



# 编译原理

好好学习!!! 天天向上!!!

任课老师: 谢晓园 邮箱: [xxie@whu.edu.cn](mailto:xxie@whu.edu.cn) 办公室: 计算机学院E301

助教: 黎源 邮箱: [1445660426@qq.com](mailto:1445660426@qq.com)



# P198 5.1.2

扩展图5-4中的SDD，使它可以像图5-1那样处理表达式。

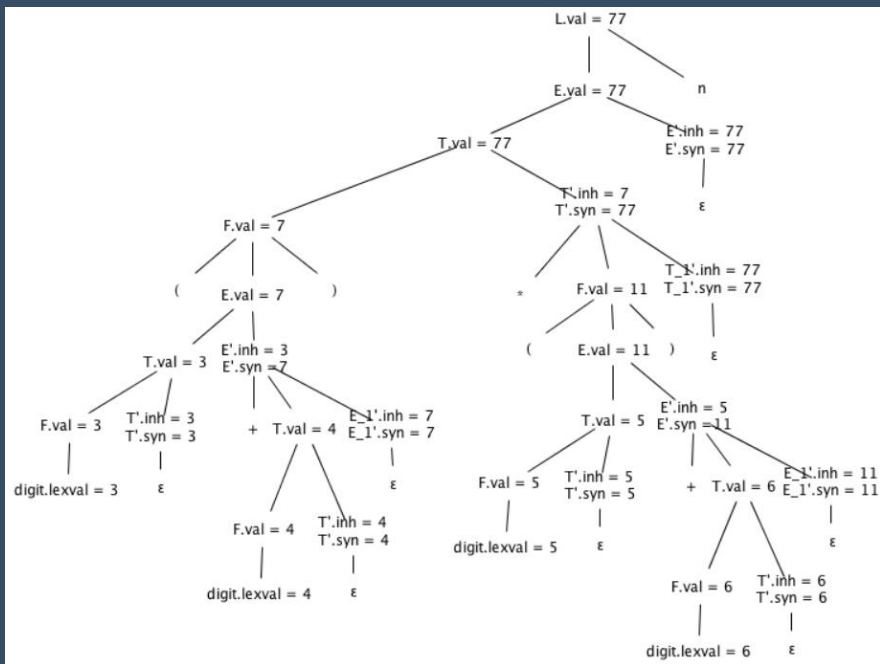
	产生式	语法规则			
			5)	$T \rightarrow FT'$	$T'.inh = F.val$ $T.val = T'.syn$
1)	$L \rightarrow En$	$L.val = E.val$			
			6)	$T' \rightarrow *FT_1'$	$T_1'.inh = T'.inh * F.val$ $T'.syn = T_1'.syn$
2)	$E \rightarrow TE'$	$E'.inh = T.val$ $E.val = E'.syn$			
			7)	$T' \rightarrow \varepsilon$	$T'.syn = T'.inh$
3)	$E' \rightarrow +TE_1'$	$E_1'.inh = E'.inh + T.val$ $E'.syn = E_1'.syn$			
			8)	$F \rightarrow (E)$	$F.val = E.val$
4)	$E' \rightarrow \varepsilon$	$E'.syn = E'.inh$	9)	$F \rightarrow digit$	$F.val = digit.lexval$



