算法基础 HW1

PB18111697 王章瀚

2020 年 10 月 7 日

1. (a). 线性查找伪代码如下:

Algorithm 1 线性查找

```
function LIN_FIND(A, v)
    n = len(A)
    for i = 1, 2, ··· n do
        if a_i == v then
            return i
    return NIL
```

使用循环不变式证明算法正确性:

- **初始化**: 每次循环开始前, 必然有$A[1..i-1] \neq v$, 这就是循环不变式
- **保持**: 如果$a_i != v$, 那么可以推出下次迭代有$A[1..i] \neq v$, 即下次迭代也为真
- **终止**: 如果$a_i == v$,那么就得到了正确的i, 直接返回它. 如果循环迭代结束了还没有找到等于v的a_i那么就会返回NIL表示找不到.

(b). 若v在A中第i个, 那就需要查找i个元素, 按题述不需要考虑v不在A的情况. 因此平均需要检查元素个数为:

$$\frac{1}{n}\sum_{i=1}^{n} i = \frac{n+1}{2}$$

最坏情况就是每次都是最后一个元素, 那么需要查找元素个数为: n

因此有

平均情况	$\Theta(\frac{n+1}{2}) = \Theta(n)$
最坏情况	$\Theta(n)$

2. 假定$f(n)$与$g(n)$都是渐进非负函数, 判断下列等式或陈述是否一定是正确的, 并简要解释你的答案.

a $f(n) = O(f(n)^2)$

b $f(n) + g(n) = \Theta(\max(f(n), g(n)))$

c $f(n) + O(f(n)) = \Theta(f(n))$

d if $f(n) = \Omega(g(n))$, then $f(n) = o(g(n))$

a. **错误**. 假设$f(n) = O(f(n)^2)$成立,那么对足够大的n就有$0 \leq f(n) \leq cf(n)^2$, 从而得到$1 \leq cf(n)$, 因此只要构造$f(n)$为单调递减的函数(如$\frac{1}{n}$), 则不能对充分大的n满足$\frac{1}{c} \leq f(n)$.

b. **正确**. 记$\phi(n) = \max(f(n), g(n))$ 只需要证明对于充分大的n, 存在$c_1, c_2 > 0$使得

$$0 \leq c_1\phi(n) \leq f(n) + g(n) \leq c_2\phi(n)$$

事实上, 由于$\phi(n) = \max(f(n), g(n))$, 那么取$c_2 = 2$必然有$f(n) + g(n) \leq c_2\phi(n)$
另一方面, 因为有渐进非负性, 必然有$\phi(n) \leq f(n) + g(n)$, 取$c_1 = 1$即可.

c. **正确**. 对于$O(f(n))$有, 对充分大的n存在$0 \leq O(f(n)) \leq c_0 f(n)$, 从而得到

$$f(n) \leq O(f(n)) + f(n) = \Theta(n) \leq (c_0 + 1)f(n)$$

因此$f(n) + O(f(n)) = \Theta(f(n))$

d. **错误**. 不妨取$f(n) = g(n) = n^2$, 那么显然$f(n) = \Omega(g(n))$, 但$\lim\limits_{n\to\infty} \frac{f(n)}{g(n)} = 1 \neq 0$, 所以$f(n) \neq o(g(n))$.

3. 证明$lg(n!) = \Theta(nlg(n))$(课本3.19式), 并证明$n! = \omega(2^n)$且$n! = o(n^n)$

- **证明$lg(n!) = \Theta(nlg(n))$.**
借助斯特林公式

$$n! = \sqrt{2\pi n}(\frac{n}{e})^n(1 + \Theta(\frac{1}{n}))$$

可以得到对于充分大的n有

$$\sqrt{2\pi n}(\frac{n}{e})^n(1 + c_1\frac{1}{n}) \leq n! \leq \sqrt{2\pi n}(\frac{n}{e})^n(1 + c_2\frac{1}{n})$$

只考虑左边的不等式, 取对数得到

$$\begin{aligned} lg(n!) &\geq nlg(\frac{n}{e}) + lg(\sqrt{2\pi n}(1 + c_1\frac{1}{n})) \\ (\text{第二项显然大于}0) &\geq nlg(\frac{n}{e}) \\ (\text{n充分大时(下面有解释)}) &\geq \frac{1}{e}nlg(n) \end{aligned}$$

