

1.将文法 $G(E)$ 改写为等价的 LL(1)文法，并给出相应的预测分析表。

$G(E)$:

$E \rightarrow [T$

$T \rightarrow F] \mid TE$

$F \rightarrow i \mid Fi$

(清华大学 2000 年硕士生入学考试试题)

解题思路：如果直接将文法消除左递归，会发现改造后的文法不是 LL(1)的，原因在于原文法是一个二义文法，要将其改写为等价的 LL(1)文法，首先需要将其改造成无二义的文法。

解答：该文法是一个二义文法，考虑句子 $[i][i][i]$ 。该文法产生的语言是 $\{([i^n])^m \mid n > 0, m > 0\}$ ，首先将文法改造成无二义文法 $G'(E)$

$E \rightarrow [F]E \mid [F]$

$F \rightarrow i \mid Fi$

消除文法的左递归，得到

$E \rightarrow [F]E \mid [F]$

$F \rightarrow iF'$

$F' \rightarrow iF' \mid \varepsilon$

提取左公因子，得到文法 $G'(E)$:

$E \rightarrow [F]E'$

$E' \rightarrow E \mid \varepsilon$

$F \rightarrow iF'$

$F' \rightarrow iF' \mid \varepsilon$

直接计算它的每个非终结符的 FIRST 集和 FOLLOW 集:

$\text{FIRST}(E) = \{[\}$ $\text{FIRST}(E') = \{[, '\}$

$\text{FIRST}(F) = \{i\}$ $\text{FIRST}(F') = \{i, '\}$

$\text{FOLLOW}(E) = \{\#\}$ $\text{FOLLOW}(E') = \{\#\}$

$\text{FOLLOW}(F) = \{]\}$ $\text{FOLLOW}(F') = \{]\}$

对于改造后的文法：检查文法的所有产生式，我们可以得到：

1. 该文法不含左递归，
2. 该文法中每一个非终结符 E, T, T', F, F' 的各个产生式的候选首符集两两不相交。
3. 该文法的非终结符 T' 和 F' ，它们都有候选式，而且

$\text{FIRST}(E') \cap \text{FOLLOW}(E') = \phi$

$\text{FIRST}(F') \cap \text{FOLLOW}(F') = \phi$

所以该文法是 LL(1)文法。其预测分析表如下：

	i	[]	#
E		$E \rightarrow [F]E'$		
E'		$E' \rightarrow E$		$E' \rightarrow \varepsilon$
F	$F \rightarrow iF'$		$F' \rightarrow \varepsilon$	

F'	F' → iF'			
----	----------	--	--	--

2.有文法 G(S)

$S \rightarrow BA$

$A \rightarrow BS \mid d$

$B \rightarrow aA \mid bS \mid c$

(1) 证明文法 G 是 LL(1)文法;

(2) 构造 LL(1)分析表;

(3) 写出句子 adccd 的分析过程。

(北京航空航天大学 2000 年硕士生入学考试试题)

解题思路：本题除考查 LL(1)分析表(即预测分析表)的构造以外，还考查运用预测分析法分析句子的能力，首先应当掌握预测分析法，并深刻理解在分析过程中符号栈的变化情况。

预测分析算法：

预测分析程序的总控程序在任何时候都是按 STACK 栈顶符号 X 和当前的输入符号 a 行事的,如图 4.4。对于任何 (X, a)，总控程序每次都执行下述三种可能的动作之一：

1. 若 $X = a = \#$ ，则宣布分析成功，停止分析过程。

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。

3. 若 X 是一个非终结符，则查看分析表 M。若 $M[A, a]$ 中存放着关于 X 的一个产生式，那么，首先把 X 逐出 STACK 栈顶，然后，把产生式的右部符号串按反序一一推进 STACK 栈（若右部符号为 ϵ ，则意味不推什么东西进栈）。在把产生式的右部符号推进栈的同时应做这个产生式相应的语义动作（目前暂且不管）。若 $M[A, a]$ 中存放着“出错标志”，则调用出错诊察程序 ERROR。

解答：

(1) 对于文法 G，计算它的每个非终结符的 FIRST 集和 FOLLOW 集：

$\text{FIRST}(S) = \{a, b, c\}$ $\text{FIRST}(A) = \{a, b, c, d\}$

$\text{FIRST}(B) = \{a, b, c\}$

$\text{FOLLOW}(S) = \{a, b, c, d, \#\}$ $\text{FOLLOW}(A) = \{a, b, c, d, \#\}$

$\text{FOLLOW}(B) = \{a, b, c, d\}$

检查文法的所有产生式，我们可以得到：

a. 该文法不含左递归，

b. 该文法中每一个非终结符 S, A, B 的各个产生式的候选首符集两两不相交。

c. 该文法的非终结符都没有有候选式。

所以文法 G 是 LL(1)文法。

(2) 文法 G 的预测分析表如下：

	a	b	c	d	#
S	$S \rightarrow BA$	$S \rightarrow BA$	$S \rightarrow BA$		
A	$A \rightarrow BS$	$A \rightarrow BS$	$A \rightarrow BS$	$A \rightarrow d$	
B	$B \rightarrow aA$	$B \rightarrow bS$	$B \rightarrow c$		

(3) 句子 adccd 的分析过程如下:

步骤	符号栈	输入串	所用产生式
0	#S	adccd#	
1	# AB	adccd#	$S \rightarrow BA$
2	# AAa	adccd#	$B \rightarrow aA$
3	# AA	dccd#	
4	# Ad	dccd#	$A \rightarrow d$
5	# A	ccd#	
6	# SB	ccd#	$A \rightarrow BS$
7	# Sc	ccd#	$B \rightarrow c$
8	# S	cd#	
9	# AB	cd#	$S \rightarrow BA$
10	# Ac	cd#	$B \rightarrow c$
11	# A	d#	
12	# d	d#	$A \rightarrow d$
13	#	#	