# P198 5.1.3(表达式2)

使用你在练习5.1.2中得到的SDD，重复练习5.1.1

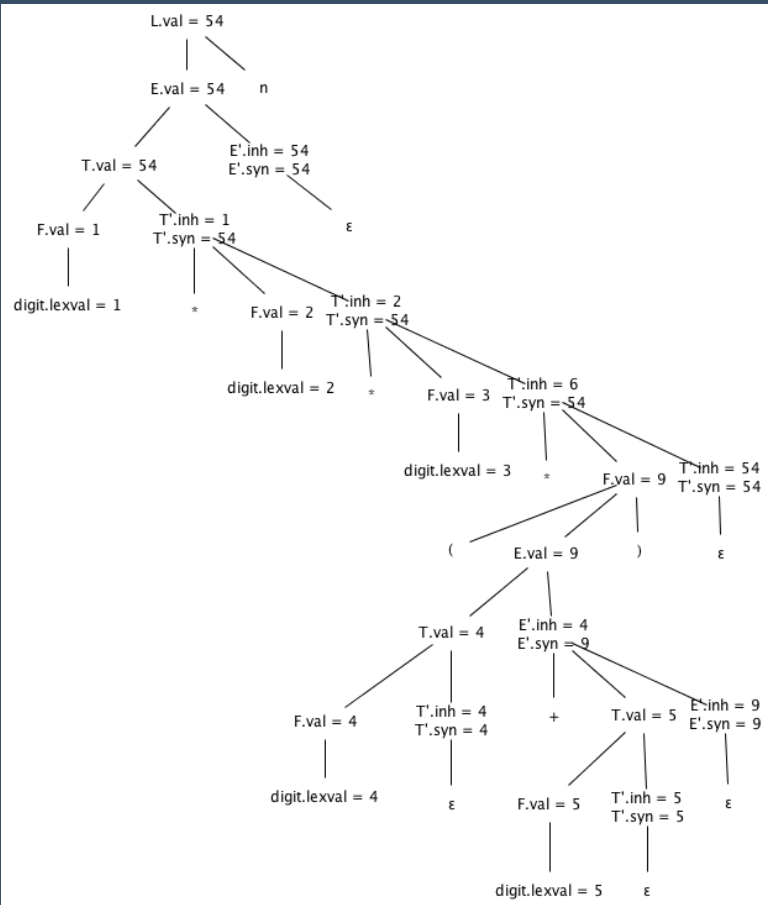
1)  $(3+4)*(5+6)n$  (拓展题)





# P198 5.1.3

2)  $1 * 2 * 3 * (4 + 5)n$





# P203 5.2.4

这个文法生成了含“小数点”的二进制数：

$$S \rightarrow L . L \mid L$$
$$L \rightarrow L B \mid B$$
$$B \rightarrow 0 \mid 1$$

设计一个 L 属性的 SDD 来计算  $S.val$ , 即输入串的十进制数值。



# P203 5.2.4

	产生式	语法规则
1)	$S \rightarrow L_1.L_2$	$L_1.isLeft = true$ $L_2.isLeft = false$ $S.val = L_1.val + L_2.val$
2)	$S \rightarrow L$	$L.isLeft = true$ $S.val = L.val$
3)	$L \rightarrow L_1B$	$L_1.isLeft = L.isLeft$ $L.len = L_1.len + 1$ $L.val = L.isLeft ? L_1.val * 2 + B.val : L_1.val + B.val * 2^{(-L.len)}$
4)	$L \rightarrow B$	$L.len = 1$ $L.val = L.isLeft ? B.val : B.val/2$
5)	$B \rightarrow 0$	$B.val = 0$
6)	$B \rightarrow 1$	$B.val = 1$

- `isLeft` 为继承属性，表示节点是否在小数点的左边
- `len` 为综合属性，表示节点包含的二进制串的长度
- `val` 为综合属性



# P207 5.3.1

下面是涉及运算符+和整数或浮点运算分量的表达式的文法。区分浮点数的方法是看他有无小数点。

$$E \rightarrow E+T \mid T$$
$$T \rightarrow \text{num.num} \mid \text{num}$$

- 1) 给出一个 SDD 来确定每个项 T 和表达式 E 的类型
- 2) 扩展 1 中得到的 SDD，使得它可以把表达式转换成后缀表达式，使用一个单目运算符 IntToFloat 把一个整数转换成相等的浮点数。



# P207 5.3.1

产生式	语法规则
$E \rightarrow E_1 + T$	$(E_1.type == T.type) ? (E.type = E_1.type) : (E.type = float)$
$E \rightarrow T$	$E.type = T.type$
$T \rightarrow num.num$	$T.type = float$
$T \rightarrow num$	$T.type = int$

