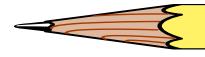
编译原理与技术



期末复习



wenshli@bupt.edu.cn

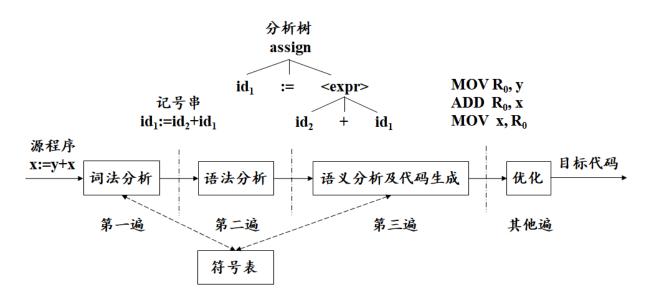


2024年2月16日星期五



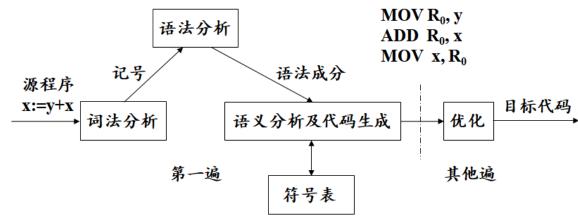
多遍和合并遍的编译程序

1. 多遍编译程序



- □词法分析阶段建立符号表
- □标识符在符号表中的位置作为记号 的属性
- □适用于非块结构语言的编译

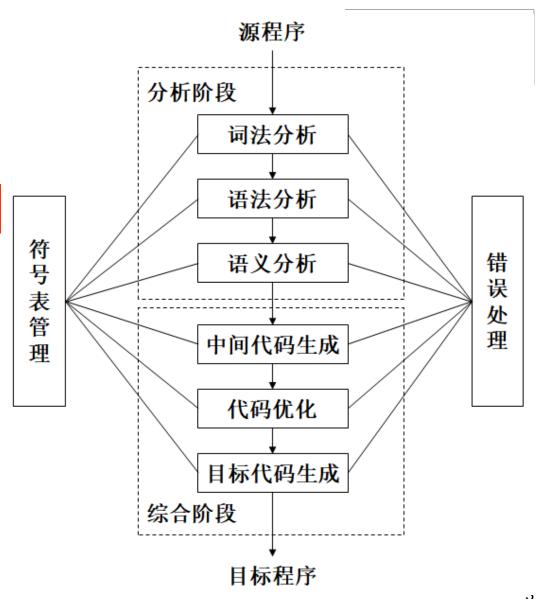
2. 合并遍的编译程序



- □语法分析程序是核心模块
- □当声明语句被识别出来时,标识符和它的属性一起写入符号表中。
- □适用于块结构语言,如C、Pascal 等

课程内容

第1章 编译概述 第2章 形式语言与自动机 (自己复习) 第3章 词法分析 第4章 语法分析 难点 符 第5章 语法制导翻译技术 号 表 重点 第6章 类型检查 管 玾 第7章 运行时刻环境 第8章 中间代码生成 第9章 代码生成 第10章 代码优化

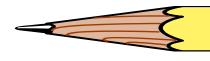


venshli@bupt.edu.cn

编译原理与技术



第5章 语法制导翻译技术



wenshli@bupt.edu.cn



2024年2月16日星期五



venshli@bupt.edu.c

学习任务

- ■作业要求
 - □利用语法制导定义/翻译方案,翻译输入符号串;
 - □对于给定文法,根据翻译目标的要求,设计语法制导定义/翻译方案。

wenshli@bupt.edu.c

语法制导翻译技术考题示例

```
四、(15分) 对文法 G[S]:
S→(L) | a
L→L,S | S
设计了如下语法制导翻译方案
S'→ {S.d=0} S { print(S.x) }
S→({L.d=S.d+1} L) { S.x=L.x }
S→ a { S.x=S.d }
L→{L<sub>1</sub>.d=L.d} L<sub>1</sub>,{S.d=L.d}S { L.x=max(L<sub>1</sub>.x, S.x) }
L→{S.d=L.d} S { L.x= S.x }
```

- (1) 说明方案中的每一个属性是综合属性还是继承属性;
- (2) 说明该翻译方案的翻译目标;
- (3) 对于输入符号串(a, (a, a))

画出其分析树,依赖图,给出翻译结果。

四、(15分)有如下文法 G[S]:

 $S \rightarrow aAbA$

 $A \rightarrow aSb \mid bSa \mid c$

设计一个翻译方案,打印输出每个c在串中的位置。

如: 输入串 acbc, 输出: 2, 4;

输入串 aaacbcbbc, 输出: 4, 6, 9;

要求: 说明翻译方案中使用的属性含义。

四、(15分)考虑如下语法制导定义:

产生式	语义规则
$S' \rightarrow S$	S.p=1
S→(L)	L.p=S.p+1
	S.n=L.n+2
S→a	S.n=1
	print(S.p)
$L\rightarrow L,S$	$L_1.p=L.p$
	$S.p=L.p+L_1.n+1$
	$L.n=L_1.n+S.n+1$
$L\rightarrow S$	S.p=L.p
	L.n=S.n

- (1) 请分析属性 p 和属性 n 是继承属性还是综合属性,并说明其含义。
- (2) 该语法制导定义是 S 属性定义还是 L 属性定义,并说明理由。
- (3) 请将该语法制导定义转换为翻译方案。
- (4)利用该翻译方案对输入符号串 ((a,a),(a),a) 进行翻译,并给出翻译结果。

四、(10分)有如下文法:

 $S \rightarrow (L) \mid a$

 $L \rightarrow L$, $S \mid S$

设计一个翻译方案,使其打印出每个 a 在输入符号串中的位置。例如,对于输入符号串(a,(a,a)),打印输出"2 5 7"。

0. 语法制导定义

■简单算术表达式求值的语法制导定义

产生式	语义规则
L→E	print(E.val)
$E \rightarrow E_1 + T$	E.val=E ₁ .val+T.val
$E \rightarrow T$	E.val=T.val
$T \rightarrow T_1 * F$	T.val=T ₁ .val*F.val
T→F	T.val=F.val
F →(E)	F.val=E.val
F→digit	F.val=digit.lexval

wenshli@bupt.edu.cn

构造分析树

5+3*4

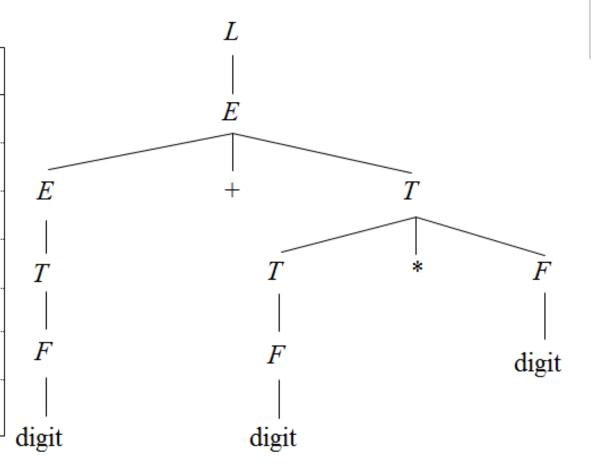
产生式	语义规则
L→E	print(E.val)
$E \rightarrow E_1 + T$	E.val=E ₁ .val+T.val
E→T	E.val=T.val
$T \rightarrow T_1 * F$	T.val=T ₁ .val*F.val
T→F	T.val=F.val
F →(E)	F.val=E.val
F→digit	F.val=digit.lexval

wenshli@bupt.edu.cn

构造分析树(答案)

5+3*4

产生式	语义规则
L→E	print(E.val)
$E \rightarrow E_1 + T$	E.val=E ₁ .val+T.val
$E \rightarrow T$	E.val=T.val
$T \rightarrow T_1 * F$	T.val=T ₁ .val*F.val
T→F	T.val=F.val
F →(E)	F.val=E.val
F→digit	F.val=digit.lexval



构造依赖图

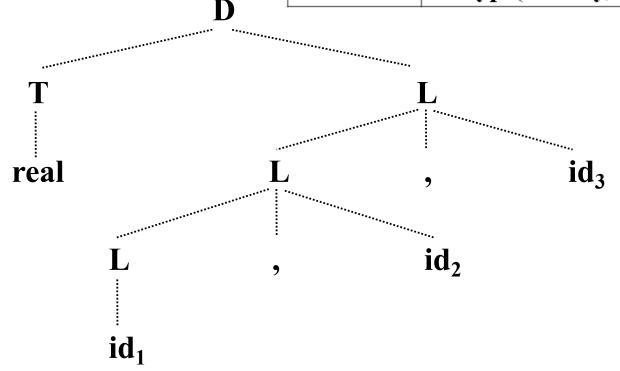
- 依赖图中:
 - □为每个属性设置一个结点
 - □如果属性b依赖于c,那么从 属性c的结点有一条有向边连 到属性b的结点。

real p, q, r real id₁, id₂, id₃

产生式	语义规则
D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real
L→L ₁ ,id	L ₁ .in=L.in; addtype(id.entry, L.in)
L→id	addtype(id.entry, L.in)

