（若发现问题，请及时告知）

2. 下图左边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 ‘:=’。每一个过程声明对应一个静态作用域。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。过程活动记录中的控制信息包括静态链 SL，动态链 DL，以及返回地址 RA。程序的执行遵循静态作用域规则。下图左边的 PL/0 程序执行到过程 p 被第二次激活时，运行栈的当前状态如下图右半部分所示（栈顶指向单元 26），其中变量的名字用于代表相应的内容。试补齐该运行状态下，单元18、19、21、22、及 23 中的内容。

```
(1)  var a,b;
(2)  procedure  p ;
(3)       var x;
(4)        procedure  r ;
(5)            var x, a;
(6)            begin
(7)                a := 3;
(8)                if   a > b  then call q;
 .                 …… /*仅含符号 x*/
 .            end;
 .       begin
 .            call r ;
 .            …… /*仅含符号 x*/
 .       end ;
 .  procedure  q ;
 .       var x;
 .       begin
(L)            if   a < b  then call p ;
 .             …… /*仅含符号 x*/
 .       end ;
 .  begin
 .       a := 1;
 .       b := 2;
 .       call q;
 .       ……
 .  end .
```

25	x	
24	?	RA
23	18	DL p
22	0	SL
21	x	
20	?	RA
19	13	DL q
18	0	SL
17	a	
16	x	
15	?	RA
14	9	DL r a=3,b=2
13	9	SL
12	x	
11	?	RA
10	5	DL
9	0	SL p a=1,b=2
8	x	
7	?	RA
6	0	DL
5	0	SL q a=1,b=2
4	b	
3	a	
2	?	RA
1	0	DL
0	0	SL

参考解答：

单元18中的内容：0；

单元19中的内容：13；

单元21中的内容：q中x的内容；

单元22中的内容：0；

单元23中的内容：18。

4. 若采取动态作用域规则，该程序的执行效果与之前有何不同？

参考解答：在第二次执行到语句 L 时，若是静态作用域规则，则 a=1, b=2，因此会再次调用 P; 若是动态作用域规则，则 a=3, b=2，因此不会调用 P。

5. 对于下图中的 Decaf/Mind 程序，（1）根据课程实验所采取的运行时存储组织方式，当变量 a 所指向的对象创建后，其对象存储空间中依次存放哪些内容？（2）class Apple 的 vtable 中依次存放哪些内容？

```
class Fruit
{
    int price;
    string name;
    void init(int p, string s)  {price=p; name=s;}
    void print(){ Print(” The price of ", name, " is ",price,"\n");}
}
class Apple extends Fruit
{
    string color;
    void setcolor(string c)  {color=c;}
    void print(){
        Print(“The price of ",color," ",name," is ", price,"\n");
    }
}
class Main {
{
    class Apple a;
    a=New Apple();
    a.setcolor("red");
    a.init(100,"apple");
    a.print();
}
```

参考解答：

（1）a 所指向的对象存储空间中依次存放：指向 Class Apple vtable 的指针，int 变量 price，name 和 color.

（2）class Apple 的 vtable 中包含内容依次为：指向 class Fruit 的 vtable 的指针，指向 class Apple 类名字串的指针，class Fruit 的 init 函数代码入口指针，class Apple 的 print 函数代码人口的指针，以及 class Apple 的 setcolor 函数代码人口的指针。