产生式	语法规则
$E \rightarrow E_1 + T$	$(E_1.type == T.type) ? (E.type = E_1.type) : (E.type = float)$ <b>if</b> $(E_1.type == int \ \&\& \ T.type == float)$ <b>then</b> $E_1 = intToFloat(E_1)$ <b>else if</b> $(T.type == int \ \&\& \ E_1.type == float)$ <b>then</b> $T = intToFloat(T)$ $E.val = (E_1.val, T.val, +)$
$E \rightarrow T$	$E.type = T.type$ $E.val = T.val$
$T \rightarrow num.num$	$T.type = float$ $T.val = num.num$
$T \rightarrow num$	$T.type = int$ $T.val = num$





文法  $G$  定义为

$$\begin{aligned} T &\rightarrow a[L] \mid a \\ L &\rightarrow L L \mid T \end{aligned}$$

其中 ‘ $a$ ’, ‘ $[$ ’, ‘ $]$ ’ 为终结符,  $T$  和  $L$  是非终结符,  $T$  是文法开始符号  
该文法所生成语句是树结构的前序遍历序列, 现要将该序列转换为后序遍历序列。如 “ $a[b\ c[d\ e[f]]\ g]$ ” 转换为 “ $[b\ [d\ [f]e]c\ g]a$ ”

为此设计以下属性:  $a.lexeme$  为终结符  $a$  所对应词素;  $T.postorder$  和  $L.postorder$  记录  $T$  和  $L$  所表示语法成分对应的后序遍历序列

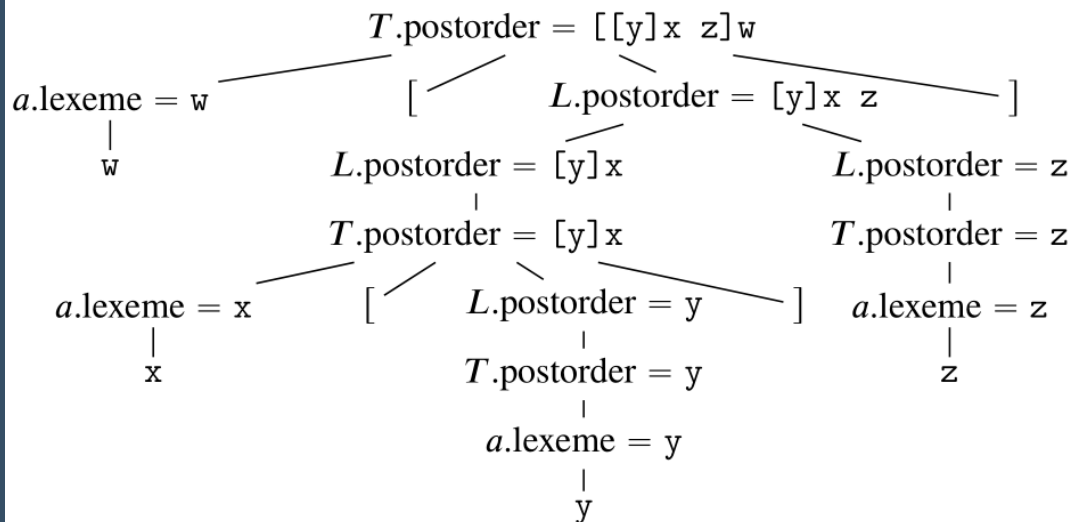
- 1) 试为上述属性设计语法制导定义 SDD
- 2) 试画出语句 “ $w[x[y]z]$ ” 的注释语法树



(1)

产生式	语义规则
$T \rightarrow a[L]$	$T.postorder = "[" + L.postorder + "]" + a.lexeme$
$T \rightarrow a$	$T.postorder = a.lexeme$
$L \rightarrow L_1 L_2$	$L.postorder = L_1.postorder + " " + L_2.postorder$
$L \rightarrow T$	$L.postorder = T.postorder$

(2) “w[x[y]z]” 的注释语法树如下所示:





# THANK YOU

Life is like riding a bicycle. To keep your balance you must keep moving.

——生活就像骑单车，只有不断前进，才能保持平衡。