产生式 语义规则

 $A \rightarrow XY$ A.a=f(X.x, Y.y)X.i=g(A.a, Y.y)



构造依赖图(答案)

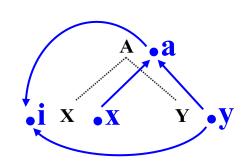
- 依赖图中:
 - □为每个属性设置一个结点
 - □如果属性b依赖于c,那么从 属性c的结点有一条有向边连 到属性b的结点。

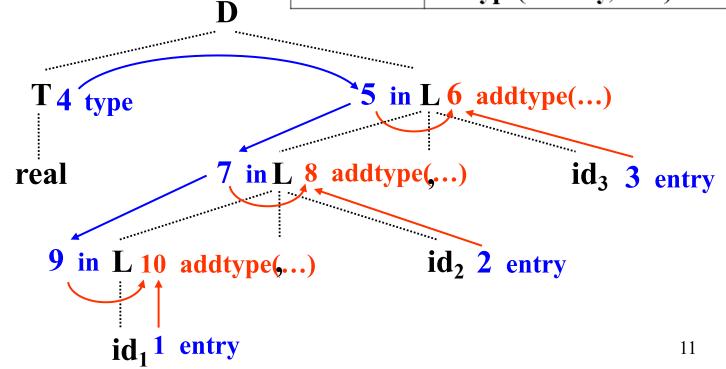
real p, q, r real id₁, id₂, id₃

产生式	语义规则
D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real
L→L ₁ ,id	L ₁ .in=L.in;
	addtype(id.entry, L.in)
L→id	addtype(id.entry, L.in)

产生式 语义规则

 $A \rightarrow XY$ A.a=f(X.x, Y.y)X.i=g(A.a, Y.y)





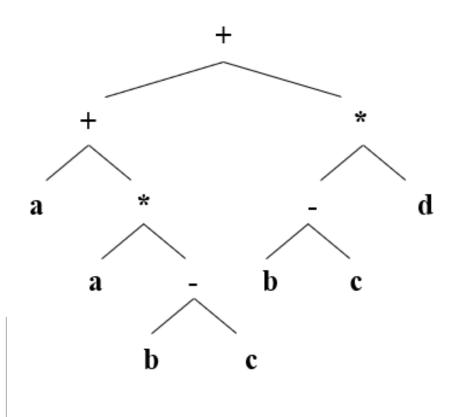
构造语法树

$$a+a*(b-c)+(b-c)*d$$

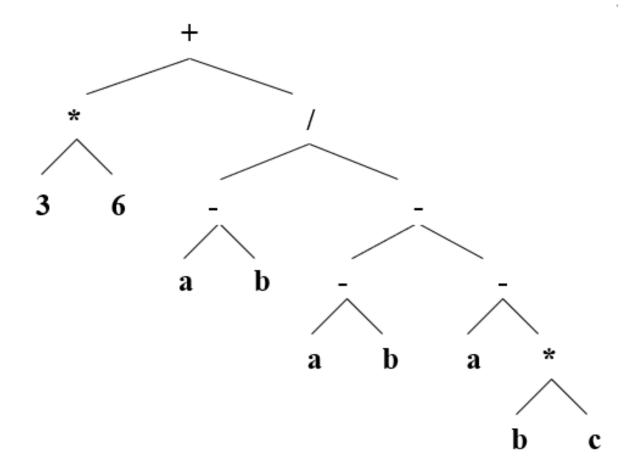
$$3*6+(a-b)/(a-b-(a-b*c))$$

构造语法树(答案)

a+a*(b-c)+(b-c)*d



3*6+(a-b)/(a-b-(a-b*c))



venshli@bupt.edu.c

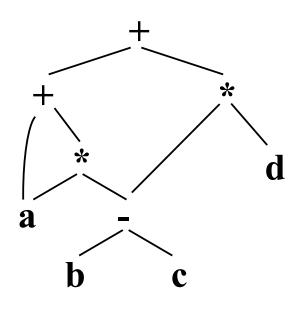
构造表达式的有向非循环图(dag)

