output NEM.

Általános felülről lefelé haladó szintaktikai elemzés

A felülről lefelé haladó szintaktikus elemzés során két alapművelettel dolgozunk:

1) **kiterjesztés** – ezen művelet során a derivációs fában egy nemterminális szimbólumot helyettesítünk annak valamely alternatívájával (a művelet során fellépő probléma a megfelelő alternatíva megtalálása),
2) **illesztés** – ezen művelet annak az ellenőrzését jelenti, hogy a kiterjesztésnél alkalmazott alternatívában szereplő terminális szimbólumok illeszkednek-e az elemezendő szó megfelelő részéhez.

Egy szó akkor eleme az $L(G)$ nyelvnek, ha az elemzése során alkalmazott valamennyi kiterjesztés illeszthető.

Általános felülről lefelé haladó szintaktikai elemzés

Alapötlet: Legyen adott a $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és legyen adott a $w \in \Sigma^*$ szó.

Minden $A \in N$ nemterminális szimbólumra rögzítsük le az A szimbólumot a bal oldalon tartalmazó szabályok jobb oldalainak egy sorrendjét, azaz fixáljuk az A alternatíváinak egy $A \longrightarrow \gamma_1 \mid \gamma_2 \mid \dots \mid \gamma_k$ sorrendjét.

Induljunk ki az S kezdő nemterminális szimbólumból mint gyökérből, és próbáljuk meg felépíteni egy olyan derivációs fát, amelynek határa a w szó.

Ehhez első lépésként az S kezdő nemterminális szimbólumot kell kiterjesztenünk. Tekintsük az S alternatíváit, legyenek ezek $S \longrightarrow \alpha_1 \mid \dots \mid \alpha_k$.

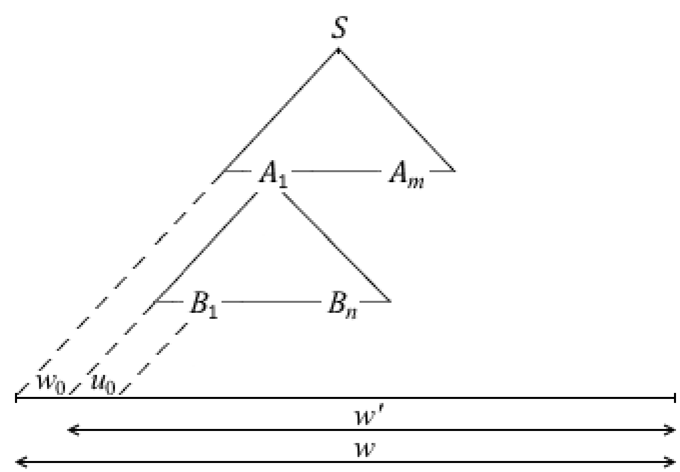
Mivel nem tudjuk, hogy melyik alternatívát kell alkalmazni, vegyük az általunk lerögzített sorrendben az elsőt. Tegyük fel, hogy $\alpha_1 = w_0 A_1 w_1 \dots A_m w_m$. Ezután ellenőrizzük, hogy w_0 illeszkedik-e a w szó elejéhez, azaz érvényes-e, hogy $w = w_0 w'$ (ha $w_0 = \lambda$, akkor ez nyilvánvalóan teljesül). Ennek az illeszkedésnek ugyanis szükségszerűen teljesülnie kell ahhoz, hogy az $S \Longrightarrow \alpha_1 = w_0 A_1 w_1 \dots A_m w_m \Longrightarrow^* w$ teljesülhessen.

Általános felülről lefelé haladó szintaktikai elemzés

Ha w_0 nem illeszkedik-e a w szó elejéhez, akkor menjünk tovább és keressünk olyan legkisebb i-t, amelyre α_i eleje illeszkedik a w szó elejéhez. Amennyiben nem találunk ilyen α_i-t, akkor $w \notin L(G)$ és az elemzési algoritmus NEM választ küld az outputra, majd leáll.

