

例题 6.3.1 设下列文法生成变量的类型说明：

$$D \rightarrow \text{id } L$$
$$L \rightarrow, \text{id } L \mid : T$$
$$T \rightarrow \text{integer} \mid \text{real}$$

(1) 构造一个翻译模式，把每个标识符的类型存入符号表；

(2) 由(1)得到的翻译模式，构造一个预测翻译器。

解题思路：这是一个对说明语句进行语义分析的题目，不需要产生代码，但要求把每个标识符的类型填入符号表中。

对给定的适合于自顶向下翻译的翻译模式，第 5.1.4 节给出了设计预测翻译器（或称递归下降翻译器）的方法。按照此方法，不难构造所求的预测翻译器。

解答：

对 D,L,T 设置综合属性 type。过程 addtype(id, type)用来把标识符 id 及其类型 type 填入到符号表中。

(1) 翻译模式如下：

$$D \rightarrow \text{id } L \quad \{\text{addtype}(\text{id.entry}, L.\text{type})\}$$
$$L \rightarrow, \text{id } L1 \quad \{\text{addtype}(\text{id.entry}, L1.\text{type}); L.\text{type} := L1.\text{type};\}$$
$$L \rightarrow : T \quad \{L.\text{type} := T.\text{type}\}$$
$$T \rightarrow \text{integer} \quad \{T.\text{type} := \text{integer}\}$$
$$T \rightarrow \text{real} \quad \{T.\text{type} := \text{real}\}$$

(2) 假设 Ttype 为已定义的表示“类型”的数据结构，预测翻译器如下：

```
procedure D;
```

```
var l_type:Ttype
```

```
begin
```

```
  if sym= "id" then
```

```
    begin
```

```
      advance;
```

```

        l_type:=L;

        addtype(id.entry, l_type)

    end

    else error

end;

procedure L;

    var l_type:Ttype;

begin

    if sym= "," then

        begin

            advance;

            if sym= "id" then

                begin

                    advance;

                    l_type:=L;

                    addtype(id.entry, l_type)

                end

                else error;

            end

        end

    else if sym= ":" then

        begin

            advance;

            l_type:=T;

        end

    end

```

```

    else error;

    return(l_type);

end;

procedure T;

    var t_type:Ttype

begin

    if sym= "integer" then

        begin

            advance;

            t_type:=integer;

        end

    else if sym= "real" then

        begin

            advance;

            t_type:=real;

        end

    else error;

    return(t_type);

end;

```

例题 6.3.2 按照 EQN 语言的功能，给出识别输入并进行格式安放的 L-属性文法。

解题思路：

文法中，非终结符 B（表示盒子）代表一个公式，产生式 $B \rightarrow BB$ 代表两个盒子并置， $B \rightarrow B_1 \text{ sub } B_2$ 代表 B_2 的大小比 B_1 的小，并且放在下角标的位置。继承属性 ps 将影响公式的高度。产生式 $B \rightarrow \text{text}$ 对应的语义规则使得 text 的标准高度乘以 ps 得到 text 的实际高度。

关于 text 的属性 h 通过查表获得，由单词 text 所表示的字符给出。当应用产生式 $B \rightarrow B1B2$ 时， $B1$ 和 $B2$ 通过复写规则继承 ps 的值。综合属性 $B.ht$ 代表 B 的高度，它取 $B1.ht$ 和 $B2.ht$ 的最大值。

当使用产生式 $B \rightarrow B1 \text{ sub } B2$ 时，函数 shrink 使 $B2.ps$ 减少 30%。函数 disp 把盒子 $B2$ 向下放置，并计算 B 的高度。在这里，产生实际排字命令的规则没有给出来。

文法中，唯一的继承属性是非终结符 B 的 ps 属性，每条定义 ps 属性的语义规则只依赖于产生式左边非终结符的继承属性。因此它是一个 L 属性文法。

解答：

所求 L-属性文法如下：

产生式	语义规则
$S \rightarrow B$	$B.ps := 10$ $S.ht := B.ht$
$B \rightarrow B1B2$	$B1.ps := B.ps$ $B2.ps := B.ps$ $B.ht := \max(B1.ht, B2.ht)$
$B \rightarrow B1 \text{ sub } B2$	$B1.ps := B.ps$ $B2.ps := \text{shrink}(B.ps)$ $B.ht := \text{disp}(B1.ht, B2.ht)$
$B \rightarrow \text{text}$	$B.ht := \text{text.h} \times B.ps$