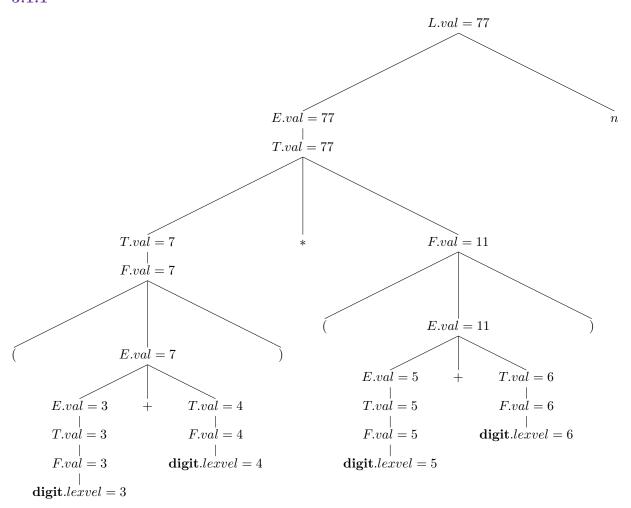
Compiler 5-1&2

孔静 2014K8009929022

5.1.1



5.1.2

产生式	语法规则
$L \to En$	L.val = E.val
E o TE'	E'.inh = T.val
$E \rightarrow IE$	E.val = E'.syn
$E' \rightarrow +TE_1'$	$E_1'.inh = E'.inh + T.val$
$E \rightarrow +I E_1$	$E'.syn = E_1'.syn$
$E' \to \epsilon$	E'.syn = E'.inh
$T \to FT'$	T'.inh = F.val

产生式	语法规则
	T.val = T'.syn
$T' \rightarrow *FT_1'$	$T_1'.inh = T'.inh \times F.val$
$I \rightarrow *FI_1$	$T'.syn = T_1'.syn$
$T' \to \epsilon$	T'.syn = T'.inh
$F \to (E)$	F.val = E.val
$F o \mathbf{digit}$	$F.val = \mathbf{digit}.lexval$

5.2.3

1. 不满足; 满足; 存在;

2. 不满足; 满足; 存在;

3. 满足; 满足; 存在;

4. 不满足; 不满足; 不存在.

5.3.1

1. SSD:

产生式	语法规则
$E \to E_1 + T$	$E.type = (E_1.type == int \&\& T.type == int) ? int : float$
$E \to T$	E.type = T.type
$T ightarrow \mathbf{num.num}$	T.type = float
$T o \mathbf{num}$	T.type = int

2. SSD:

产生式	语法规则
	$E.type = (E_1.type == int \&\& T.type == int) ? int : float$
	$if (E_1.type == T.type)$
$E \rightarrow E_1 + T$	$E.post = E_1.post \ T.post +$
$E \rightarrow E_1 + I$	else if $(E_1.type == int)$
	$E.post = \mathbf{intToFloat}(E_1.post) \ T.post +$
	else $E.post = E_1.post$ intToFloat $(T.post) +$
$E \to T$	E.type = T.type
	E.post = T.post
$T \rightarrow \mathbf{num}$	T.type = int
	$T.post = \mathbf{num}$
$T \rightarrow \mathbf{num.num}$	T.type = float
	$T.post = \mathbf{num.num}$

5.4.2

 $A \rightarrow 0A'$

 $A' \to \{a\}BA'|B\{b\}A'|\epsilon$

 $B \to 1 B'$

 $B' \to \{c\}AB'|A\{d\}B'|\epsilon$

5.4.6

• SSD:

产生式	语法规则
$S \to B$	B.ps = 10
	$B_1.ps = B.ps$
	$B_2.ps = B.ps$
$B \rightarrow B_1 \ B_2$	$B.ht = max(B_1.ht, B_2.ht)$
	$B.dp = max(B_1.dp, B_2.dp)$
	$B.le = B_1.le + B_2.le$
	$B_1.ps = B.ps$
	$B_2.ps = 0.7 \times B.ps$
$B \to B_1 \text{ sub } B_2$	$B.ht = max(B_1.ht, B_2.ht - 0.25 \times B.ps)$
	$B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps)$
	$B.le = B_1.le + 0.7 \times B_2.le$
	$B_1.ps = B.ps$
$D \rightarrow (D)$	$B.ht = B_1.ht$
$B \to (B_1)$	$B.dp = B_1.dp$
	$B.le = B_1.le + getLe(B_1.ps, '(') + getLe(B_1.ps, ')')$
	$B.ht = getHt(B.ps, \mathbf{text}.lexval)$
$B o \mathbf{text}$	$B.dp = getDp(B.ps, \mathbf{text}.lexval))$
	$B.le = getLe(B.ps, \mathbf{text}.lexval)$

• SDT:

产生式	语义动作
$\begin{array}{c} S \to \\ B \end{array}$	$\{B.ps = 10;\}$
	$\{B_1.ps = B.ps;\}$
$B \rightarrow$	$\{B_2.ps = B.ps;\}$
B_1	$B.ht = max(B_1.ht, B_2.ht);$
B_2	$B.dp = max(B_1.dp, B_2.dp);$
	$B.le = B_1.le + B_2.le; $
	$\{B_1.ps = B.ps;\}$
$B \rightarrow$	$\{B_2.ps = 0.7 \times B.ps;\}$
R. cub	$\{B.ht = max(B_1.ht, B_2.ht - 0.25 \times B.ps);$
D_1 sub	$(B.me = max(B_1.me, B_2.me = 0.29 \times B.pe),$
B_1 sub B_2	$B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$
B_2	$B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$
	$B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ $B.le = B_1.le + 0.7 \times B_2.le; \}$
B_2 $B \to $ ($B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ $B.le = B_1.le + 0.7 \times B_2.le; \}$ $\{B_1.ps = B.ps; \}$
B_2	$B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ $B.le = B_1.le + 0.7 \times B_2.le; \}$ $\{B_1.ps = B.ps; \}$ $\{B.ht = B_1.ht;$
B_2 $B \to $ ($B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ $B.le = B_1.le + 0.7 \times B_2.le; \}$ $\{B_1.ps = B.ps; \}$ $\{B.ht = B_1.ht;$ $B.dp = B_1.dp;$
B_2 $B \to $ ($B.dp = max(B_1.dp, B_2.dp + 0.25 \times B.ps);$ $B.le = B_1.le + 0.7 \times B_2.le; \}$ $\{B_1.ps = B.ps; \}$ $\{B.ht = B_1.ht;$ $B.dp = B_1.dp;$ $B.le = B_1.le + getLe(B_1.ps, '('') + getLe(B_1.ps, ')'); \}$