Tegyük fel, hogy az i-edik alternatíva megfelelő volt, vagyis $\alpha_1 = w_0 A_1 w_1 \ldots A_m w_m$ és $w = w_0 w'$.

Ezután azt kell kideríteni, hogy $A_1 w_1 \ldots A_m w_m \Longrightarrow^* w'$ teljesül-e. Ehhez az A_1 alternatívái között kell keresnünk, vagyis az A_1 nemterminális szimbólumot kell kiterjesztenünk (lásd ábra).



Általános felülről lefelé haladó szintaktikai elemzés

Legyenek az A_1 alternatívái $A_1 \longrightarrow \beta_1 \mid \beta_2 \mid \dots \mid \beta_l$. Keressük meg az általunk lerögzített sorrendben az első olyat, amelyik illeszkedik a w' szó elejéhez. Legyen ez a j-edik alternatíva, tehát legyen $\beta_j = u_0 B_1 u_1 \dots B_n u_n$, ahol $w' = u_0 w''$. Ez ugyanis szükséges feltétel ahhoz, hogy az

$$A_1 w_1 \dots A_m w_m \Longrightarrow u_0 B_1 u_1 \dots B_n u_n w_1 \dots A_m w_m \Longrightarrow^* w'$$

teljesülhessen.

Amennyiben így haladunk tovább, felmerül a következő kérdés: Mi van akkor, ha egy nemterminális szimbólum (pl. A_1) kiterjesztése estén nem találunk olyan alternatívát, amely illeszkedik a w szó megmaradt részéhez (ami ebben az esetben a w' szó). Ekkor nyilván nem mondhatjuk azt, hogy $w \notin L(G)$, mivel valamennyi előző kiterjesztésnél (jelen esetben az S kezdő nemterminális szimbólum kiterjesztésénél) a legelső olyan alternatívát választottuk, ami illeszkedett. Ezért, ha valamely szinten nem találunk megfelelő alternatívát, akkor egy szintet vissza kell mennünk (ún. ***backtrack***-et hajtunk végre), és az ott korábban választott alternatíva után következő első olyan alternatívát kell választanunk, amely illeszkedik.

Általános felülről lefelé haladó szintaktikai elemzés

Amennyiben nincs ilyen alternatíva, akkor további szintet esetleg szinteket kell visszamenni. Ha ilyen módon találunk a w szónak egy, az S kezdő nemterminális szimbólumból kiinduló levezetését, akkor $w \in L(G)$. Így a w szónak egy baloldali levezetését kapjuk meg, mivel a már elért mondatformában (ami kezdetben S) mindig a legbaloldalibb nemterminális szimbólumból kiterjesztését keressük meg. Ha a backtrack-ek során valamennyi illeszkedő lehetőséget kipróbáltuk és nem találtunk a w szónak egy, az S kezdő nemterminális szimbólumból kiinduló levezetését, akkor $w \notin L(G)$.

Általános felülről lefelé haladó szintaktikai elemzés

6.1 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és egy $w = a_1 a_2 \ldots a_n \in \Sigma^*$ (ahol $n \geq 0$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy baloldali levezetése, egyébként NEM.

Módszer: Minden $A \in N$ nemterminális szimbólumra rögzítsük le az A alternatíváinak egy $A \longrightarrow \gamma_1 \mid \gamma_2 \mid \ldots \mid \gamma_k$ sorrendjét. Az A nemterminális szimbólum i-edik alternatívájára A_i-vel fogunk hivatkozni.

Maga az elemzés (s, i, α, β) alakú konfigurációk sorozata, ahol

- s az elemzés állapota, $s \in \{q, b, t\}$. Értéke lehet q (*normál*), b (*backtrack*) és t (*elfogadó*). Kezdő értéke q.
- i egy mutató (pointer), amely a w szó i-edik szimbólumára mutat. Értéke lehet $1 \leq i \leq n + 1$, ahol $n = |w|$. Kezdő értéke 1. Ha $i = n + 1$, akkor definíció szerint $a_{n+1} = \$$, ahol $\$ \notin \Sigma$ (erre a szimbólumra technikai okok miatt van szükség).

