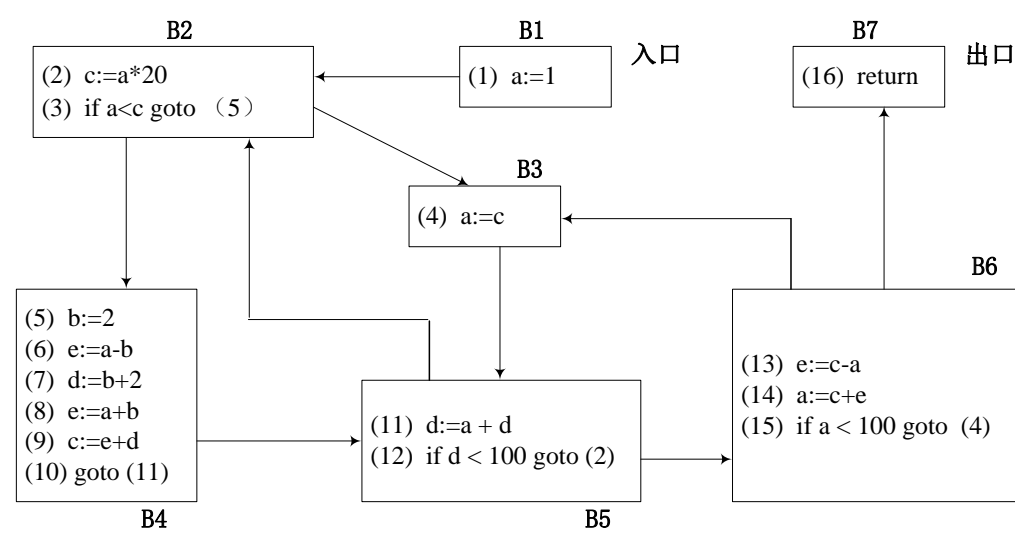


一、下图是包含 7 个基本块的流图，其中 B1 为入口基本块，B7 为出口基本块：



1. 指出在该流图中存在的回边，以及该回边所对应的自然循环（即指出循环中所包含的基本块）。
2. 对于上图所给出的流图，采用迭代求解数据流方程，对**到达-定值**（*reaching definitions*）数据流信息进行分析，迭代结束时 IN 和 OUT 的结果如下图所示。其中，基本块 B4 对应的 GEN、IN 和 OUT 集合未给出，请补齐之。

	GEN	KILL	IN	OUT
<b>B1</b>	{1}	$\emptyset$	$\emptyset$	{1}
<b>B2</b>	{2}	{9}	{1, 2, 4, 5, 8, 9, 11, 13}	{1, 2, 4, 5, 8, 11, 13}
<b>B3</b>	{4}	{1, 14}	{1, 2, 4, 5, 8, 9, 11, 13, 14}	{2, 4, 5, 8, 9, 11, 13}
<b>B4</b>		{2, 11,13}		
<b>B5</b>	{11}	{7}	{1, 2, 4, 5, 7, 8, 9, 11, 13}	{1, 2, 4, 5, 8, 9, 11, 13}
<b>B6</b>	{13, 14}	{1,4,8}	{1, 2, 4, 5, 8, 9, 11, 13}	{2, 5, 9, 11, 13, 14}
<b>B7</b>	$\emptyset$	$\emptyset$	{2, 5, 9, 11, 13, 14}	{2, 5, 9, 11, 13, 14}

3. 对于上图所给出的流图，采用迭代求解数据流方程的方法对**活跃变量信息**进行

分析。假设基本块 B7 的 LiveOut 信息为  $\emptyset$ ，则迭代结束时的结果如下图所示。其中，基本块 B4 的 LiveUse 以及基本块 B5 的 LiveIn 和 LiveOut 信息未给出，请补齐之。

	LiveUse	DEF	LiveIn	LiveOut
B1	$\emptyset$	{a}	{d}	{a, d}
B2	{a}	{c}	{a, d}	{a, c, d}
B3	{c}	{a}	{c, d}	{a, c, d}
B4		{b, c, d, e}	{a}	{a, c, d}
B5	{a, d}	$\emptyset$		
B6	{a, c}	{e}	{a, c, d}	{c, d}
B7	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

4. 请指出在基本块 B4 内第 (7) 条语句之前处的活跃变量信息。
5. 请指出在基本块 B4 内第 (8) 条语句使用变量 a 的 UD 链。
6. 请指出在基本块 B2 内第 (2) 条语句使用变量 c 的 DU 链。
7. 请给出基本块 B4 的 DAG 图。

参考解答：

1. 该流图中存在唯一的回边 B5→B2，该回边所对应的自然循环包含基本块B2，B3，B4，B5和B6。

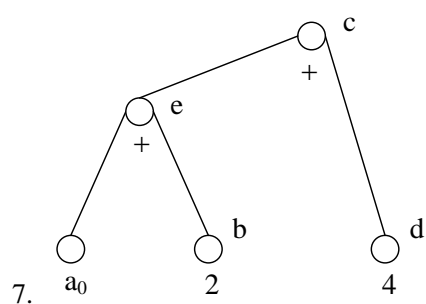
2.

