算法设计与分析. 2024秋.   32学时. 2学分

- **教材与参考书**

  1. 算法设计与分析讲义. 何瑞
  2. 算法设计 (Algorithm Design)

     Jon Kleinberg, Eva Tardos 著

     张立昂, 屈婉玲 译

- **内容安排**

  ① 引言: 算法渐近分析、平铺问题、匹配问题

  ② 贪心: 区间调度问题. 图上问题 (MST)

  ③ 分治

  ④ 动态规划

  ⑤ 网络流

  ⑥ 可计算性与计算复杂性

  ⑦ 回溯算法*

  ⑧ 分枝限界算法

  ⑨ 近似算法

  ⑩ 随机算法*

- **成绩评定**
  - 平时: 30%
  - 期末: 70%

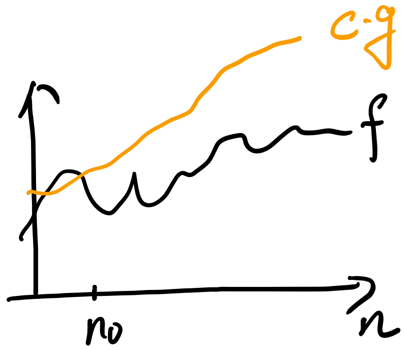
LN1. 算法渐近分析　　$O(\cdot), \Omega(\cdot), \Theta(\cdot)$

1. 上界 $O(\cdot)$:

Def. $f = O(g)$

$\exists n_0, c>0, \forall n \geq n_0,$

$f(n) \leq c \cdot g(n)$



例. $f(n) = 3n^2+6n+5 = O(n^2)$

$f(n) = pn^2+qn+r$, $p, q, r \in N^+$

$\leq \underbrace{(p+q+r)}_{c} n^2 = cn^2$

$\exists c = p+q+r, n_0 = 1, \forall n \geq n_0, f(n) \leq cn^2$

$\therefore f(n) = O(n^2)$

例. $f(n) = pn^2+qn+r \overset{?}{=} O(n^3)$ ✓

例. $f(n) = pn^2+qn+r \overset{?}{=} O(n)$ ×

证: 假设: $\exists c, n_0, n \geq n_0$, $pn^2+qn+r \leq cn$ 不可能成立.

$\exists n_0,$ $n \geq n_0$　$pn^2+qn+r \geq pn^2 > cn$　矛盾

$n \geq n_0 = \lceil \frac{c}{p} \rceil$

$\Rightarrow$ $O(n^2)$ 是 $f(n)$ 的最小上界.

Thm. 多项式定理:

$A(n) = a_m n^m + \cdots + a_1 n + a_0.$　$a_0 \cdots a_m$ 常数.

则 $A(n) = O(n^m)$.

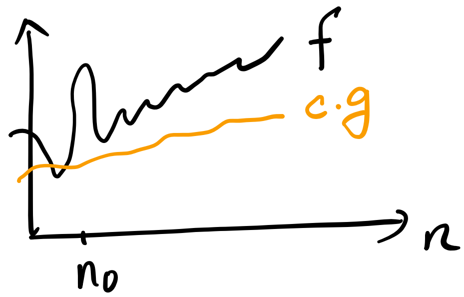
证: 取 $n_0=1$. 找 c. $n \geq n_0$, $A(n) \leq c \cdot n^m$.

$$A(n) \leq |a_m| n^m + \cdots + |a_1| n + |a_0|$$

$$\leq \underbrace{\left(|a_m| + \frac{|a_{m-1}|}{n} + \cdots + \frac{|a_0|}{n^m} \right)}_{c} \cdot n^m$$

$$= c \cdot n^m$$

2. 下界 Ω



Def. $f(n) = \Omega(g(n))$

$\exists\, n_0, c > 0$, $\forall n \geq n_0$,

$f(n) \geq c \cdot g(n)$

例1. $f(n) = pn^2 + qn + r$ $\quad p, q, r \in N^+$

$= \Omega(n^2)$

证: $n_0 = 1$. $f(n) = pn^2 + qn + r \geq pn^2 \geq cn^2$. 取 $c \leq p$ 常数.

Q $\begin{cases} f(n) \overset{?}{=} \Omega(n) & ✓ \\ f(n) \overset{?}{=} \Omega(n^3) & \times \end{cases}$ 反证法.

∴ n^2 是 $f(n)$ 的最大下界

Thm. 多项式定理.

