

编译原理作业 5

练习 4.3.1

(1) 这个文法没有任何左公因子, 提取后仍是原文法.

(2) 不能, 因为有左递归.

(3)

```
1 | rexpr -> rterm + rexpr | rterm
2 | rterm -> rfactor rterm | rfactor
3 | rfactor -> rprimary stars
4 | stars -> * stars | ε
5 | rprimary -> a | b
```

由于该文法的特殊性, 我没有使用标准的消除左递归方法. 我的做法会改变输入串对应的语法树, 但正则表达式的并和连接本就是可结合的, 所以并不会产生问题, 反而更加简洁.

(4) 适用. 没有左递归就能做自顶向下语法分析.

练习 4.3.3

考虑形如

```
1 | if expr then if expr then other else if expr then other else other
```

的一个句型, 它既可以视为

```
1 | if expr
2 | then
3 |     if expr
4 |     then other
5 |     else
6 |         if expr
7 |         then other
8 | else other
```

也可以视为

```
1 | if expr
2 | then
3 |     if expr
4 |     then other
5 |     else
6 |         if expr
7 |         then other
8 |     else other
```

不难验证对应的两颗语法树都可以被题目中的文法生成.

练习 4.4.1

(1)

首先修改文法使其是 LL(1) 的:

```
1 | S -> 0 T
2 | T -> 1 | S 1
```

再给出用到的 FIRST 和 FOLLOW 集合:

FIRST(S) = {0}.

FIRST(T) = {0, 1}.

FIRST(S 1) = {0}.

FOLLOW(S) = {1, \$}.

FOLLOW(T) = {1, \$}.

终结符的 FIRST 集合, 以及首字符是终结符的串的 FIRST 集合没有给出, 它们太显然了, 下同.

最后给出预测分析表:

	0	1	\$
S	S -> 0 T		
T	T -> S 1	T -> 1	

表项为空表示不存在这样的产生式, 输入串不合法. 下同.

(3)

首先修改文法使其是 LL(1) 的:

```
1 | S -> ( S ) S | ε
```

再给出用到的 FIRST 和 FOLLOW 集合:

FIRST(S) = {(, ε}.

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

最后给出预测分析表:

	()	\$
S	S -> (S) S	S -> ε	S -> ε

(5)

为了避免歧义, 我将用分号替代原文法中的逗号.

首先修改文法使其是 LL(1) 的:

```
1 | S -> ( S T ) | a
2 | T -> ; S T | ε
```

再给出用到的 FIRST 和 FOLLOW 集合:

$\text{FIRST}(S) = \{a, (\}$.

$\text{FIRST}(T) = \{;, \epsilon\}$.

$\text{FOLLOW}(S) = \{;,), \$\}$.

$\text{FOLLOW}(T) = \{)\}$.

最后给出预测分析表:

	a	()	;	\$
S	S -> a	S -> (L)			
T			T -> ε	T -> ; S T	