

(若发现问题, 请及时告知)

第 5 讲书面作业包括两部分。第一部分为 Lecture05.pdf 中课后作业题目中的第

3 题。第二部分为以下题目:

A1 以下是某简单语言的一段代码。语言中不包含数据类型的声明, 所有变量的类型默认为整型(假设占用一个存储单元)。语句块的括号为‘begin’和‘end’组合; 赋值号为 ‘:=’, 不等号为‘<>’。每一个过程声明对应一个静态作用域(假定采用多遍扫描机制, 在静态语义检查之前每个作用域中的所有表项均已生成)。该语言支持嵌套的过程声明, 但只能定义无参过程, 且没有返回值。

```
(1)  var a0, b0, a2;
(2)  procedure fun1 ;
(3)      var a1, b1;
(4)      procedure fun2 ;
(5)          var a2;
(6)          begin
(7)              a2 := a1 + b1;
(8)              if(a0 <> b0) then call fun3;
.              ..... /*不含任何 call 语句和声明语句*/
.          end;
.      begin
.          a1 := a0 - b0;
.          b1 := a0 + b0;
(x)      If  a1 < b1  then  call fun2 ;
.          ..... /*不含任何 call 语句和声明语句*/
.      end ;
.  procedure fun3 ;
.      var a3;
.      begin
.          a3 := a0*b0 ;
(y)          if(a2 <> a3) call fun1 ;
.          ..... /*不含任何 call 语句和声明语句*/
.      end ;
.  begin
.      a0 := 1;
.      b0 := 2;
.      a2 := a0/b0 ;
.      call fun3;
.      ..... /*不含任何 call 语句和声明语句*/
```

. end .

若实现该语言时符号表的组织采用多符号表结构，即每个静态作用域均对应一个符号表。试指出：在分析至语句（x）时，当前开作用域有几个？分别包含哪些符号？在分析至语句（y）时，所访问的a2是在哪行语句声明的？

参考解答：

在分析至语句（x）时，当前开作用域有2个；分别包含符号{ a0,b0,a2,fun1, fun3}和{ a1,b1,fun2}。在分析至语句（y）时，所访问的a2是在第（1）行语句声明的。

.....

以下是 Lecture05 文档中的题目

.....

3. PL0编译器的符号表如4.1节所述的那样，采用一个全局的单符号表栈结构。对于下列的PL0程序片断，当PL0编译器在处理到第一个 call p 语句（第 7 行）以及第二个 call p 语句（第 t 行，即过程 q 的第 4 行）时，试分别列出每个开作用域中的符号。

```
(1)  var a,b;
(2)  procedure p ;
(3)      var s;
(4)      procedure r ;
(5)          var v;
(6)          begin
(7)              call p;
.              .....
.              end;
.          begin
.              If a < b then call r ;
.              .....
.          end ;
.  procedure q ;
.      var x,y;
.      begin
(t)          call p ;
.          .....
```

```

.           end ;
.       begin
.           a := 1;
.           b := 2;
.           call q;
.           .....
.       end .

```

### 参考解答:

在处理到第一个 call p 语句（第 7 行）时，每个开作用域中的符号：

主过程作用域中的 a, b, p

过程 p 作用域中的 s, r

过程 p 内过程 r 作用域中的 v

在处理到第二个 call p 语句（第 t 行，即过程 q 的第 4 行）时，每个开作用域中的符号：

主过程作用域中的 a, b, p, q

过程 q 作用域中的 x, y