为表达式 a+a*(b-c)+(b-c)*d 构造dag

wenshli@bupt.edu.c

构造表达式的有向非循环图(dag)(答案)

为表达式 a+a*(b-c)+(b-c)*d 构造dag

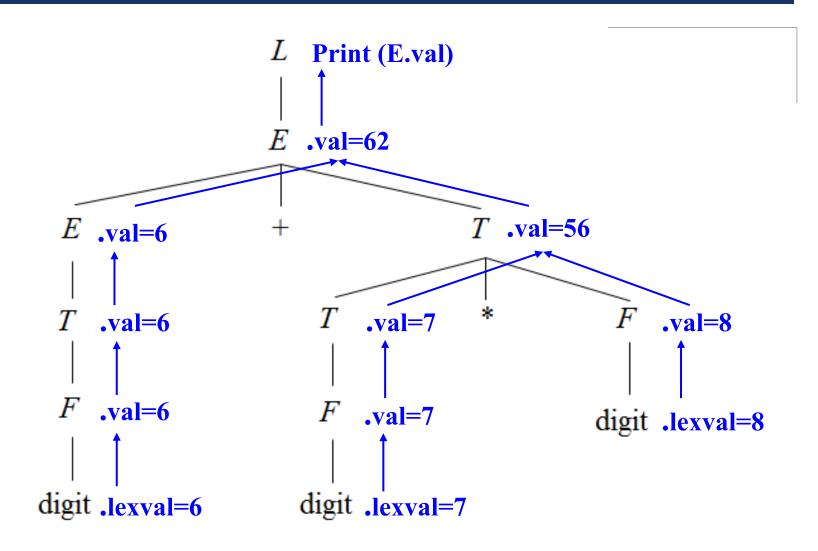


wenshli@bupt.edu.cn

1. 综合属性

6+7*8的分析树加注释的过程

■ 分析树中,如果一个结点的某一属性由其子结点的属性确定,则这种属性为该结点的综合属性。



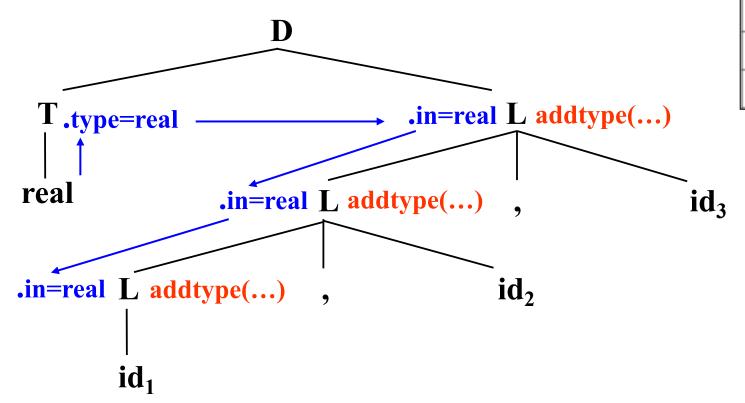
继承属性

用继承属性Lin传递类型信息

- 分析树中,一个结点的继承属性 值由该结点的父结点和/或它的 兄弟结点的属性值决定。
- 句子: int a, b real x, y
- T.type: 由类型关键字确定
- L.in:
 - □ D→TL L.in继承T的类型信息
 - □ $L\rightarrow L_1$, id L_1 .in继承L的类型信息

产生式	语义规则
D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real
L→L ₁ ,id	L ₁ .in=L.in; addtype(id.entry, L.in)
L→id	addtype(id.entry, L.in)

语句 real id₁, id₂, id₃ 的注释分析树



D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real

把类型信息在分析树中向下传递

把类型信息填入符号表中 标识符记录中

wenshli@bupt.edu.c

2. S属性定义和L属性定义

- S属性定义: 仅涉及综合属性的语法制导定义
- L属性定义:一个语法制导定义是L属性定义,如果与每个产生式 $A \rightarrow X_1 X_2 ... X_n$ 相应的每条语义规则计算的属性都是:
 - □ A的综合属性, 或是
 - □ X_i (1≤j≤n) 的继承属性,而该继承属性仅依赖于:
 - >A的继承属性:
 - \rightarrow 产生式中 X_j 左边的符号 X_1 、 X_2 、...、 X_{j-1} 的属性。
- 每一个S属性定义都是L属性定义

