

5-不带回溯的自上而下分析-LL（1）文法构建

原理

自上而下分析

- ▶ 构造不带回溯的自上而下分析算法
 - ▶ 消除文法的左递归
 - ▶ 提取左公共因子，克服回溯
- ▶ 计算FIRST和FOLLOW集合
- ▶ LL(1)文法的条件

LL(1)文法

► 构造不带回溯的自上而下分析的文法条件

1. 文法不含左递归
2. 对于文法中每一个非终结符A的各个产生式的候选首符集两两不相交。即，若

$$A \rightarrow \alpha_1 | \alpha_2 | \dots | \alpha_n$$

$$\text{则 } \text{FIRST}(\alpha_i) \cap \text{FIRST}(\alpha_j) = \phi \quad (i \neq j)$$

3. 对文法中的每个非终结符A，若它存在某个候选首符集包含 ϵ ，则

$$\text{FIRST}(\alpha_i) \cap \text{FOLLOW}(A) = \phi \\ i=1, 2, \dots, n$$

如果一个文法G满足以上条件，则称该文法G为LL(1)文法。

解读一下

1. 左边必须是终结符，有的可能看的是非终结符，但是化简后一定是终结符开头的
2. 虽然已经是终结符了，但是不做提炼化简，一样开头的候选表达式一样会产生二义性，这一步就是要求你做好最后的提取公因式
3. 好像是已经很全面的定义了，但是考虑到空集，如果他空了，这整句话的first就变成他的下一个表达式了，也就是follow(A)，所以，这个也不能跟已有的first集合重复，如果有重复，继续对他化简

以上三条满足后即可做LL1文法分析

算法

预测分析过程

...X

$a_1 a_2 \dots a_n \dots \#$

$X \rightarrow X_1 X_2 \dots X_k$

► 总控程序根据当前栈顶符号X和输入符号a，执行下列三动作之一：

1. 若 $X = a = \#$ ，则宣布分析成功，停止分析。
2. 若 $X = a \neq \#$ ，则把X从STACK栈顶逐出，让a指向下一个输入符号。✓ 匹配
3. 若X是一个非终结符，则查看分析表M。

► 若 $M[X, a]$ 中存放着关于X的一个产生式，把X逐出STACK栈顶，把产生式的右部符号串按反序——推进STACK栈(若右部符号为 ϵ ，则意味不推什么东西进栈)。

练习题

3.6 a. 计算下面文法的 nullable、FIRST 和 FOLLOW 集合：

$S \rightarrow u B D z$

$B \rightarrow B v$

$B \rightarrow w$

$D \rightarrow E F$

$E \rightarrow y$

$E \rightarrow$

$F \rightarrow x$

$F \rightarrow$

b. 构造 LL(1) 分析表。

c. 给出证据说明该文法不是 LL(1) 文法。

d. 尽可能少地修改该文法使它成为一个接收相同语言的 LL(1) 文法。

答案

No. 3.6

a. The nullable, FIRST, and FOLLOW for this grammar:

	nullable	First	Follow
S	no	u	
B	no	w	v, x, y, z
D	yes	x, y	z
E	yes	y	x, z
F	yes	x	z

问题的思考：参看 P50 页的算法。

b. Construct the LL(1) parsing table

	u	z	v	W	x	y
S	$S \rightarrow uBDz$					
B				$B \rightarrow Bv$ $B \rightarrow w$		
D		$D \rightarrow EF$			$D \rightarrow EF$	$D \rightarrow EF$
E		$E \rightarrow$			$E \rightarrow$	$E \rightarrow y$
F		$F \rightarrow$			$F \rightarrow x$	

问题的思考：参看 P51 页的定义。

c. There are duplicate entries in the LL(1) parsing table, so it's not LL(1) grammar.

问题的思考：参看 P52 页的定义。

d. Left Recursion eliminating

$B \rightarrow Bv$ $B \rightarrow wB'$
 $B \rightarrow w$ $B' \rightarrow vB'$
 $B' \rightarrow$

问题的思考：参看 P52 页的左递归消除方法。