

第一周作业

Lecturer: 杨启哲

Last modified: 2025 年 9 月 23 日

截止日期 2025 年 9 月 29 日晚 24: 00

1. 用 True 或者 False 填空:

$f(n)$	$g(n)$	$f = O(g)$	$f = o(g)$	$f = \Omega(g)$	$f = \Theta(g)$	$f = \omega(g)$
$n^3 + 3n$	$100n^2 + 2n + 100$					
$50n + \log n$	$10n + \log \log n$					
$50n^2 \log n$	$10n(\log n)^3$					
$n + \log n$	$\log^{100} n + \log n$					
$n!$	$2^n + n^{100}$					

2. 请找到两个单调递增函数 $f(n)$ 和 $g(n)$, 使得 $f \neq O(g)$ 并且 $g \neq O(f)$ 。

3. 考虑如下算法 COUNT6:

算法: COUNT6

输入: 正整数 n

输出: 第 6 步的执行次数 count

```

1: count ← 0
2: for i ← 1 to  $\lfloor \log n \rfloor$  do
3:   for j ← i to  $i + 5$  do
4:     for k ← 1 to  $i^2$  do
5:       count ← count + 1
6:   end for
7: end for
8: end for

```

(1) 第 5 步执行了多少次?

(2) 该算法的时间复杂性是什么?

4. 考虑如下的一个算法 EUCLID:

算法: EUCLID

输入: 两个正整数 a, b 输出: $\gcd(a, b)$

```

1: repeat
2:   r ← b mod a

```

```
3:     b ← a  
4:     a ← r  
5: until a = 0  
6: return b
```

- (1) 假定 $b \geq a$, 用 b 来表示该算法的运行时间。
- (2) 这个算法的时间复杂性是什么?

Remark 0.1

这个算法实际上就求最大公约数的辗转相除法, 当然我们这里省去了对其正确性的证明。

5. (鸡蛋掉落) 假设现在有一幢 N 层高的楼和一些鸡蛋。对于这些鸡蛋来说, 存在一层楼 T , 使得当这些鸡蛋从 T 层楼或更高的楼层摔落下去时鸡蛋会碎, 反之鸡蛋则不会碎。你现在的目标是在下述条件下设计算法找到这个楼层 T :

- (1) 你只有 1 个鸡蛋, 但有 T 次机会。
- (2) 你有 $\log N$ 个鸡蛋和 $\log N$ 次机会。
- (3) 你有 $\log T$ 个鸡蛋和 $2 \log T$ 次机会。
- (4) 你有 2 个鸡蛋和 $2\sqrt{N}$ 次机会。
- (5) 你有 2 个鸡蛋和 $c\sqrt{T}$ 次机会, 这里 c 是一个与 T, N 无关的常数。

Remark 0.2

这是个很经典的问题, 但我还是希望大家先自己尝试思考一下。