

Homework 5

PB17000297 罗晏宸

October 15 2019

1 Exercise 4.12

文法如下：

$$\begin{aligned} S &\rightarrow (L) \mid a \\ L &\rightarrow L, S \mid S \end{aligned}$$

(a) 写一个翻译方案，它输出每个 a 的嵌套深度。例如，对于句子 $(a, (a, a))$ ，输出的结果是 1 2 2。

(b) 写一个翻译方案，它打印出每个 a 在句子中是第几个字符。例如，当句子是 $(a, (a, (a, a), (a)))$ 时，打印的结果是 2 5 8 10 14。

解

(a) 用继承属性 **depth** 表示嵌套深度，则翻译方案如下：

$$\begin{array}{ll}
 S' \rightarrow & \{ S.\text{depth} = 0; \} \\
 & S \\
 S \rightarrow & \{ L.\text{depth} = S.\text{depth} + 1; \} \\
 & (L) \\
 S \rightarrow a & \{ \text{print}(S.\text{depth}); \} \\
 L \rightarrow & \{ L_1.\text{depth} = L.\text{depth}; \} \\
 & L_1, \\
 & S \\
 L \rightarrow & \{ S.\text{depth} = L.\text{depth}; \} \\
 & S
 \end{array}$$

(b) 用继承属性 **before** 表示句子中在文法符号前的字符，用综合属性 **out** 表示文法符号推出的字符总数，则翻译方案如下：

$$\begin{array}{ll}
 S' \rightarrow & \{ S.\text{before} = 0; \} \\
 & S \\
 S \rightarrow & \{ L.\text{before} = S.\text{before} + 1; \} \\
 & (L) \\
 S \rightarrow a & \{ S.\text{out} = 1; \text{print}(S.\text{before} + 1); \} \\
 L \rightarrow & \{ L_1.\text{before} = L.\text{before}; \} \\
 & L_1, \\
 & S \\
 L \rightarrow & \{ S.\text{before} = L.\text{before}; \} \\
 & S
 \end{array}$$

2 Non-textbook Exercise

(1)

给出习题 4.12(a) 和 (b) 的翻译方案所对应的属性栈代码。

解 为了自底向上计算，必须确定继承属性在属性栈里的位置，为此引入标记非终结符 P 、 Q 和 R 及其继承属性 i 与综合属性 s 。更改后的翻译方案对应的属性栈代码如下：

$S' \rightarrow P$	{ $S.depth = P.s$; }	
S		
$P \rightarrow \varepsilon$	{ $P.s = 0$; }	<code>stack[ntop].val = 0;</code>
$S \rightarrow$	{ $Q.i = S.depth$; }	
$($		
Q	{ $L.depth = Q.s$; }	
$L)$		
$Q \rightarrow \varepsilon$	{ $Q.s = Q.i + 1$; }	<code>stack[ntop].val = stack[top - 1].val + 1;</code>
$S \rightarrow a$	{ <code>print($S.depth$);</code> }	<code>print(stack[top].val);</code>
$L \rightarrow$	{ $L_1.depth = L.depth$; }	
$L_1,$	{ $R.i = L.depth$; }	
R	{ $S.depth = R.s$; }	
S		
$R \rightarrow \varepsilon$	{ $R.s = R.i$; }	<code>stack[ntop].val = stack[top - 2].val;</code>
$L \rightarrow$	{ $S.depth = L.depth$; }	
S		

$S' \rightarrow P$	{ $S.before = P.s$; }	
S		
$P \rightarrow \varepsilon$	{ $P.s = 0$; }	<code>stack[ntop].val = 0;</code>
$S \rightarrow$	{ $Q.i = S.before$; }	
$($		
Q	{ $L.before = Q.s$; }	
$L)$	{ $S.out = L.out + 2$; }	<code>stack[ntop].val = stack[top - 1].val + 2;</code>
$Q \rightarrow \varepsilon$	{ $Q.s = Q.i + 1$; }	<code>stack[ntop].val = stack[top - 1].val + 1;</code>
$S \rightarrow a$	{	
	$S.out = 1$;	<code>stack[ntop].val = 1;</code>
	$print(S.before + 1)$;	<code>print(stack[top].val + 1);</code>
	}	
$L \rightarrow$	{ $L_1.before = L.before$; }	
$L_1,$	{ $R.i = L.before + L_1.out$; }	
R	{ $S.before = R.s$; }	
S	{ $L.out$	
	$= L_1.out + S.out + 1$; }	<code>stack[ntop].val = stack[top - 3].val + stack[top].val + 1;</code>
$R \rightarrow \varepsilon$	{ $R.s = R.i + 1$; }	<code>stack[ntop].val = stack[top - 2].val + stack[top - 1].val + 1;</code>
$L \rightarrow$	{ $S.before = L.before$; }	
S	{ $L.out = S.out$; }	<code>stack[ntop].val = stack[top].val;</code>

(2)

给出习题 4.12(a) 和 (b) 的翻译方案所对应的 YACC 语义代码。

解

(a) 翻译方案对应的语义代码如下：

```

1 Start :P S
2       ;
3 P      :          { $$ = 0; }
4       ;
5 S      : '('Q L')' { $$ = $0; }
6       ;
7 Q      :          { $$ = $-1 + 1; }
8       ;
9 S      :a          { $$ = $0; printf("%d\n", $$); }
10      ;
11 L      :L', 'R S   { $$ = $0; }
12      |S           { $$ = $0; }
13      ;
14 R      :          { $$ = $-2; }
15      ;

```

(b) 翻译方案对应的语义代码如下：

```

1 Start :P S
2       ;
3 P      :          { $$ = 0; }
4       ;
5 S      : '('Q L')' { $$ = $3 + 2; }
6       ;
7 Q      :          { $$ = $-1 + 1; }
8       ;
9 S      :a          { $$ = 1; printf("%d\n", $0 + 1); }
10      ;
11 L      :L', 'R S   { $$ = $1 + $4 + 1; }
12      |S           { $$ = $1; }
13      ;
14 R      :          { $$ = $-2 + $-1 + 1; }
15      ;

```

(3)

针对以下文法

$$\begin{aligned}
 E &\rightarrow E \text{'>'} E \\
 &| E \text{'<'} E \\
 &| \textbf{number}
 \end{aligned}$$

设计语法制导定义，使之能计算诸如 $1 < 2 < 3$ 的表达式值为 True；而计算表达式 $1 < 5 > 3$ 的值也为 True。

解 用综合属性 *left* 和 *right* 表示 *E* 推出的字符序列中最左和最右的数字，综合属性 *bool* 表示 *E* 推出的表达式的布尔值，语法制导的定义如下：

产生式	语义规则
$E' \rightarrow E$	$print(E.bool);$
$E \rightarrow E_1 \text{'>'} E_2$	$E.left = E_1.left; E.right = E_2.right;$ $E_1.bool =$ $(E_1.right > E_2.left \ \&\& \ E_1.bool \ \&\& \ E_2.bool) ? \text{true} : \text{false}$
$E \rightarrow E_1 \text{'<'} E_2$	$E.left = E_1.left; E.right = E_2.right;$ $E_1.bool =$ $(E_1.right < E_2.left \ \&\& \ E_1.bool \ \&\& \ E_2.bool) ? \text{true} : \text{false}$
$E \rightarrow \textbf{number}$	$E.left = \textbf{number.val}; E.right = \textbf{number.val}; E.bool = \text{true};$