

第六次作业参考答案

4.2.1. 文法预测中为什么会产生回溯？仅使用 FIRST 集合可以避免回溯吗？为什么？

即使当非终结符用某个产生式匹配成功，但是这种成功可能只是暂时的，因为没有足够的信息来唯一地确定可能的产生式，所以分析过程就会产生回溯。
不可以。例如对于产生式 $A \Rightarrow \alpha|\beta$, $FIRST(\alpha)$ 与 $FIRST(\beta)$ 交集为空集，但 ϵ 是其中某个 FIRST 集合的元素，不失一般性，假设 $\epsilon \in FIRST(\alpha)$ ，想要避免回溯，则还需要考虑FOLLOW(A)与FIRST(β)的情况。

4.2.2. 考虑下面的文法

lexp \rightarrow atom | list
atom \rightarrow number | identifier
list \rightarrow (lexp-seq)
lexp-seq \rightarrow lexp-seq lexp | lexp

- 1. 消除左递归
- 2. 求该文法的 FIRST 和 FOLLOW 集
- 3. 说明所得的文法是LL(1)文法
- 4. 为所得的文法构造 LL(1) 分析表
- 5. 对于输入(a(b(2))(c))给出LL(1)分析程序的动作

1.

lexp \rightarrow atom | list
atom \rightarrow number | identifier
list \rightarrow (lexp-seq)
lexp-seq \rightarrow lexp lexp-seq'
lexp-seq' \rightarrow lexp lexp-seq' | ϵ

2.

	FIRST	FOLLOW
lexp	number, identifier, (\$,), number, identifier, (
atom	number, identifier	\$,), number, identifier, (
list	(\$,), number, identifier, (
lexp-seq	number, identifier, ()
lexp-seq'	[ϵ , number, identifier, ()

3. （参考龙书P142，答案不唯一）
可以根据 LL(1) 文法的定义来证明。因为对于：

- lexp 为左部的产生式，有 $FIRST(atom) \cap FIRST(list) = \emptyset$;
- atom 为左部的产生式， $FIRST(number) \cap FIRST(identifier) = \emptyset$;
- lexp-seq' 为左部的产生式， $FIRST(lexp\ lexpseq') \cap FIRST(\epsilon) = \emptyset$ ，且 $FIRST(lexp-seq') \cap FOLLOW(lexp-seq') = \emptyset$

所以该文法是 LL(1) 文法

4.

	number	identifier	()	\$
lexp	lexp \rightarrow atom	lexp \rightarrow atom	lexp \rightarrow list		
atom	atom \rightarrow number	atom \rightarrow identifier			
list			list \rightarrow (lexp-seq)		
lesp-seq	lexp-seq \rightarrow lexp lexp-seq'	lexp-seq \rightarrow lexp lexp-seq'	lexp-seq \rightarrow lexp lexp-seq'		
lexp-seq'	lexp-seq' \rightarrow lexp lexp-seq'	lexp-seq' \rightarrow lexp lexp-seq'	lexp-seq' \rightarrow lexp lexp-seq'	lexp-seq' \rightarrow ϵ	

5.

栈	输入	动作
\$ E	(a(b(2))(c))\$	
\$ L	(a(b(2))(c))\$	E \rightarrow L
\$) S((a(b(2))(c))\$	L \rightarrow (S)
\$) S	a(b(2))(c))\$	match
\$) S'E	a(b(2))(c))\$	S \rightarrow E S'
\$) S'A	a(b(2))(c))\$	E \rightarrow A
\$) S'id	a(b(2))(c))\$	A \rightarrow id
\$) S'	(b(2))(c))\$	match
\$) S'E	(b(2))(c))\$	S' \rightarrow E S'
\$) S'L	(b(2))(c))\$	E \rightarrow L
\$) S') S ((b(2))(c))\$	L \rightarrow (S)
\$) S') S	b(2))(c))\$	match
\$) S') S'E	b(2))(c))\$	S \rightarrow E S'
\$) S') S'A	b(2))(c))\$	E \rightarrow A
\$) S') S'id	b(2))(c))\$	A \rightarrow id
\$) S') S'	(2))(c))\$	match
\$) S') S'E	(2))(c))\$	S' \rightarrow E S'
\$) S') S'L	(2))(c))\$	E \rightarrow L
\$) S') S') S ((2))(c))\$	L \rightarrow (S)
\$) S') S') S	2))(c))\$	match
\$) S') S') S'E	2))(c))\$	S \rightarrow E S'
\$) S') S') S'A	2))(c))\$	E \rightarrow A
\$) S') S') S'num	2))(c))\$	A \rightarrow num
\$) S') S') S'))(c))\$	match
\$) S') S')))(c))\$	S' \rightarrow ϵ
\$) S') S')(c))\$	match
\$) S'))(c))\$	S' \rightarrow ϵ
\$) S'	(c))\$	match
\$) S'E	(c))\$	S' \rightarrow E S
\$) S'L	(c))\$	E \rightarrow L
\$) S')S((c))\$	L \rightarrow (S)
\$) S')S	c))\$	match
\$) S')S'E	c))\$	S \rightarrow E S'
\$) S')S'A	c))\$	E \rightarrow A
\$) S')S'id	c))\$	A \rightarrow id
\$) S')S'))\$	match
\$) S')))\$	S' \rightarrow ϵ
\$) S')\$	match
\$))\$	S' \rightarrow ϵ
\$	\$	Match