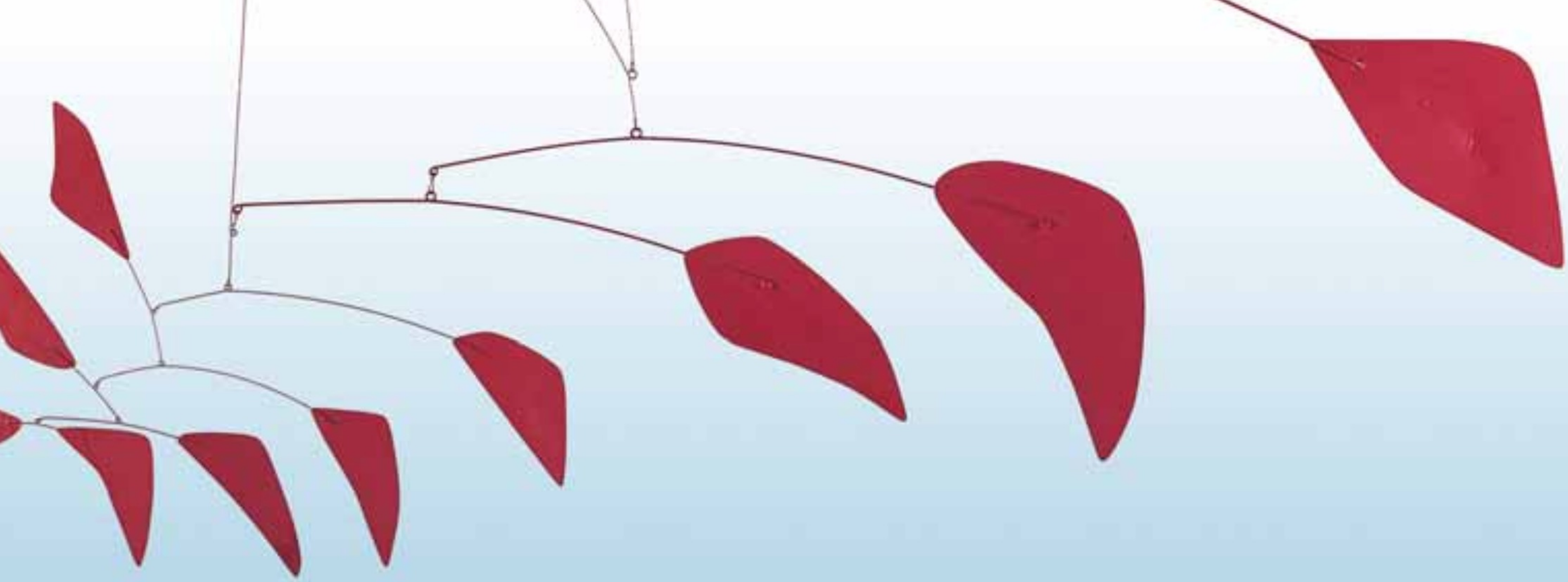


# 第一部分：基础知识

INTRODUCTION TO

# ALGORITHMS

THIRD EDITION

# 函数的增长

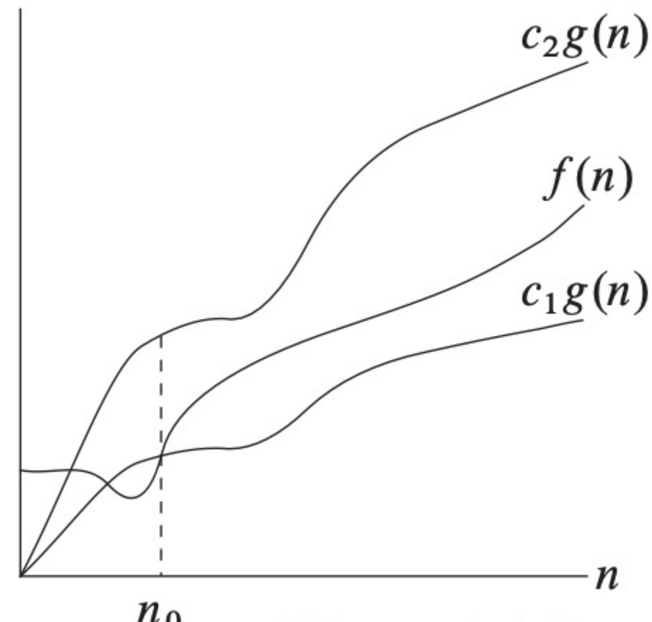
## 《算法导论》—— 第2讲

*jiacaicui@163.com*

# 内容提要

- 学习算法的渐近（asymptotic）效率
  - 描述函数的增长
  - 忽略低阶项和系数以抓住重点
- 学习一些比较函数增长的记号
  - $O \approx \leq$
  - $\Omega \approx \geq$
  - $\Theta \approx =$
  - $o \approx <$
  - $\omega \approx >$