$A(n) = a_m n^m + \cdots + a_1 n + a_0 = \Omega(n^m)$

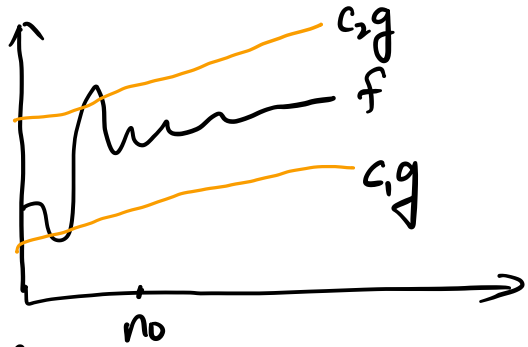
$a_m, \cdots a_0$ 常数. (hw1)

3. Θ: $f(n) = \Theta(g(n))$: $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$

Def. $f(n)=\Theta(g(n))$,

$\exists\, c_1, c_2, n_0$, $\forall n \geq n_0$

$c_1 g(n) \leq f(n) \leq c_2 g(n)$



例1. $f(n)=pn^2+qn+r$. $p.q.r \in N^+$.

$=\Theta(n^2)$

Thm. 多项式定理. $A(n)=\Theta(n^m)$

4. 性质:

① 传递性. f.g.h 函数.

$f=O(g)$, $g=O(h)$. $\Rightarrow f=O(h)$

② 可加性. $f=O(h)$. $g=O(h) \Rightarrow f+g=O(h)$

③ $f.\ g \geq 0$, $g=O(f) \Rightarrow f+g=O(f)$

例: $g=3n^2+4n+5$ $f=n^2$

$f+g=4n^2+4n+5=O(n^2)$

5. 常见的渐近函数:

① 算术级:

$O(1)<O(\log n)<O(n)<O(n\log n)<O(n^2)<\cdots$

折半查找 快速排序 冒泡排序

② 指数级:

$O(2^n) < O(n!) < O(n^n)$ (hw2)

例：SAT可满足性问题：$x_1, x_2 \cdots x_n \in \{0,1\}$

文字 $\begin{cases} x_1, x_2 \cdots x_n \\ \bar{x}_1, \bar{x}_2 \cdots \bar{x}_n \end{cases}$

子句：$C_j = x_1 \vee \bar{x}_2 \vee x_5$

公式 $F = C_1 \wedge C_2 \wedge \cdots \wedge C_m$　　合取范式 CNF

是否存在 $x_1 \cdots x_n$ 的赋值，使 $F = 1$

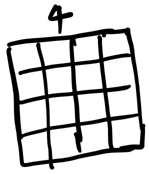
SAT的穷举算法 $O(2^n)$

LN2. 平铺问题 (Tiling 问题)

*问题定义：给定形状 S，给定骨牌 d，问能否用有限多个 d 平铺 S（不重叠的 cover）

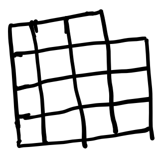
• 例：Dominos：

① S: 4 × 4



✓

②

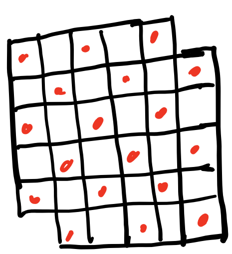


✗

数数：S有奇数个，d是偶数。

$\boxed{\text{周煜杰}}$

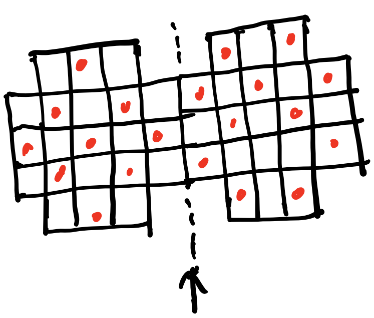
③ 6x6:



红：18，白：16

染色：红格与白格。⇒ 无法平铺。

④ 5X5X2



左：9R，12W

右：$\underbrace{\text{12R, 9W}}$ 需要加 3W

但左边只有2W 相邻

存在鸽子阻碍 (pigeon obstruction)

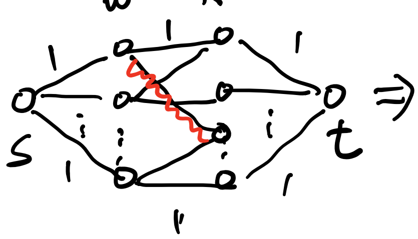
↑ 红块加白块的子集. 其相邻的白/红块少于需要的

Thm. 对于上述的 Domino-Tiling 问题, ∃多项式的确定性算法. 要么返回一个解, 要么返回一个 pigeon obstruction.

Q: 多项式复杂度的算法?

① $|W| \overset{?}{=} |R| = n$

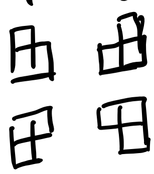
② 网络流问题: ⇒



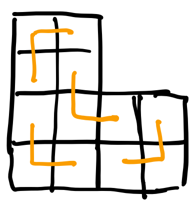
⇒ 最大流等于 n

最大匹配等于 n

* 骨牌改为 triomino:

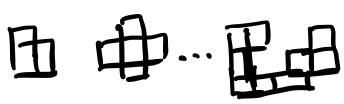


例: S:



对于一般的 S, 不存在多项式时间的算法. (前提: P≠NP)

* 有限多种形状的骨牌, S为无限的平面.



① 对于一些具体的例. 可解

② 对于通用的情形. 不可判定问题.