Curs 4 - Gramatici și limbaje LL

~~09~~ Martie 2019
M

Prop: Fie $G = (N, \Sigma, S, P)$ g.i.c fără sb. inutilizabile.
Atunci G este $LL(1)$ ddacă este $LL(1)$ tare.

Dem: Fie G de tip $LL(1)$. Pp că G nu este $LL(1)$ tare
Rez că $\exists A \to \beta$, $A \to \gamma \in P$, $\beta \neq \gamma$ a.i.
$First(\beta\, Follow(A)) \cap First(\gamma\, Follow(A)) \neq \phi$
Fie $x \in \ldots \cap \ldots$, $|x| \leq 1$ deci $x \in \Sigma \cup \{\lambda\}$
Avem cazurile:

I) $x \in \Sigma$. Atunci rez. că există derivările stânga:
$\beta \overset{*}{\underset{s}{\Rightarrow}} a\beta'$, $\gamma \overset{*}{\underset{s}{\Rightarrow}} a\gamma'$ (adică $a \in First(\beta) \cap First(\gamma)$)

Deoarece G nu are sb. inutilizabile, rez. că în G $(\exists)$ o derivare de forma
$S \overset{*}{\underset{s}{\Rightarrow}} wA\alpha$. At. avem:

$$S \overset{*}{\underset{s}{\Rightarrow}} wA\alpha \Rightarrow w\beta\alpha \overset{*}{\underset{s}{\Rightarrow}} wa\beta'\alpha$$
$$\Rightarrow w\gamma\beta \overset{*}{\underset{s}{\Rightarrow}} wa\gamma'\alpha$$

Avem $First(a\beta'\alpha) = First(a\gamma'\alpha) = \{a\}$
Cf. cu def. gram $LL(1)$, avem $\beta = \gamma$ $\cancel{\alpha}$

II) $\beta \overset{*}{\underset{s}{\Rightarrow}} a\beta'$; $\gamma \overset{*}{\Rightarrow} \lambda$, $a \in Follow(A)$ $a \in \Sigma$
Rez. că $\exists$ o derivare
$S \overset{*}{\underset{s}{\Rightarrow}} wA\alpha$ și $a \in First(\alpha)$ adică $\alpha \overset{*}{\underset{s}{\Rightarrow}} a\alpha'$

Avem
$$S \overset{*}{\underset{A}{\Rightarrow}} wA\alpha \Rightarrow w\beta\alpha \overset{*}{\underset{s}{\Rightarrow}} wa\beta'\alpha$$
$$\Rightarrow w\gamma\alpha \overset{*}{\Rightarrow} w\alpha \overset{*}{\underset{s}{\Rightarrow}} wa\alpha'$$

$First(a\beta'\alpha) = First(a\alpha') = \{a\}$ deci $\beta = \gamma$ $\cancel{\alpha}$

III. $a = \lambda$; $\beta \overset{*}{\Rightarrow} \lambda$, $\lambda \in Follow(A)$, $\gamma \overset{*}{\Rightarrow} \lambda$
$\lambda \in Follow(A) \Rightarrow \exists$ $S \overset{*}{\underset{s}{\Rightarrow}} wA\alpha$; $\lambda \in First(\alpha)$
deci $\alpha \overset{*}{\underset{s}{\Rightarrow}} \lambda$

Atunci

$$S \xRightarrow[s]{*} wA\alpha \Rightarrow w\beta\alpha \xRightarrow[s]{*} w\alpha \Rightarrow w$$

$$S \xRightarrow[s]{*} wA\alpha \Rightarrow w\gamma\alpha \xRightarrow[s]{*} w\alpha \Rightarrow w$$

$First(\lambda) = First(\lambda) - \{\lambda\}$, G este $LL(1) \Rightarrow \beta = \gamma$

<u>Prop</u>: Orice gram. de tip LL nu este recursivă la stânga.