语法制导定义示例

S属性定义

产生式	语义规则
L→E	print(E.val)
$E \rightarrow E_1 + T$	E.val=E ₁ .val+T.val
$E \rightarrow T$	E.val=T.val
$T \rightarrow T_1 * F$	T.val=T ₁ .val*F.val
T→F	T.val=F.val
F →(E)	F.val=E.val
F→digit	F.val=digit.lexval

L属性定义

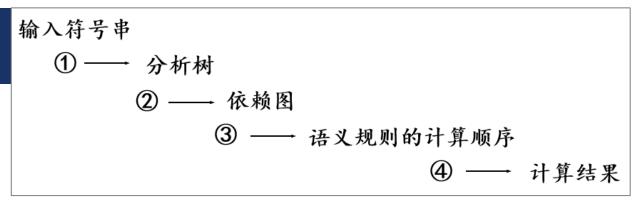
产生式	语义规则
D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real
L→L ₁ ,id	L ₁ .in=L.in;
	addtype(id.entry, L.in)
L→id	addtype(id.entry, L.in)

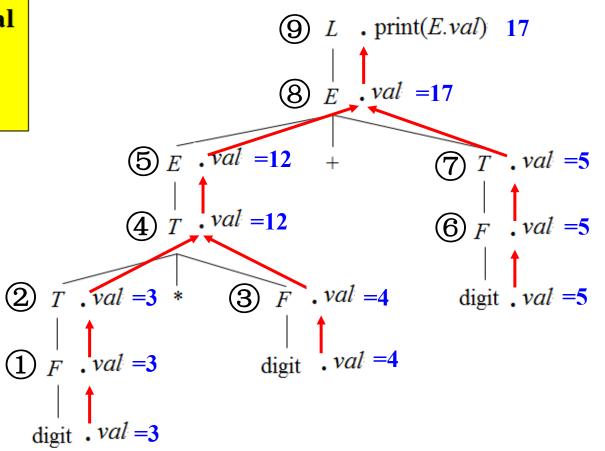
非L属性定义

语义规则
L.i=l(A.i)
M.i=m(L.s)
A.s=f(M.s)
R.i=r(A.i)
Q.i=q(R.s)
A.s=f(Q.s)

示例: 翻译 3*4+5

L→E	Print(E.val)
$E \rightarrow E_1 + T$	E.val=E ₁ .val+T.val
$E \rightarrow T$	E.val=T.val
$T \rightarrow T_1 * F$	T.val=T ₁ .val*F.val
T→F	T.val=F.val
$F \rightarrow (E)$	F.val=E.val
F →digit	F.val=digit.val





wenshli@bupt.edu.c

3. 语法制导翻译的整体思路

- ■首先,根据翻译目标来确定每个产生式的语义;
- 其次,根据产生式的含义,分析每个符号的语义;
- 再次,把这些语义以属性的形式附加到相应的文法符号上(即把语义和语言结构联系起来);
- 然后,根据产生式的语义给出符号属性的求值规则(即语义规则), 从而形成语法制导定义。

■ 翻译:

根据语法分析过程中所使用的产生式,执行与之相应的语义规则,完成符号属性值的计算,从而完成翻译。

- 翻译目标: 计算表达式的值
- 分析确定每个产生式的语义;
- 分析确定每个符号的语义;
- 确定文法符号的属性;
- 设计语义规则,形成语法制导定义。

 - □语法制导定义:产生式 语义规则
- 翻译表达式 3*4+5:

 $E \rightarrow E_1 + T$ $E \rightarrow T$ $T \rightarrow T_1 * F$ $T \rightarrow F$

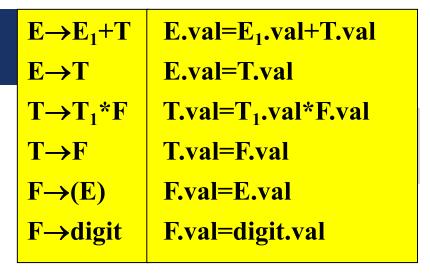
 $F \rightarrow (E)$

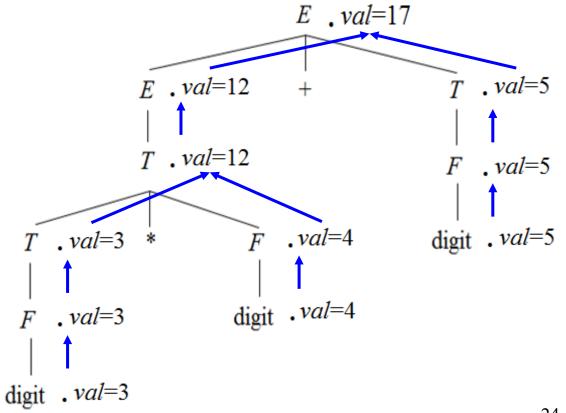
F→**digit**

语法制导翻译示例1(答案)

