

编译原理作业：第七章&第八章

1 第七章

```
int f(int n) {  
    int t, s;  
    if (n < 2) return 1;  
    s = f(n-1);  
    t = f(n-2);  
    return s+t;  
}
```

图 1. 问题1的程序

问题 1. (原书7.2.3,薄书7.2.3)图1中是递归计算Fibonacci数列的C语言代码。假设 f 的活动记录按顺序包含下列元素：

1. 返回值
2. 参数 n
3. 局部变量 s
4. 局部变量 t

通常在活动记录中还会有其它元素。下面的问题假设初始调用是 $f(5)$ 。问：

1. 给出完整的活动树
2. 当第一个 $f(1)$ 调用即将返回时，运行时刻栈和其中的活动记录是什么样子的？

○

```
fun main () {  
    let  
        fun fib0(n) =  
            let  
                fun fib1(n) =  
                    let  
                        fun fib2(n) = fib1(n-1) + fib1(n-2)  
                    in  
                        if n >= 4 then fib2(n)  
                        else fib0(n-1) + fib0(n-2)  
                    end  
                in  
                    if n >= 2 then fib1(n)  
                    else 1  
                end  
            in  
                fib0(4)  
        end;  
end;
```

图 2. 计算Fibonacci数的嵌套函数

问题 2. (原书7.3.1,薄书7.3.1)图2中给出了一个按照非标准方式计算Fibonacci数的ML语言的函数

main。函数fib0计算第 n 个Fibonacci数($n \geq 0$)。嵌套在fib0中的是fib1，它假设 $n \geq 2$ 并计算第 n 个Fibonacci数。嵌套在fib1中的是fib2，它假设 $n \geq 4$ 。请注意，fib1和fib2均不需要检查基本情况。

我们考虑从对main的调用开始，直到对fib0(1)的第一次调用即将返回的时段，请描述出当时的活动记录栈，并给出栈中各个活动记录的访问链。你的活动记录中应该至少包含函数名，参数，访问链这三个信息。

(提示：此题考察对访问链概念的理解，你不需要访问链构造算法也能直接得到答案。一些简单的函数式编程知识可能是有益的)

问题 3. (原书7.4.1, 薄书7.4.1) 假设堆区从0地址开始编址，由数个存储块组成。按照地址顺序，这些存储块的大小分别是80, 30, 60, 50, 70, 20, 40个字节。当我们在一个存储块中放入一个对象时

1. 如果剩余空间大于等于8个字节，那么我们将此对象放在块的高端，剩下的空间仍然是空闲空间
2. 否则，如果剩余空间小于8个字节，那么我们将此对象放在块的低端，并把整个块分配给它

如果我们申请空间的顺序是32, 64, 48, 16，那么满足这些请求之后的空闲空间列表是什么样子？写出每个空闲空间的开始地址和大小（假设起始的七个空闲空间是连续的，即总长350）。假设选择存储块的方法是

1. First-fit
2. Best-fit

。

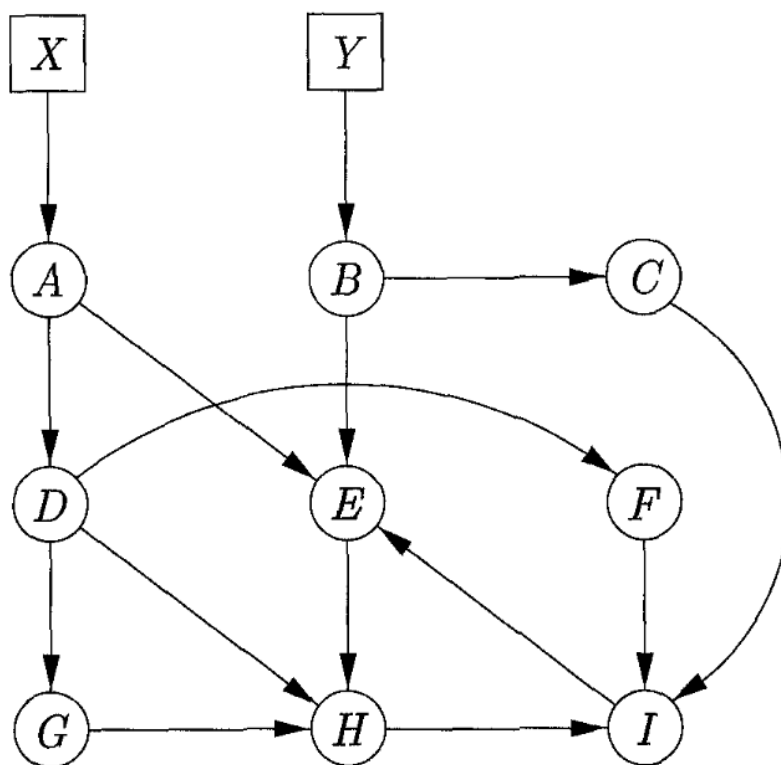


图 3. 一个对象网络

问题 4. (原书7.5.2,薄书7.5.2)对于图3中的对象网络, 在以下情况发生时, 引用计数会发生什么样的变化? 描述引用计数变化的整个过程, 假定初始情况下每个圆圈节点的引用计数等于该节点的入度。

1. 从A到D的指针被删除时
2. 从Y到B的指针被删除时

○

问题 5. (原书7.6.3,薄书7.6.3)当图3中从A到D的指针被删除时, 假定我们执行了一个标记并压缩垃圾回收的过程。同时假设

1. 每个对象的大小是100个字节
2. 在开始时刻, 堆区的9个对象按照字母顺序从堆区的第0个字节开始排列

那么垃圾回收之后各对象的地址是什么? 由于对象大小相同, 你只需要列出对象的顺序。

○

2 第八章

问题 6. (原书8.2.2,薄书8.2.2)假设 a 和 b 是元素占四字节数组, 为下面的三地址语句序列生成机器码

1.

$$\begin{aligned}x &= a[i] \\ y &= b[j] \\ a[i] &= y \\ b[j] &= x\end{aligned}$$

2.

$$\begin{aligned}x &= a[i] \\ y &= b[i] \\ z &= x * y\end{aligned}$$

问题 7. (原书8.2.3,薄书8.2.3)假设 p 和 q 放在内存位置中, 为下列的三地址语句序列生成机器码

$$\begin{aligned}y &= *q \\ q &= q + 4 \\ *p &= y \\ p &= p + 4\end{aligned}$$