Algoritm de eliminare a recursivității la stânga:

1) Eliminarea recursiv. imediate la stânga:

$A \to A\alpha$

Fie $G = (N, \Sigma, S, P)$ gic și

$A \to A\alpha_1 \mid A\alpha_2 \mid \dots \mid A\alpha_m \mid \beta_1 \mid \dots \mid \beta_n$

toate A producțiile, $m \geq 1$, $n \geq 1$

$\alpha_i \neq \lambda$, $i = \overline{1, m}$

β_j nu începe cu A, $j = \overline{1, \dots, n}$

Introducem neterminalii noi B, C, D și înlocuim producțiile de mai sus cu:

~~for (i=1, ..., p)~~
~~{ for (j=1, ..., i-1) {~~
~~for (fiecare prod $A_i \to A_j \gamma \in P$)~~
~~înlocuim $A_i \to A_j \gamma$ cu prod~~

$B \to \beta_1 \mid \dots \mid \beta_n$

$C \to \alpha_1 \mid \dots \mid \alpha_m$

$A \to BD \mid B$

$D \to CD \mid C$

2) Eliminarea recursivității la stânga $A \xRightarrow{+} A\alpha$

Fie $G = (N, \Sigma, S, P)$ gic recursivă la stânga fără λ-producții, fără cicluri $A \xRightarrow{+} A$

Pp. că $N = \{A_1, \dots, A_p\}$, $A_1 = S$

```
for (i = 1..p) {
    for (j = 1, .., i-1) {
        for (fiecare prod. Ai → Aj γ ∈ P)
            înlocuim Ai → Aj γ cu prod.
                Ai → β1γ | ... | βsγ, unde
```

$\vdash A_j \to \beta_1 | \dots | \beta_s$ sunt toate A_j - producțiile
}

elimină recursivitatea imediată la stânga pt. A_i
}

Fie $G = (N, \Sigma, S, P)$ gr. i.c. $k \geq 1$ dat.

Algoritm iterativ pt. calculul mulțimilor $First_k(x)$, $x \in N \cup \Sigma$

① Inițializare: $First_k(\lambda) = \{\lambda\}$, $First_k(a) = \{a\}$

Pt. fiecare $X \in N$, $X \to z \in P$, $z \in \Sigma^*$

$First_k(x) \stackrel{+}{=} First_k(z)$: dacă $|z| \leq k$, $First_k(z) = \{z\}$
$|z| > k$, $z = z'z''$
$|z'| > k$
$First(z) = \{z'\}$

② Pt. fiecare producție $A \to X_1 X_2 \dots X_m \in P$

$m \geq 1$, $\{x_1, \dots, x_m\} \cap N \neq \emptyset$

Dacă $First_k(x_1) \neq \emptyset, \dots, First_k(x_m) \neq \emptyset$ atunci:

$First_k(A) \stackrel{+}{=} First_k(First_k(x_1) \dots First_k(x_m))$

③ Pt. un șir $\alpha \in (N \cup \Sigma)^*$, $\alpha = x_1 \dots x_m$, $m > 1$

$First_k(\alpha) = First_k(First(x_1) \dots First(x_m))$

Calcul $Follow_k(A)$, $A \in N$

① Inițializare: $Follow_k(S) \leftarrow \{\lambda\}$

② Se fac treceri repetate peste producțiile gramaticii de forma

$A \to \alpha B \beta$, $B \in N$

$Follow_k(B) \stackrel{+}{=} First_k(\beta\, Follow_k(A))$ până când nu se mai adaugă noi șiruri

$Follow_k(A) = \bigcup_{C \to uAv} First_k(v\, Follow_k(C))$

PARSER k predicativ pt. gramatici $LL(k)$ tari

Fie $G = (N, \Sigma, S, P)$ gic., $k \geq 1$ dat.

Notăm cu $G' = (N \cup \{S'\}, \Sigma \cup \{\$\}, S', P \cup \{S' \to S\$\})$

$L(G') = L(G)\#$

<u>Notație</u>: $First'_k$, $Follow'_k$ relativ la G'

$$\Sigma_k = \{ w \in \Sigma^* \mid |w| = k \}$$

$$\Sigma_{\le k} = \{ w \in \Sigma^* \mid |w| \le k \}$$

I. Calculul tabelei de analiză sintactică k - pt. $G' = (N \cup \{S'\}, \Sigma \cup \{\$\}, S', P \cup \{S' \to S\$\})$

$M: (\Sigma_{\le k-1}\$ \cup \Sigma_k) \times (N \cup \Sigma \cup \{\$\}) \to \{accept, error, delete\} \cup \{(\beta, i) \mid i: A \to \beta \in P\}$