Általános felülről lefelé haladó szintaktikai elemzés

6.1 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és egy $w = a_1 a_2 \ldots a_n \in \Sigma^*$ (ahol $n \geq 0$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy baloldali levezetése, egyébként NEM.

Módszer: Minden $A \in N$ nemterminális szimbólumra rögzítsük le az A alternatíváinak egy $A \longrightarrow \gamma_1 \mid \gamma_2 \mid \ldots \mid \gamma_k$ sorrendjét. Az A nemterminális szimbólum i-edik alternatívájára A_i-vel fogunk hivatkozni.

Maga az elemzés (s, i, α, β) alakú konfigurációk sorozata, ahol

- α egy verem, amelynek teteje a jobb végén van. Ez a verem fogja tartalmazni a szintaktikus elemzés mindenkori történetét azért, hogy *backtrack* estén tudjuk mely alternatívákat választottuk korábban. Ez a verem teszi lehetővé azt is, hogy a w szó esetleges elfogadása esetén ki tudjuk adni az outputra a w szó baloldali levezetését. Kezdő értéke λ.

Általános felülről lefelé haladó szintaktikai elemzés

6.1 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és egy $w = a_1 a_2 \dots a_n \in \Sigma^*$ (ahol $n \geq 0$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy baloldali levezetése, egyébként NEM.

Módszer: Minden $A \in N$ nemterminális szimbólumra rögzítsük le az A alternatíváinak egy $A \longrightarrow \gamma_1 \mid \gamma_2 \mid \dots \mid \gamma_k$ sorrendjét. Az A nemterminális szimbólum i-edik alternatívájára A_i-vel fogunk hivatkozni.

Maga az elemzés (s, i, α, β) alakú konfigurációk sorozata, ahol

- β egy verem, amelynek teteje a bal végén van. Ez a verem fogja tartalmazni a levezetett baloldali mondatformának azt a részét, amelyet még ki kell terjeszteni. Ez a verem aktív verem, a tetején lévő szimbólumot pedig aktív szimbólumnak nevezzük, amelyet vagy illeszteni kell (ha az terminális szimbólum) vagy kiterjeszteni (ha az nemterminális szimbólum). Kezdő értéke S.

Általános felülről lefelé haladó szintaktikai elemzés

6.1 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és egy $w = a_1 a_2 \ldots a_n \in \Sigma^*$ (ahol $n \geq 0$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy baloldali levezetése, egyébként NEM.

Módszer: Minden $A \in N$ nemterminális szimbólumra rögzítsük le az A alternatíváinak egy $A \longrightarrow \gamma_1 \mid \gamma_2 \mid \ldots \mid \gamma_k$ sorrendjét. Az A nemterminális szimbólum i-edik alternatívájára A_i-vel fogunk hivatkozni.

Maga az elemzés (s, i, α, β) alakú konfigurációk sorozata.

Az elemzés kezdő konfigurációja a $(q, 1, \lambda, S)$. A befejező konfigurációk $(t, n+1, \alpha, \lambda)$ alakúak, ahol α egy terminális szimbólumokból és alternatívákból álló sorozat. Érvényes tehát, hogy

$$w \in L(G) \iff (q, 1, \lambda, S) \vdash^* (t, n+1, \alpha, \lambda).$$

Általános felülről lefelé haladó szintaktikai elemzés

6.1 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ 2-típusú nyelvtan és egy $w = a_1 a_2 \ldots a_n \in \Sigma^*$ (ahol $n \geq 0$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy baloldali levezetése, egyébként NEM.