# 渐近记号

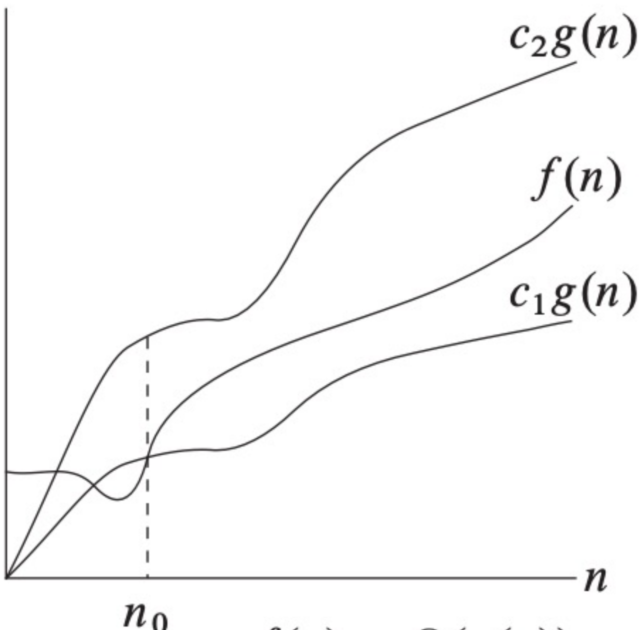



$f(n) = \Theta(g(n))$

# 渐近紧确界（asymptotically tight bound）

$$\Theta(g(n)) = \{f(n) \mid \exists c_1 > 0, c_2 > 0, n_0 > 0, \forall n \geq n_0, 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)\}$$

- 我们常将 $f(n) \in \Theta(g(n))$ 简写作 $f(n) = \Theta(g(n))$ 。
  - 后续的所有渐进记号都遵守这个约定。




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

- 求证：$\frac{1}{2}n^2 - 3n = \Theta(n^2)$

证明：$\Leftarrow c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2, n \geq n_0$

$\Leftarrow c_1 \leq \frac{1}{2} - \frac{3}{n} \leq c_2, n \geq n_0$

$\Leftarrow$ 取 $c_1 = \frac{1}{14}, c_2 = \frac{1}{2}, n_0 = 7$ 即可。

# 渐近上界 / 渐近下界

$$O(g(n)) = \{f(n) | \exists c > 0, n_0 > 0, \forall n \geq n_0, 0 \leq f(n) \leq cg(n)\}$$

- 当描述一个算法的运行时间为 $O(g(n))$ 时，通常指的是其在最坏情况下的运行时间的最紧上界。
  - 不同情况下运行时间不同
  - 上界有很多，最紧的上界才是最公允、精确的

$$\Omega(g(n)) = \{f(n) | \exists c > 0, n_0 > 0, \forall n \geq n_0, 0 \leq cg(n) \leq f(n)\}$$

- 定理3.1：

$$f(n) = \Theta(g(n)) \Leftrightarrow f(n) = O(g(n)) \land f(n) = \Omega(g(n))$$

  - 渐近紧确界 = 渐近上界 + 渐近下界

# 非渐近紧确界

$$o(g(n)) = \{f(n) \mid \forall c > 0, \exists n_0 > 0, \forall n \geq n_0, 0 \leq f(n) < cg(n)\}$$

$$\Leftrightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

$$\omega(g(n)) = \{f(n) \mid \forall c > 0, \exists n_0 > 0, \forall n \geq n_0, 0 \leq cg(n) < f(n)\}$$

