

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它有无小数点。

$E \rightarrow E + T \mid T$

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

- 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为后缀表达式。

解: (1)

产生式

语义规则

$E \rightarrow E_1 + T$

$E.type = E_1.type == \text{float} \parallel T.type == \text{float} ? \text{float} : \text{int}$

$E \rightarrow T$

$E.type = T.type$

$T \rightarrow \text{num.num}$

$T.type = \text{float}$

$T \rightarrow \text{num}$

$T.type = \text{int}$

(2)

产生式

语义规则

$E \rightarrow E_1 + T$

$E.type = E_1.type == \text{float} \parallel T.type == \text{float} ? \text{float} : \text{int}$   
 $\text{print}(+)$

$E \rightarrow T$

$E.type = T.type$

$T \rightarrow \text{num.num}$

$T.type = \text{float} \quad \text{print}(\text{num.num})$

$T \rightarrow \text{num}$

$T.type = \text{int}, \quad \text{print}(\text{num})$

5.4.2: 改写下面的SDT:

$A \rightarrow A\{a\}B \mid AB\{b\} \mid 0$

$B \rightarrow B\{c\}A \mid BA\{d\} \mid 1$

使得基础文法变成非左递归的。其中a、b、c和d是语义动作，0和1是终结符号

解:  $A \rightarrow 0A' \mid 0A''$

$A' \rightarrow \{a\}BA' \mid \epsilon$

$A'' \rightarrow B\{b\}A'' \mid \epsilon$

$B \rightarrow 1B' \mid 1B''$

$B' \rightarrow \{c\}AB' \mid \epsilon$

$B'' \rightarrow A\{d\}B'' \mid \epsilon$

5.4.6: 修改图5-25中的SDD，使它包含一个综合属性B.le，即一个Box的长度。两个Box并列后得到的Box的长度是这两个Box的长度和。然后，将你的新规则加入到图5-26中SDT的合适位置上

产生式	语义规则
1) $S \rightarrow B$	$B.ps = 10$
2) $B \rightarrow B_1 B_2$	$B_1.ps = B.ps$ $B_2.ps = B.ps$ $B.ht = \max(B_1.ht, B_2.ht)$ $B.dp = \max(B_1.dp, B_2.dp)$
3) $B \rightarrow B_1 \text{ sub } B_2$	$B_1.ps = B.ps$ $B_2.ps = 0.7 \times B.ps$ $B.ht = \max(B_1.ht, B_2.ht - 0.25 \times B.ps)$ $B.dp = \max(B_1.dp, B_2.dp + 0.25 \times B.ps)$
4) $B \rightarrow ( B_1 )$	$B_1.ps = B.ps$ $B.ht = B_1.ht$ $B.dp = B_1.dp$
5) $B \rightarrow \text{text}$	$B.ht = \text{getHt}(B.ps, \text{text.lexval})$ $B.dp = \text{getDp}(B.ps, \text{text.lexval})$

图 5-25 方框排版的 SDD

产生式	语义动作
1) $S \rightarrow B$	{ $B.ps = 10;$ }
2) $B \rightarrow B_1 B_2$	{ $B_1.ps = B.ps;$ } { $B_2.ps = B.ps;$ } { $B.ht = \max(B_1.ht, B_2.ht);$ } { $B.dp = \max(B_1.dp, B_2.dp);$ }
3) $B \rightarrow B_1 \text{ sub } B_2$	{ $B_1.ps = B.ps;$ } { $B_2.ps = 0.7 \times B.ps;$ } { $B.ht = \max(B_1.ht, B_2.ht - 0.25 \times B.ps);$ } { $B.dp = \max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ }
4) $B \rightarrow ( B_1 )$	{ $B_1.ps = B.ps;$ } { $B.ht = B_1.ht;$ } { $B.dp = B_1.dp;$ }
5) $B \rightarrow \text{text}$	{ $B.ht = \text{getHt}(B.ps, \text{text.lexval});$ } { $B.dp = \text{getDp}(B.ps, \text{text.lexval});$ }

图 5-26 方框排版的 SDT

添加：

产生式	语义动作
1) $S \rightarrow B$	{ $B.ps = 10;$ } { $B.le = B.le; \}$
2) $B \rightarrow B_1 B_2$	{ $B_1.ps = B.ps;$ } { $B_2.ps = B.ps;$ } { $B.ht = \max(B_1.ht, B_2.ht);$ } { $B.dp = \max(B_1.dp, B_2.dp);$ } { $B.le = B_1.le + B_2.le; \}$
3) $B \rightarrow B_1 \text{ sub } B_2$	{ $B_1.ps = B.ps;$ } { $B_2.ps = 0.7 \times B.ps;$ } { $B.ht = \max(B_1.ht, B_2.ht - 0.25 \times B.ps);$ } { $B.dp = \max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ } { $B.le = B_1.le + B_2.le \times 0.25 \times B.ps; \}$
4) $B \rightarrow ( B_1 )$	{ $B_1.ps = B.ps;$ } { $B.ht = B_1.ht;$ } { $B.dp = B_1.dp;$ } { $B.le = B_1.le; \}$
5) $B \rightarrow \text{text}$	{ $B.ht = \text{getHt}(B.ps, \text{text.lexval});$ } { $B.dp = \text{getDp}(B.ps, \text{text.lexval});$ } { $B.le = \text{getLe}(B.ps, \text{text.lexval}); \}$

图 5-26 方框排版的 SDT