例如:考虑算术表达式文法

- 翻译目标: 计算表达式的值
- 分析确定每个产生式的语义;
 - $\square E \rightarrow E_1 + T$: 两个子表达式的值相加得到表达式的值
 - □ F→digit: 表达式的值即数字的值
- 分析确定每个符号的语义;
 - \square E, T, F, digit, +, *, (,)
- 确定文法符号的属性;
 - □ E.val、T.val、F.val、digit.val
- 设计语义规则,形成语法制导定义。
 - \square E \rightarrow E₁+T: E.val=E₁.val+T.val
 - □语法制导定义:产生式 语义规则
- 翻译表达式 3*4+5:



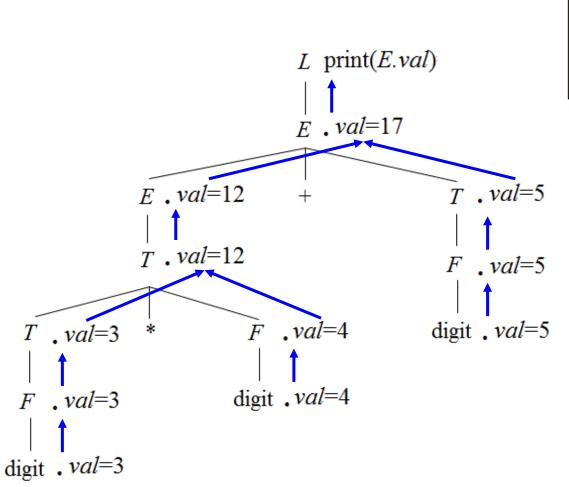


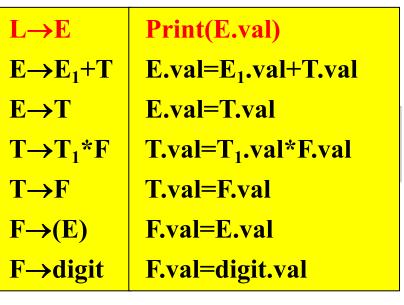
语法制导翻译示例 2

例如:考虑算术表达式文法

■ 翻译目标: 计算并打印表达式的值

- 拓广文法:
 - □ 增加 L→E
 - □ Print(E.val)
- 例如: 3*4+5
 - □分析树:





wenshli@bupt.edu.

翻译方案的设计

■对于S属性定义:

- □为每一个语义规则建立一个包含赋值的动作;
- □把这个动作放在相应的产生式右边 末尾。

例:产生式 语义规则 T→T₁*F T.val=T₁.val*F.val

如下安排产生式和语义动作:

$$T \rightarrow T_1 *F \{ T.val = T_1.val *F.val \}$$

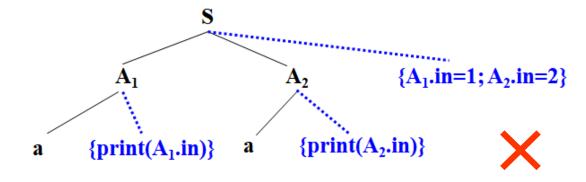
■ 对于L属性定义:

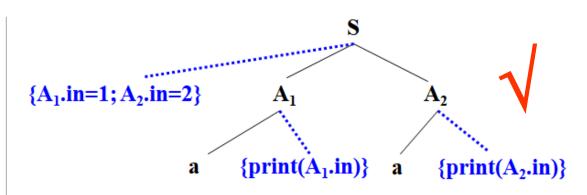
- □动作不能引用其右边文法符号的综 合属性。
- □左部符号的综合属性:
 - > 只有在它所引用的所有属性都计 算出来之后才能计算
 - > 放在产生式右端末尾
- □右部某文法符号的继承属性:
 - >必须在这个符号以前的动作中计 算出来
 - > 放在该文法符号前面

翻译方案设计示例

考虑如下翻译方案:

S
$$\rightarrow$$
A₁A₂ {A₁.in=1; A₂.in=2 }
A \rightarrow a { print(A.in) }





■语法制导定义

产生式	语义规则
D→TL	L.in=T.type
T→int	T.type=integer
T→real	T.type=real
L→L ₁ ,id	L ₁ .in=L.in; addtype(id.entry, L.in)
L→id	addtype(id.entry, L.in)

■翻译方案

```
\begin{array}{l} D \!\!\to\!\! T \; \{ \, L.in \!\!=\!\! T.type \, \} \; L \\ T \!\!\to\!\! int \; \{ \, T.type \!\!=\!\! integer \, \} \\ T \!\!\to\!\! real \; \{ \, T.type \!\!=\!\! real \, \} \\ L \!\!\to\!\! \{ \, L_1.in \!\!=\!\! L.in \, \} \; L_1,id \; \{ \, addtype(id.entry, \, L.in) \, \} \\ L \!\!\to\!\! id \; \{ \, addtype(id.entry, \, L.in) \, \} \end{array}
```

venshli@bupt.edu.c

★期末考试

语法制导翻译技术-例1

有如下文法:

$$S\rightarrow (L) \mid a$$

 $L\rightarrow L, S \mid S$

设计一个翻译方案,打印输出每个a在输入符号串中的位置,并统计输出a的个数。 比如,对于输入符号串(a,(a,a)), 打印输出:

第1个a的位置是2 第2个a的位置是5 第3个a的位置是7 一共有3个a。

★期末考试

语法制导翻译技术-例1(答案)

有如下文法:

$$S \rightarrow (L) \mid a$$

$$L\rightarrow L, S \mid S$$

设计一个翻译方案,打印输出每个a在输入符号串中的位置,并统计输出a的个数。

比如,对于输入符号串(a,(a,a)),

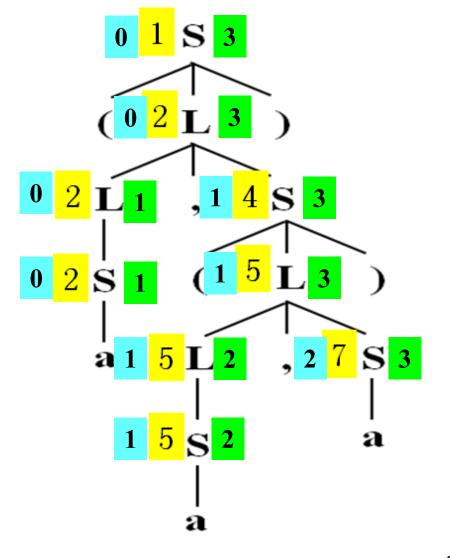
打印输出:

第1个a的位置是2 第2个a的位置是5 第3个a的位置是7 一共有3个a。

pos len

num

tol



★例1(答案)

```
继承属性: pos, S/L推导出的符号串的首字符的位置
           num, 到目前为止已经识别的a的个数
综合属性: len, S/L推导出的字符串的长度
           tol, S/L完全展开后, 字符串中含有a的个数
S' \rightarrow \{ S.pos=1; S.num=0; \} S \{ print("一共有%d个a\n", S.tol); \} \}
S \rightarrow \{\{L.pos=S.pos+1; L.num=S.num;\} L\} \{\{S.len=L.len+2; S.tol=L.tol\}\}
S→a { S.len=1; S.tol=S.num+1; print("第%d个a的位置是%d\n", S.tol, S.pos); }
L \rightarrow \{L_1.pos=L.pos; L_1.num=L.num;\} L_1,
   \{ S.pos=L.pos+L_1.len+1; S.num=L_1.tol \} S \{ L.len=L_1.len+S.len+1; L.tol=S.tol \}
L\rightarrow \{ S.pos=L.pos; S.num=L.num \} S \{ L.len=S.len; L.tol=S.tol \}
```

venshli@bupt.edu.cr

★期末考试 语法制导翻译技术-例2

有如下文法:

 $S \rightarrow (L) \mid a$

 $L\rightarrow L, S \mid S$

设计一个翻译方案,使其打印出每个a在输入符号串中的位置,并统计输出a的个数。

比如,对于输入符号串(a,(a,a)),打印输出:

第1个a的位置是2, 嵌套深度是1

第2个a的位置是5, 嵌套深度是2

第3个a的位置是7, 嵌套深度是2

字符串长度为9,一共有3个a。

enshli@bupt.edu.c

例2(答案)