$$\Leftrightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$$

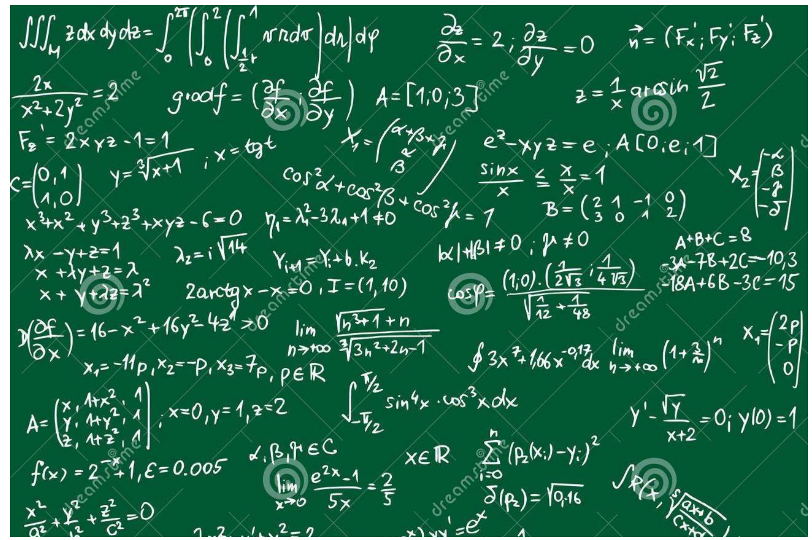
- $n^{1.99999} = o(n^2), n^2 \neq o(n^2), n^2 = O(n^2)$
- $n^{2.00001} = \omega(n^2), n^2 \neq \omega(n^2), n^2 = \Omega(n^2)$

# 渐近记号的性质

- 传递性：$f(n)=\Theta\big(g(n)\big) \wedge g(n)=\Theta\big(h(n)\big) \Rightarrow f(n)=\Theta(h(n))$
  - 对于 $O, \Omega, o, \omega$ 记号也成立。
- 自反性：$f(n)=\Theta(f(n))$
  - 对于 $O, \Omega$ 记号也成立。
- 对称性：$f(n)=\Theta\big(g(n)\big) \Leftrightarrow g(n)=\Theta(f(n))$
- 转置对称性： $f(n)=O\big(g(n)\big) \Leftrightarrow g(n)=\Omega(f(n))$
  - 对于 $o$ 和 $\omega$ 也是如此。
- 三分性：$\forall a \in \mathbb{R}, b \in \mathbb{R}, a<b \vee a=b \vee a>b$
  - 实数两两可比，但是函数不行。例如：$n$ 和 $n^{1+\sin(n)}$

# 标准记号与常用函数

## 必要的数学复习

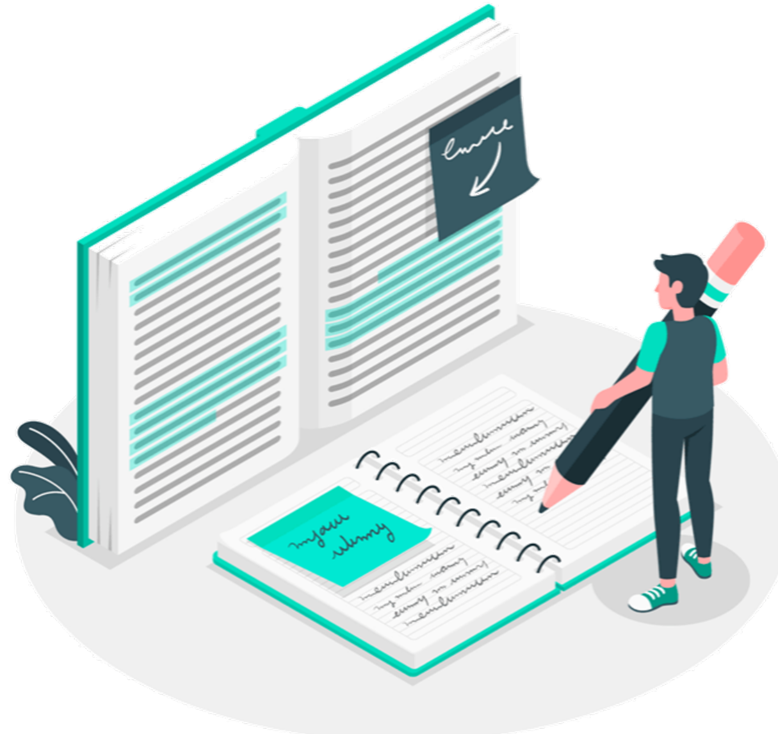


# 复习一些数学

- 单调性
- 向下取整与向上取整
- 模运算
- 多项式
- 指数
- 对数
- 阶乘
- 多重函数
- 多重对数函数
- 斐波那契数

斯特林公式：$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n\left(1+\Theta\left(\frac{1}{n}\right)\right)$

自行复习：中文版教材30-34页，英文版教材53-60页



# 本讲小结

# 内容提要

- 学习算法的渐近（asymptotic）效率
  - 描述函数的增长
  - 忽略低阶项和系数以抓住重点
- 学习一些比较函数增长的记号
  - $O \approx \leq$
  - $\Omega \approx \geq$
  - $\Theta \approx =$
  - $o \approx <$
  - $\omega \approx >$

# The End!