算法第一次作业

2154312 郑博远

1. 证明 $gcd(m, n) = gcd(n, m \bmod n)$

不妨设$m > n$（若$m < n$, 则经过一次辗转相除m与n倒置）

- 先证 $gcd(m, n) \Rightarrow gcd(n, m \bmod n)$
  记$r = m \bmod n$, 即$m = kn + r(k \in \mathbb{Z}^+)$
  $\forall d|m$ 且 $d|n$, $\frac{r}{d} = \frac{m}{d} - \frac{kn}{d}$ 为整数, 即$d|r$, 得证.
- 再证 $gcd(n, m \bmod n) \Rightarrow gcd(m, n)$
  $\forall d|r$ 且 $d|n$, $\frac{m}{d} = \frac{kn}{d} + \frac{r}{d}$ 为整数, 即$d|m$, 得证.

2. 设计计算$\lfloor \sqrt{n} \rfloor$的算法

```
# 二分法
def sqrt(num):
    l = 0
    r = num + 1
    # 左闭右开区间
    while(r - l > 1):
        mid = (l + r) // 2
        if(mid * mid > num):
            r = mid
        else:
            l = mid
    return l
```

3. 证明主定理

若 $T(n) = aT(\lfloor \frac{n}{b} \rfloor) + O(n^d)$, 则最顶层（记为第0层）用于合并各子问题的时间复杂度为$O(n^d)$, 问题被划分为规模为$\lfloor \frac{n}{b} \rfloor$的$a$个子问题(以下下取整省略不写);
对于第1层，问题的总规模为 $O(\frac{n}{b})$, 用于合并子问题的时间复杂度为 $O((\frac{n}{b})^d)$, 问题被继续划分为规模为$\lfloor \frac{n}{b^2} \rfloor$的

a个子问题;

同理可得, 对于第i层, 将会以 $O((\frac{n}{b^i})^d)$ 的时间复杂度合并子问题;

这样的递归层数共计 $\log_b n$ 层.

故总的时间复杂度为:

$T(n) = n^d((\frac{a}{b^d})^0 + (\frac{a}{b^d})^1 + \ldots + (\frac{a}{b^d})^{log_b n})$

- 当$d = \log_b a$ 时, 即 $T(n) = O(n^d \log n)$.
- 当$d > \log_b a$ 时, $\frac{a}{b^d} < 1$. 因此 $T(n) = n^d \frac{1}{1-\frac{a}{b^d}} = O(n^d)$.
- 当$d < \log_b a$ 时, $T(n) = n^d \frac{1-(\frac{a}{b^d})^{\log_b n}}{1-\frac{a}{b^d}} = O(n^d(1 - n^{\log_b a - d})) = O(n^{\log_b a})$

4. 课本1.6

(以下log均视作以10为底的对数)

1. $f(n) = \Theta(g(n))$

   理由: $\exists c_1 = 1, c_2 = 2, \forall n \geq 10^5, c_1 g(n) \leq f(n) \leq c_2 g(n)$
2. $f(n) = O(g(n))$

   理由: $\exists c = 1, \forall n \geq 0, f(n) \leq cg(n)$
3. $f(n) = \Omega(g(n))$

   理由: $\exists c = 1, \forall n \geq 10, cg(n) \leq f(n)$
4. $f(n) = \Omega(g(n))$

   理由: $\exists c = 1, \forall n \geq 0, cg(n) \leq f(n)$
5. $f(n) = \Theta(g(n))$

   理由: $\exists c_1 = 1, c_2 = 20, \forall n \geq 0, c_1 g(n) \leq f(n) \leq c_2 g(n)$
6. $f(n) = \Omega(g(n))$

   理由: $\exists c = 1, \forall n \geq 10, cg(n) \leq f(n)$
7. $f(n) = \Omega(g(n))$

   理由: $\exists c = 0.01, \forall n \geq 4, cg(n) \leq f(n)$
8. $f(n) = O(g(n))$

   理由: $\exists c = 1, \forall n \geq 1, f(n) \leq cg(n)$

5. 课本1.7

$\lim_{x\to+\infty}\frac{n^n}{n!}=\lim_{x\to+\infty}\frac{n}{n}\cdot\frac{n}{n-1}\cdots\frac{n}{2}\cdot\frac{n}{1}>n=+\infty$

根据无穷大的定义, $\forall c_0>0, \exists n_0>0$, 当$n\geq n_0$, 有$\frac{n^n}{n!}>c_0$.

令$c=\frac{1}{c_0}$, 则 $\forall c>0, \exists n_0>0, n!<cn^n$.

所以根据定义, $n!=o(n^n)$.