FELÜLRŐL_LEFELÉ_ELEMZÉS (G, w)

Kezdő konfiguráció: $C \leftarrow (q, 1, \lambda, S)$

while létezik C', amelyre $C \vdash C'$ **do**

    $C \leftarrow C'$

if $C = (t, n+1, \alpha, \lambda)$ valamely α-ra **then**

    return IGEN

else return NEM

6.1 példa: Legyen adott a $G = (N, \Sigma, P, K)$ 2-típusú nyelvtan, ahol $N = \{K, T\}$, $\Sigma = \{a, b\}$ és $P = \{K \longrightarrow T + K, K \longrightarrow T, T \longrightarrow a, T \longrightarrow b\}$.

Legyen az alternatívák sorrendje a következő:

$$K_1\colon\ K \longrightarrow T + K \qquad K_2\colon\ K \longrightarrow T \qquad T_1\colon\ T \longrightarrow a \qquad T_2\colon\ T \longrightarrow b$$

Általános felülről lefelé haladó szintaktikai elemzési algoritmussal elemezzük a $w = b + a$ szót:

Ekkor $n = 3$, $a_1 = b$, $a_2 = +$, $a_3 = a$, $a_4 = \$$.

$$\begin{aligned}
(q, 1, \lambda, K) &\vdash (q, 1, K_1, T + K) \vdash (q, 1, K_1T_1, a + K) \vdash (b, 1, K_1T_1, a + K) \vdash \\
&\vdash (q, 1, K_1T_2, b + K) \vdash (q, 2, K_1T_2b, +K) \vdash (q, 3, K_1T_2b+, K) \vdash \\
&\vdash (q, 3, K_1T_2b + K_1, T + K) \vdash (q, 3, K_1T_2b + K_1T_1, a + K) \vdash \\
&\vdash (q, 4, K_1T_2b + K_1T_1a, +K) \vdash (b, 4, K_1T_2b + K_1T_1a, +K) \vdash \\
&\vdash (b, 3, K_1T_2b + K_1T_1, a + K) \vdash (q, 3, K_1T_2b + K_1T_2, b + K) \vdash \\
&\vdash (b, 3, K_1T_2b + K_1, T + K) \vdash (q, 3, K_1T_2b + K_2, T) \vdash \\
&\vdash (q, 3, K_1T_2b + K_2, T_1, a) \vdash (q, 4, K_1T_2b + K_2T_1a, \lambda) \vdash \\
&\vdash (t, 4, K_1T_2b + K_2T_1a, \lambda)
\end{aligned}$$

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

Általános alulról felfelé haladó szintaktikai elemzés

6.1 definíció: (ciklusmentes nyelvtan)

Egy G 2-típusú nyelvtan **ciklusmentes**, ha nincs olyan $A \in N$ szimbólum, amelyre $A \Longrightarrow^{+} A$ teljesül.

6.1 tétel: Bármely G 2-típusú nyelvtanhoz szerkeszthető vele ekvivalens λ-mentes és ciklusmentes G' 2-típusú nyelvtan.

Általános alulról felfelé haladó szintaktikai elemzés

Az alulról felfelé haladó szintaktikus elemzés során két alapművelettel dolgozunk:

1) **shiftelés** – ezen művelet a w input szó egy szimbólumának sorrend szerint történő beolvasása,
2) **redukálás** – ezen művelet során egy nyelvtani szabály jobb oldalát a bal oldalával helyettesítjük.

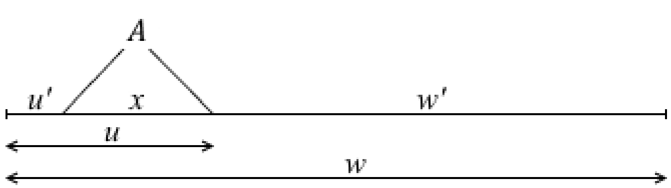
Általános alulról felfelé haladó szintaktikai elemzés

Alapötlet: Legyen adott a $G = (N, \Sigma, P, S)$ λ-mentes és ciklusmentes 2-típusú nyelvtan és legyen adott a $w \in \Sigma^*$ szó.