这里最后一步很好证明, 两边除以n后, 稍作移项就能得到$lg(n) - 1 \geq \frac{1}{e}lg(n)$, 这对于充分大的$n$是显然成立的.
而又显然有

$$lg(n!) = \sum_{i=1}^{n} lg(i) \leq nlg(n)$$

因此对充分大的n, 我们有

$$\frac{1}{e}nlg(n) \leq lg(n!) \leq nlg(n)$$

因此可以说明$lg(n!) = \Theta(nlg(n))$

- **证明$n! = \omega(2^n)$且$n! = o(n^n)$**
根据刚才得到的不等式, 即对充分大的n有

$$\frac{1}{e}nlg(n) \leq lg(n!) \leq nlg(n)$$

而对n充分大的时候,其不等号严格成立是显然的, 即

$$\frac{1}{e}nlg(n) < lg(n!) < nlg(n)$$

稍作变换得到

$$lg((n^{\frac{1}{e}})^n) < lg(n!) < lg(n^n)$$

当n充分大, 显然有$2 < n^{\frac{1}{e}}$, 因此

$$0 \le 2^n < n! < n^n$$

故$n! = \omega(2^n)$且$n! = o(n^n)$, 证毕!

4. 使用代入法证明$T(n) = T(\lceil n/2 \rceil) + 1$的解为$O(lgn)$

- 首先对于$n = 1$的情况, 等式是不成立的, 不必考虑
- 对于初始情况n=2, 有$0 \le T(2) = T(1) + 1 \le clg(n)$, 这里只需要c充分大即可, 因此初始情况成立
- 假设当$2 \le k < n$时均有$T(k) = T(\lceil k/2 \rceil) + 1$, 那么

$$\begin{aligned} T(n) &= T(\lceil n/2 \rceil) + 1 \\ &\le clg(\lceil n/2 \rceil) + 1 \\ &\le clg(n) + clg\left(\frac{\lceil n/2 \rceil}{n}\right) + 1 \end{aligned}$$

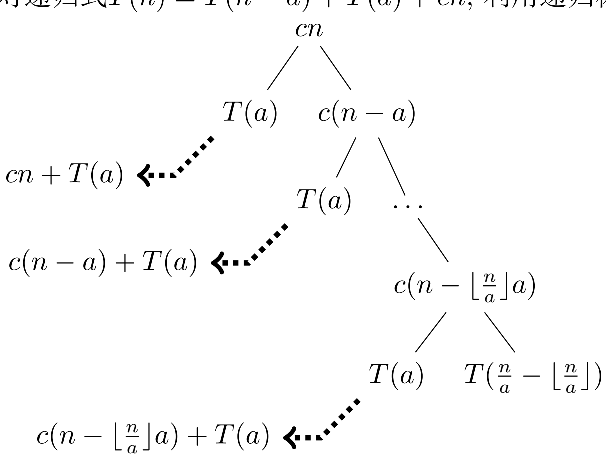
这里$\lceil n/2 \rceil \in [\frac{1}{2}, \frac{1}{2} + \frac{1}{2n})$, 因此对于$n \ge 2$, 必然有$lg\left(\frac{\lceil n/2 \rceil}{n}\right) \le 0$, 这意味着对于充分大的$c$可以有

$$T(n) \le clg(n) + clg\left(\frac{\lceil n/2 \rceil}{n}\right) + 1 \le clg(n)$$

因此当$k = n$时也成立

综合上述几点通过代入法就知: $T(n) = T(\lceil n/2 \rceil) + 1$的解为$O(lgn)$

5. 对递归式$T(n) = T(n-a) + T(a) + cn$, 利用递归树给出一个渐进紧确解, 其中$a \ge 1$和$c > 0$为常数.



从图中可以知道(如果令n为a的倍数, 先去估计一个解)

$$
\begin{aligned}
T(n) &= \Theta(\lfloor\frac{n}{a}\rfloor(cn+T(a)) - \sum_{i=0}^{\lfloor\frac{n}{a}\rfloor} ia + T(\frac{n}{a}-\lfloor\frac{n}{a}\rfloor)) \\
&= \Theta(\lfloor\frac{n}{a}\rfloor(cn+T(a)) - \frac{1}{2}(\lfloor\frac{n}{a}\rfloor)(\lfloor\frac{n}{a}\rfloor+1)a + T(\frac{n}{a}-\lfloor\frac{n}{a}\rfloor)) \\
&= \Theta(\frac{n}{a}(cn+T(a)) - \frac{1}{2}(\frac{n}{a})(\frac{n}{a}+1)a + T(\frac{n}{a}-\frac{n}{a})) \\
&= \Theta(\frac{n^2}{a}((c-\frac{1}{2})) + (\frac{T(a)}{a}-\frac{1}{2})n) \\
&= \Theta(n^2)
\end{aligned}
$$

下面用归纳法证明$T(n)=\Theta(n^2)$.

