中南大学
CENTRAL SOUTH UNIVERSITY

编译原理

COMPILATION PRINCIPLE

中南大学计算机学院

陈志刚　czg@csu.edu.cn

第四章 语法分析-自上而下分析

LL(1)教学案例分享

三、LL(1)分析条件

- 预测分析（LL(1)）法是实现自上而下分析的一种有效方法。它使用一个分析栈和一张分析表。

- 分析表矩阵元素M[A, a]指出非终结符A，面临输入符号a时，应选用的候选式（或产生式）。若A不该面临a，则放一出错标志。

	i	+	*	(	)	#
E	E→TE’			E→TE’		
E’		E’→+TE’			E’→ε	E’→ε
T	T→FT’			T→FT’		
T’		T’→ε	T’→*FT’		T’→ε	T’→ε
F	F→i		F→（E）			

4.3 LL(1)分析法

1、LL(1)分析法的工作过程

开始往输入串末尾和分析栈stack中放“#”，然后把文法开始符号压栈。预测分析程序总是按stack栈顶符号X和当前输入符号a行事。

① 若X=a=“#”，则分析成功，停止分析

② 若X=a≠“#”，则把X从栈顶弹出，a指向下一个输入符号

③ 若$X \in V_n$，则查分析表。

若M[X, a]为某候选式，则弹出X，把该候选式反序压栈；若M[X, a]=ε,则弹出X，什么也不压；若M[X, a]=error，则报错。

4.3 LL(1)分析法

- 为了便于描述分析过程，我们定义：$X \xrightarrow[i]{a} Y$，代表栈X面临输入符号a采取第i条动作后，栈变为Y。
  - 例1．分析i*i#

$$\#E \xrightarrow[3]{i} \#E'T \xrightarrow[3]{i} \#E'T'F \xrightarrow[3]{i} \#E'T'i \xrightarrow[2]{i} \#E'T' \xrightarrow[3]{*} \#E'T'F*$$

$$\xrightarrow[2]{*} \#E'T'F \xrightarrow[3]{i} \#E'T'i \xrightarrow[2]{i} \#E'T' \xrightarrow[3]{\#} \#E' \xrightarrow[3]{\#} \# \xrightarrow[1]{\#} acc$$

4.3 LL(1)分析法

一个学生发起的讨论

学生发现LL(1)分析法的工作过程没有判断栈顶符号X（为终结符时）和当前输入符号a不相等的情况，课堂上通过讲解理解为能匹配才能入栈中，课下该学生找到一个例子，参照LL(1)分析法的工作过程无法进行下去，由此展开了讨论。

以下是学生的推理过程

4.3 LL(1)分析法

例 文法：S→ab|c

- 一个文法是LL(1)的，当且仅当G的每个非终结符A的任何两个侯选式α和β有

①FIRST(α)∩FIRST(β)=ϕ

②若 ε∈FIRST(β)，则有FIRST(α)∩FOLLOW(A)=ϕ

满足条件

说明：S→ab|c是LL(1)文法

4.3 LL(1)分析法

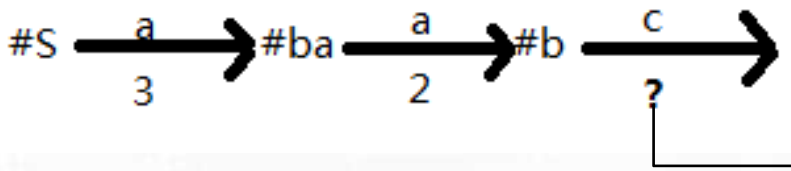
S→ab|c

First(S) = {a,c}　　Follow(S) = {#}

LL(1)分析表：

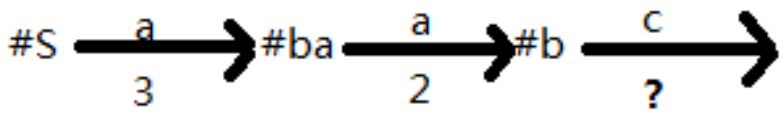
	a	b	c	#
S	ab		c	

现分析符号串ac



LL(1)分析法并没有栈顶符号≠当前输入符号这一步

4.3 LL(1)分析法



开始往输入串末尾和分析栈stack中放“#”，然后把文法开始符号压栈。预测分析程序总是按stack栈顶符号X和当前输入符号a行事。

① 若X=a=”#”，则分析成功，停止分析

② 若X=a≠”#”，则把X从栈顶弹出，a指向下一个输入符号

③ 若X∈Vn，则查分析表。

若M[X, a]为某候选式，则弹出X，把该候选式反序压栈；若M[X, a]=ε,则弹出X，什么也不压；若M[X, a]=error，则报错。

经过师生共同讨论得到如下结果：

增加判断项：若X∈V_T且X≠a，则报错。

4.3 LL(1)分析法

例：G[E]:(1) E ->TE′ (2) E′ ->+TE′
(3) E′ ->ε (4) T ->FT′
(5) T′ ->*FT′ (6) T′ ->ε
(7) F ->(E) (8) F ->a

试判断文法G是不是LL(1)文法。

分析：

E’ -> +TE’ | ε FIRST(+TE’)= {+}
FOLLOW(E′)= {), ＃}

T’ -> *FT’ | ε FIRST(*FT’)= {*}
FOLLOW(T′)= {+,), ＃}

F -> (E) | a FIRST((E))= {(}
FIRST(a)= {a}

所以G[E]是LL（1）的

3、LL(1)分析表的构造

1. 对文法G的每个产生式A→α执行第2步和第3步；
2. 对每个终结符a∈FIRST(α)，把A→α加至M[A,a]中；
3. 若ε∈FIRST(α)，则对任何b∈FOLLOW(A)把A→α加至M[A,b]中，
4. 把所有无定义的M[A,a]标上“出错标志”。

可以证明，一个文法G的预测分析表不含多重入口，当且仅当该文法是LL(1)的。