A w szóból kiindulva megpróbálunk felépíteni egy olyan derivációs fát, amelynek határa w, gyökere pedig S. Az S kezdő nemterminális szimbólumot a w szóból kiindulva shiftelések és redukálások sorozatával próbáljuk elérni.

Vegyük a $w \in \Sigma^*$ szót. Kezdjük el olvasni (shiftelni) az elejéről egészen addig, amíg az elolvasott rész jobb oldali végén ki nem alakul valamelyik nyelvtani szabály jobb oldala.

Legyen u olyan, amelyre teljesül, hogy $w = uw'$ és léteznek olyan u' és x szavak, hogy $u = u'x$ és az $A \longrightarrow x$ szabály benne van a P szabályhalmazban. Most helyettesítsük x-et az előbbi szabály bal oldalával, vagyis az A nemterminális szimbólummal. Eredményül az $u'Aw'$ mondatformát kapjuk, amelyre érvényes, hogy $u'Aw' \Longrightarrow w$ (lásd ábra).



Általános alulról felfelé haladó szintaktikai elemzés

Ezt az eljárást ismét elvégezzük az $u'Aw'$ mondatformára úgy, hogy a w' szó elejétől kezdve annak szimbólumait hozzá shifteljük az $u'A$ mondatformához egészen addig, amíg az így kapott mondatforma jobb oldali végén ismét ki nem alakul valamelyik P halmazbeli szabály jobb oldala. Ekkor ismét redukálunk (a következő olyan szabály jobb oldalába, amely szerint redukálunk, beleeshet az A nemterminális szimbólum, valamint az u' szó egy része is). Az ismertetett eljárást egészen addig ismételjük, amíg el nem érjük az S kezdő nemterminális szimbólumot.

Általános alulról felfelé haladó szintaktikai elemzés

6.2 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ ciklusmentes és λ-mentes 2-típusú nyelvtan és egy $w = a_1 a_2 \dots a_n \in \Sigma^*$ (ahol $n \geq 1$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy jobboldali levezetése, egyébként NEM.

Módszer: Rögzítsük le a P szabályhalmaz szabályainak egy sorrendjét.

Maga az elemzés (p, i, α, β) alakú konfiguráció sorozata, ahol

- p az elemzés állapota, $p \in \{q, b, t\}$. Értéke lehet q (*normál*), b (*backtrack*) és t (*elfogadó*). Kezdő értéke q.
- i egy mutató (pointer), amely a w szó i-edik szimbólumára mutat. Értéke lehet $1 \leq i \leq n+1$, ahol $n = |w|$. Kezdő értéke 1.
- α egy verem, amelynek teteje a jobb végén van. Ez a verem fogja tartalmazni a w szó elolvasott részeiből redukálásokkal keletkezett $(N \cup \Sigma)^*$-beli mondatformát. Kezdő értéke λ.

Általános alulról felfelé haladó szintaktikai elemzés

6.2 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ ciklusmentes és λ-mentes 2-típusú nyelvtan és egy $w = a_1 a_2 \dots a_n \in \Sigma^*$ (ahol $n \geq 1$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy jobboldali levezetése, egyébként NEM.

Módszer: Rögzítsük le a P szabályhalmaz szabályainak egy sorrendjét.

Maga az elemzés (p, i, α, β) alakú konfiguráció sorozata, ahol

- β egy verem, amelynek teteje a bal végén van. Ez a verem fogja tartalmazni az elemzés történetét, vagyis a végrehajtott shiftelések és redukálások sorozatát. Kezdő értéke λ.

Általános alulról felfelé haladó szintaktikai elemzés

6.2 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ ciklusmentes és λ-mentes 2-típusú nyelvtan és egy $w = a_1 a_2 \dots a_n \in \Sigma^*$ (ahol $n \geq 1$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy jobboldali levezetése, egyébként NEM.

Módszer: Rögzítsük le a P szabályhalmaz szabályainak egy sorrendjét.

Maga az elemzés (p, i, α, β) alakú konfiguráció sorozata.