	GEN	KILL	IN	OUT
B1	{1}	$\emptyset$	$\emptyset$	{1}
B2	{2}	{9}	{1, 2, 4, 5, 8, 9, 11, 13}	{1, 2, 4, 5, 8, 11, 13}
B3	{4}	{1, 14}	{1, 2, 4, 5, 8, 9, 11, 13, 14}	{2, 4, 5, 8, 9, 11, 13}
B4	{5, 7, 8, 9}	{2, 11, 13}	{1, 2, 4, 5, 8, 11, 13}	{1, 4, 5, 7, 8, 9}
B5	{11}	{7}	{1, 2, 4, 5, 7, 8, 9, 11, 13}	{1, 2, 4, 5, 8, 9, 11, 13}
B6	{13, 14}	{1, 4, 8}	{1, 2, 4, 5, 8, 9, 11, 13}	{2, 5, 9, 11, 13, 14}
B7	$\emptyset$	$\emptyset$	{2, 5, 9, 11, 13, 14}	{2, 5, 9, 11, 13, 14}

3.

	LiveUse	DEF	LiveIn	LiveOut
<b>B1</b>	$\emptyset$	{a}	{d}	{a,d}
<b>B2</b>	{a}	{c}	{a,d}	{a,c,d}
<b>B3</b>	{c}	{a}	{c,d}	{a,c,d}
<b>B4</b>	{a}	{b,c,d,e}	{a}	{a,c,d}
<b>B5</b>	{a,d}	$\emptyset$	{a,c,d}	{a,c,d}
<b>B6</b>	{a,c}	{e}	{a,c,d}	{c,d}
<b>B7</b>	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

4. 在基本块 B4 内第 (7) 条语句之前处的活跃变量信息为{a, b}。
5. 在基本块 B4 内第 (8) 条语句使用变量 a 的 UD 链为{1, 4}。
6. 在基本块 B2 内第 (2) 条语句使用变量 c 的 DU 链为{3, 4, 13, 14}。



(1)	var a,b;	25		
(2)	procedure p ;	24	x	RA
(3)	var x;	23	?	DL
(4)	procedure r ;	22		SL
(5)	var x, b;	21	x	
(6)	begin	20	?	RA
(7)	b := 0;	19		DL
(8)	call q;	18		SL
.	..... /*仅含符号 x*/	17	b	
.	end;	16	x	
.	begin	15	?	RA
.	call r ;	14	9	DL
.	..... /*仅含符号 x*/	13	9	SL
.	end ;	12	x	
.	procedure q ;	11	?	RA
.	var x;	10	5	DL
.	begin	9	0	SL
(L)	if a < b then call p ;	8	x	
.	..... /*仅含符号 x*/	7	?	RA
.	end ;	6	0	DL
.	begin	5	0	SL
.	a := 1;	4	b	
.	b := 2;	3	a	
.	call q;	2	?	RA
.	.....	1	0	DL
.	end .	0	0	SL

上图左边是某简单语言的一段代码。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。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。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，最外层的作用域编号为第0层，依次类推。我们假设在实现该语言是采用多符号表结构，每个静态作用域对应一个符号表，且通过单独的一遍建立符号表。例如，上图程序中第0层作用域包含符号 a, b, p, q。不同的作用域包含不同的符号，但可以重名，并遵守一般的可见性原则。

1. 假设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（用于返回调用者过程实例的下一条执行语句）。对于上图左边的程序，当过程 p 第二次激活时，运行栈的当前状态如上图右边所示（栈顶为单元26）。试补齐该运行状态下，单元18、19、22、及 23 中的内容。
2. 以上执行遵循静态作用域规则，若是动态作用域规则，指出上图左边的程序第二次执行到第 L 行时，其执行效果有何差异？
- 3.

若按照某种运行时存储组织方式，如下函数 `p` 被激活时的过程活动记录如右图所示。其中 `d` 是动态数组。

```
static int N;

void p( int a) {
    float b;
    float c[10];
    float d[N];
    float e;
    ...
}
```



请给出函数 `p` 中访问 `d[i]` ( $0 \leq i < N$ ) 时相对于活动记录基址的 *Offset* 值：

参考解答：

- 单元 18 中的内容：0

单元 19 中的内容：13

单元 22 中的内容：0

单元 23 中的内容：18
- 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=1, b=0`，因此不会调用 `p`。
- $30+2*i$