Presupunem că prod. lui G sunt numerotate de la $\overline{1, |P|}$

1. Pt. fiecare producție $i: A \to \alpha \in P$, execută 2
2. Pt. fiecare $x \in First'_k(\alpha Follow'_k(A))$
   $M[x, A] \leftarrow (\alpha, i)$
3. $M[\$, \$] = accept$
4. $M[av, a] = delete$, $|v| \le k-1$
5. $M[\cdot, \cdot] = error$

<u>Obs</u>: G este $LL(k)$ tare $\iff$ tabela M nu are intrări multiple.

- <u>Alg. k-predictiv pt. gramatica $G(G')$ bazat pe tabela M.</u>

Fie $w \in \Sigma^*$ șirul analizat

Configurația inițială: $(w, \$, S\$, \lambda)$

Se lucrează cu configurații de forma: (x, α, π), $x \in \Sigma^*\$$
$\alpha \in (N \cup \Sigma)^*\$$
$\pi \in \{1, \dots, |P|\}^*$

Schimbările de configurație:

1) $(x, A\alpha, \pi) \vdash (x, \beta\alpha, \pi i) \iff M[First'_k(x), A] = (\beta, i)$, $i: A \to \beta$
2) $(av, a\alpha, \pi) \vdash (v, \alpha, \pi)$ $\forall a \in \Sigma$
3) $(\$, \$, \pi) \vdash accept$
4) $(x, z\alpha, \pi) \vdash error$, $M[First'_k(x), z] = error$

ex: $E' \to E\$$

1) $E \to TR$

2) $R \to +TR$

3) $ \to *TR$

4) $ \to \lambda$

5) $T \to (E)$

6) $ \to n$

Follow'	I	II
E	$,)	
T	+, *, $	)
R	$	)

$E \to TR$

(1) $Follow'(T) \overset{+}{=} First'(R\,Follow'(E)) \overset{I}{=} \{+, *, \$\}$

(2) $Follow(R) \overset{+}{=} First'(\lambda\,Follow'(E)) \overset{I}{=} Follow'(E) = \{\$\}$

$R \to +TR$

$Follow'(T) \overset{+}{=} First'(R\,Follow'(R)) \overset{I}{=} \{+, *, \$\}$

$Follow'(R) \overset{+}{=} Follow'(R)$

$R \to *TR$ - ca mai sus

$T \to (E)$

$Follow'(E) \overset{+}{=} First'(\,)\,) \cdot Follow'(T) \overset{I}{=} \{)\}$

II: (1) := $\{)\}$
(2) := $\{)\}$

1) $E \to TR$

$First'(TR \cdot Follow'(E)) = \{(, n\}$

2) $E \to +TR$

$First'(+TR \cdot Follow'(R)) = \{+\}$

3) $R \to *TR$

$First'(*TR \cdot Follow'(R)) = \{*\}$

4) $R \to \lambda$

$First'(\lambda \cdot Follow'(R)) = \{\$,)\}$

5) $T \to (E)$

$First'((E) \cdot Follow'(T)) = \{(\}$

6) $T \to n$

$First'(n \cdot Follow'(T)) = \{n\}$

	+	*	(	)	n	$	E	T	R
+	del								(+TR,2)
*		del							(+TR,3)
(			del				(TR,1)	((E),5)	
)				del					(λ,4)
n					del		(TR,1)	(n,6)	
$						acc			(λ,4)

- unde nu e completat ⇒ error

$(n+n\$, E\$, \lambda) \vdash (n+n\$, TR\$, 1) \vdash (n+n\$, nR\$, 16)$
$\vdash (+n\$, R\$, 16) \vdash (+n\$, +TR\$, 162) \vdash (n\$, TR\$, 162)$
$\vdash (n\$, nR\$, 1626) \vdash (\$, R\$, 1626) \vdash (\$, \$, 16264)$

<u>Obs</u>: Sintaxa limbajului PASCAL este formalizată de o gram LL(1).