

编译技术第四章

1、对文法G[S]

$S \rightarrow a \mid \wedge \mid (T)$

$T \rightarrow T, S \mid S$

(1) 给出 $(a, (a, a))$ 和 $((a, a), \wedge, (a)), a$ 的最左推导。

答：

$S \Rightarrow (T) \Rightarrow (T, S) \Rightarrow (S, S) \Rightarrow (a, S) \Rightarrow (a, (T)) \Rightarrow (a, (T, S)) \Rightarrow (a, (S, S)) \Rightarrow (a, (a, S)) \Rightarrow (a, (a, a))$

$S \Rightarrow (T) \Rightarrow (T, S) \Rightarrow (S, S) \Rightarrow ((T), S) \Rightarrow ((T, S), S) \Rightarrow ((T, S, S), S) \Rightarrow ((S, S, S), S) \Rightarrow ((T, S), S, S), S) \Rightarrow (((S, S), S, S), S) \Rightarrow (((a, S), S, S), S) \Rightarrow (((a, a), S, S), S) \Rightarrow (((a, a), \wedge, S), S) \Rightarrow (((a, a), \wedge, (T)), S) \Rightarrow (((a, a), \wedge, (S)), S) \Rightarrow (((a, a), \wedge, (a)), S) \Rightarrow (((a, a), \wedge, (a)), a)$

(2) 对文法G，进行改写，然后对每个非终结符写出不带回溯的递归子程序。

消除左递归改写后：

$S \rightarrow a \mid \wedge \mid (T)$

$T \rightarrow SA$

$A \rightarrow \epsilon, SA \mid \epsilon$

对于每个非终结符不带回溯的递归子程序如下：

P(S) :

```
{
    If ch= 'a' then read(ch);
    Else if ch= '^' then read(ch);
    Else if ch= '(' then
    {
        Read(ch);
        P(T);
        If ch= ')' then read(ch);
        Else error();
    }
    Else error();
}
```

P(T) :

```
{
    If ch in FIRST(S) then
    {
        P(S);
        P(A);
    }
}
```

```

    }
    Else error();
}
P(A):
{
    If ch= ',' then
    {
        Read(ch);
        P(S);
        P(A);
    }
    Else if ch in FOLLOW(A) then return;
    Else error();
}

```

(3) 经改写后的文法是否是LL(1)的?给出它的预测分析表。

答:

各个非终结符的FIRST和FOLLOW集合如下:

$FIRST(S) = \{a, \wedge, \epsilon\}$

$FOLLOW(S) = \{\#, ,,)\}$

$FIRST(T) = \{a, \wedge, \epsilon\}$

$FOLLOW(T) = \{\}$

$FIRST(A) = \{,, \epsilon\}$

$FOLLOW(A) = \{\}$

显然: $SELECT(S \rightarrow a) \cap SELECT(S \rightarrow \wedge) \cap SELECT(S \rightarrow \epsilon) \cap SELECT(S \rightarrow (T)) = \emptyset$

$SELECT(A \rightarrow SA) \cap SELECT(T \rightarrow \epsilon) = \emptyset$

故该文法是LL(1)文法

预测分析表:

	a	\wedge	(,)	#
S	$\rightarrow a$	$\rightarrow A$	$\rightarrow (T)$			
T	$\rightarrow SA$	$\rightarrow SA$	$\rightarrow SA$			
A				$\rightarrow ST$	$\rightarrow \epsilon$	

(4) 给出输入串(a, a)#的分析过程, 并说明该串是否为G 的句子。

答:

步骤	分析栈	剩余输入串	所用产生式
1	#S	(a, a)#	$S \rightarrow (T)$
2	#)T((a, a)#	(匹配
3	#)T	a, a)#	$T \rightarrow SA$
4	#)AS	a, a)#	$S \rightarrow a$
5	#)Aa	a, a)#	a 匹配

6	#)A	, a)#	A→, SA
7	#AS,	, a)#	, 匹配
8	#)AS	a)#	S→a
9	#)Aa	a)#	a 匹配
10	#)A)#	T→ε
11	#))#) 匹配
12	#	#	接受

由分析结果可知：该串是 G 的句子

2、已知文法G[S]：

$S \rightarrow MH \mid a$

$H \rightarrow LSo \mid \epsilon$

$K \rightarrow dML \mid \epsilon$

$L \rightarrow eHf$

$M \rightarrow K \mid bLM$

判断 G 是否是 LL(1) 文法，如果是，构造 LL(1) 分析表。

答：

非终结符的FIRST集合和FOLLOW集合如下：

$FIRST(S) = \{a, d, b, \epsilon, e\}$

$FOLLOW(S) = \{\#, o\}$

$FIRST(M) = \{d, \epsilon, e\}$

$FOLLOW(M) = \{e, \#, o\}$

$FIRST(H) = \{\epsilon, e\}$

$FOLLOW(H) = \{\#, f, o\}$

$FIRST(L) = \{e\}$

$FOLLOW(L) = \{a, d, b, e, o, \#\}$

$FIRST(K) = \{d, \epsilon\}$

$FOLLOW(K) = \{e, \#, o\}$

对相同左部的产生式可知：

$SELECT(S \rightarrow MH) \cap SELECT(S \rightarrow a) = \emptyset$

$SELECT(H \rightarrow LSo) \cap SELECT(H \rightarrow \epsilon) = \emptyset$

$SELECT(K \rightarrow dML) \cap SELECT(K \rightarrow \epsilon) = \emptyset$

$SELECT(M \rightarrow K) \cap SELECT(M \rightarrow bLM) = \emptyset$

所以文法是LL(1)的

预测分析表：

	a	o	d	e	f	b	#
S	a	MH	MH	MH		MH	MH
H		ε		LSo	ε		ε
K		ε	dML	ε			ε
L				eHf			

M		K	K	K		bLM	K
---	--	---	---	---	--	-----	---