- 假设当$k<n$的时候都有$T(n)\in[c_1k^2, c_2k^2]$, 那么考虑$k=n$的情况有:

  $$
  \begin{aligned}
  T(n) = T(n-a)+T(a)+cn &\in [c_1(n-a)^2+a^2+cn, c_2(n-a)^2+a^2+cn] \\
  &= [c_1n^2+c_1(a^2-2an)+a^2+cn, c_2n^2+c_2(a^2-2an)+a^2+cn] \\
  &\subset [c_1n^2, c_2n^2](\text{这里取}c_1(a^2-2an)+a^2+cn\geq 0,\ c_2(a^2-2an)+a^2+cn\leq 0)
  \end{aligned}
  $$

  以上当$c_1\leq\frac{a^2+cn}{2an-a^2}\leq c_2$(其中$n\geq\frac{a}{2}$)时成立.

- 再考虑初始情况: 对于合适的n_0, 当$n<n_0$时必然有$T(n)=\Theta(1)$(这是因为在n_0给定的情况下, $T(n)$必然都是有限的数), 因此只需要找到足够小的c_1和足够大的c_2就能够满足$c_1n^2<T(n)<c_2n^2$

至此归纳证明结束.

6. 对下列递归式, 使用主方法求出渐进紧确解:

(a). $T(n)=2T(n/4)+\sqrt{n}$

(b). $T(n)=2T(n/4)+n^2$

(a). 这里a=2, b=4, 从而$n^{log_b a}=n^{log_4 2}=\sqrt{n}$, 显然有$\sqrt{n}=\Theta(\sqrt{n})=\Theta(n^{log_b a})$(自反性)
从而可以使用主定理的第二种情况直接得出

$$T(n)=\Theta(\sqrt{n}lg(n))$$

(b). 这里这里a=2, b=4, 从而$n^{log_b a+\epsilon}=n^{log_4 2+\epsilon}=n^{\frac{1}{2}+\epsilon}$,
因此对于$\epsilon=\frac{3}{2}$有, $f(n)=n^2=\Omega(n^2)=\Omega(n^{log_4 2+\epsilon})$(自反性).
又有, $af(n/b)=2f(n/4)=n^2/8\leq\frac{1}{8}n^2$,
即对充分大的n, 取$c=\frac{1}{8}$, 有$af(n/b)\leq cf(n)$,
故由主定理的第三种情况直接得到,

$$T(n)=\Theta(n^2)$$

7. 主方法能应用于递归式$T(n)=4T(n/2)+n^2\lg(n)$吗? 请说明为什么可以或为什么不可以. 给出这个递归式的一个渐进上界.

- 这里$a=4$, $b=2$, $n^{log_b a}=n^2$. 分别考虑主方法的三种情况:

(1). $n^{log_b a-\epsilon}=n^{2-\epsilon}$,
若要能应用第一种情况, 则要存在c, 使得对充分大的n都有$f(n)=n^2\lg n\le cn^{2-\epsilon}$,
即$c\ge n^{\epsilon}\lg n$, 这显然无法做到.

(2). i. 如果按老师的PPT: 显然有$n^2\lg n=\Theta(n^2\lg n)$, 因此可以应用第二种情况, 直接能够得到

$$T(n)=\Theta(n^2\lg^2 n)$$

ii. 如果按课本: 那么$n^2\lg n!=\Theta(n^2)$, 因此不能使用.

(3). 若要能应用第三种情况, 则要存在c, 使得对充分大的n都有$f(n)=n^2\lg n\ge cn^{2+\epsilon}$,
即$c\le n^{-\epsilon}\lg n$, 这也是做不到的, 因为$\epsilon>0$, 不等式右边是指数衰减的(这里直接描述了, 从数学上也是很好证明的).

综上所述, 如果按老师的PPT来, 就可以; 如果按课本来, 就不行. 原因上面已经说清楚了.

- i. 如果按老师的PPT: 因为刚才已经给出了渐进紧确解, 那么它也是一个渐进上界, 即

$$T(n)=O(n^2\lg^2 n)$$

ii. 如果按课本: 那么需要手动构造一个渐进上界. 我们可以猜测

$$T(n)=O(n^2\lg^2 n)$$

– 假设对于$k<n$, 都有$T(k)<ck^2\lg^2 k$成立. 那么就有

$$\begin{aligned}T(n)&=4T(n/2)+n^2\lg n\\&\le 4c\left(\frac{n}{2}\right)^2\lg^2\frac{n}{2}+n^2\lg n\\&=cn^2\lg^2 n-cn^2\lg 2+n^2\lg n\\&\le cn^2\lg^n-n^2(\lg n-c\lg 2)\\&\le cn^2\lg^2\quad(\text{当}\lg n>c\lg 2\text{即}n>2^c)\end{aligned}$$

– 再考虑初始情况: 对于合适的n_0, 当$n<n_0$时必然有$T(n)=\Theta(1)$(这是因为在n_0给定的情况下, $T(n)$必然都是有限的数), 因此只需要找到足够大的c就能够满足$T(n)<cn^2lg^n$