Az elemzés kezdő konfigurációja a $(q, 1, \lambda, \lambda)$. A befejező konfigurációk $(t, n+1, S, \gamma)$ alakúak, ahol γ egy s (shift) szimbólumból és P szabályhalmazbeli szabályok sorszámaiból álló sorozat. Érvényes tehát, hogy

$$w \in L(G) \iff (q, 1, \lambda, \lambda) \vdash^* (t, n+1, S, \gamma).$$

Általános alulról felfelé haladó szintaktikai elemzés

6.2 algoritmus:

Input: egy $G = (N, \Sigma, P, S)$ ciklusmentes és λ-mentes 2-típusú nyelvtan és egy $w = a_1 a_2 \dots a_n \in \Sigma^*$ (ahol $n \geq 1$) szó.

Output: IGEN, ha $w \in L(G)$ és a w szó egy jobboldali levezetése, egyébként NEM.

ALULRÓL_FELFELÉ_ELEMZÉS (G,w)
1 Kezdő konfiguráció: $C \leftarrow (q, 1, \lambda, \lambda)$
2 **while** lehet redukálni **do** redukálás
3 hajts végre shiftelést
4 **goto** 2
5 **if** $C = (q, n+1, S, \beta)$ valamely β-ra **then**
6 $\quad C \leftarrow (t, n+1, S, \beta)$
7 $\quad$ **return** IGEN

Általános alulról felfelé haladó szintaktikai elemzés

8 **if** $C = (q, n+1, \alpha, \beta)$ **and** $\alpha \neq S$ **then** $C' \leftarrow (b, n+1, \alpha, \beta)$

9 **if** $C = (b, i, \alpha A, j\beta)$ **and** a j-edik szabály $A \longrightarrow \gamma$ és az $\alpha\gamma$ redukálható egy további $B \longrightarrow \gamma'$ szabály szerint is **then**

10 $C \leftarrow (q, i, \alpha' B, k\beta)$

11 **goto** 2

12 **if** $C = (b, i, \alpha A, j\beta)$ **and** a j-edik szabály $A \longrightarrow \gamma$ és $i \neq n+1$, de az $\alpha\gamma$ mondatforma nem redukálható más szabállyal **then**

13 $C \leftarrow (q, i+1, \alpha\gamma a_i, s\beta)$

14 **goto** 2

15 **if** $C = (b, n+1, \alpha A, j\beta)$ **and** a j-edik szabály $A \longrightarrow \gamma$, de az $\alpha\gamma$ mondatforma nem redukálható más szabállyal és már shiftelni sem tudunk **then**

16 $C \leftarrow (b, n+1, \alpha\gamma, \beta)$

17 **goto** 8

Általános alulról felfelé haladó szintaktikai elemzés

```
  if C = (b, i, αa, sβ) and i > 1 then
      C ← (b, i − 1, α, β)
      goto 8
return NEM
```

6.2 példa: Legyen adott a $G = (N, \Sigma, P, K)$ 2-típusú nyelvtan, ahol $N = \{K, T\}$, $\Sigma = \{a, b\}$ és $P = \{K \longrightarrow T + K, K \longrightarrow T, T \longrightarrow a, T \longrightarrow b\}$.

Legyen a nyelvtani szabályok sorrendje a következő:

$$1\colon\ K \longrightarrow T + K \qquad 2\colon\ K \longrightarrow T \qquad 3\colon\ T \longrightarrow a \qquad 4\colon\ T \longrightarrow b$$

Általános alulról felfelé haladó szintaktikai elemzési algoritmussal elemezzük a $w = b + a$ szót:

$$(q, 1, \lambda, \lambda) \vdash (q, 2, b, s) \vdash (q, 2, T, 4s) \vdash (q, 2, K, 24s) \vdash (q, 3, K+, s24s) \vdash$$
$$\vdash (q, 4, K + a, ss24s) \vdash (q, 4, K + T, 3ss24s) \vdash$$
$$\vdash (q, 4, K, 13ss24s) \vdash (t, 4, K, 13ss24s)$$

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