定义继承属性:

pos,记录S或者L推导出的符号串的首字符的位置deep,记录目前符号串的嵌套深度num,记录到目前为止已经识别的a的个数综合属性:

len,记录S或者L推导出的字符串的长度 tol.记录S或者L完全展开后,字符串中含有a的个数

例2(答案)

```
S' \rightarrow \{ S.pos=1; S.num=0; S.deep=0; \} S
    {print("字符串长度为%d,一共有%d个a\n", S.len, S.tol); }
S \rightarrow \{\{L.pos=S.pos+1; L.num=S.num; L.deep=S.deep+1; \} L \}
    { S.len=L.len+2; S.tol=L.tol }
S\rightarrow a \{ S.len=1; S.tol=S.num+1; \}
       print("第%d个a的位置是%d, 嵌套深度是%d\n", S.tol, S.pos, S.deep); }
L \rightarrow \{L_1.pos=L.pos; L_1.num=L.num; L_1.deep=L.deep;\} L_1,
    \{ S.pos=L.pos+L_1.len+1; S.num=L_1.tol; S.deep=L.deep; \} S
    { L.len=L<sub>1</sub>.len+S.len+1; L.tol=S.tol }
L\rightarrow \{S.pos=L.pos; S.num=L.num, S.deep=L.deep; \} S \{L.len=S.len; L.tol=S.tol\}
```

wenshli@bupt.edu.c

★期末考试 语法制导翻译技术-例3

针对如下文法,设计一个翻译方案,打印出输入二进制数中每个1的权值。

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

如对于输入101.101,

打印输出: 4, 1, 0.5, 0.125。

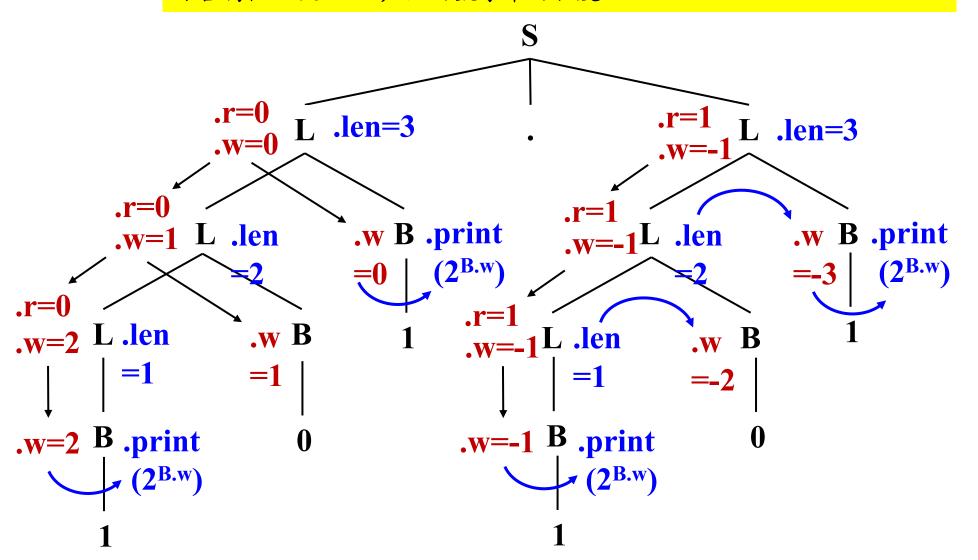
继承属性 L.r: 整数部分 L.r=0, 小数部分 L.r=1

L.w:整数部分L.w表示L产生的末位数字的权位信息,

小数部分 L.w 表示 L产生的首位数字的权位信息

B.w: B产生的数字的权位信息

综合属性 L.len: L产生的数字串的长度



翻译方案

```
S \rightarrow \{ L_1.r=0; L_1.w=0 \} L_1. \{ L_2.r=1; L_2.w=-1 \} L_2
S \rightarrow \{ L.r=0; L.w=0 \} L
L\rightarrow \{L_1.r=L.r;
       if (L.r==0) L_1.w=L.w+1;
       else L_1.w=L.w; L_1
     { if (L.r==0) B.w=L.w;
       else B.w=-(L_1.len+1); } B
     \{ L.len=L_1.len+1 \}
L \rightarrow \{ B.w=L.w; \} B \{ L.len=1 \}
B\rightarrow 1 \{ print(2^{B.w}) \}
B \rightarrow 0
```

