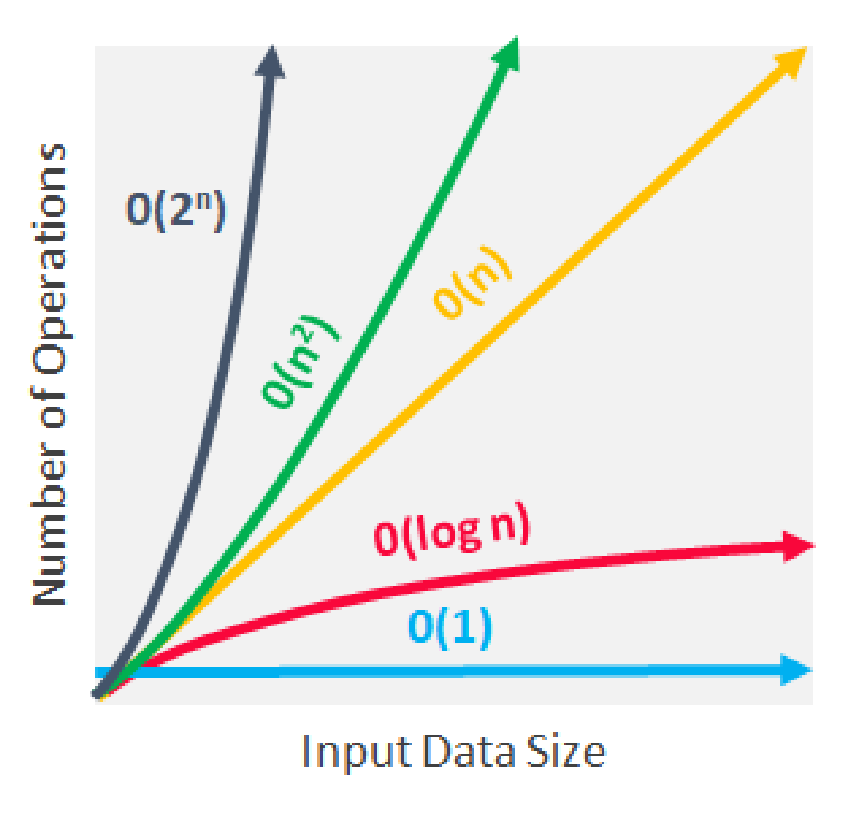


Phân tích độ phức tạp thuật toán

GV: Nguyễn Thanh Sơn

Nhóm 7:

- Nguyễn Trung Tuấn – 19522477
- Nguyễn Khả Tiến – 19522337
- Trịnh Nhật Tân – 19522179

Nội dung

1. Tham số quyết định và phép toán cơ sở.
2. Tỉ số tăng là gì?
3. Worst-case, Best-case, and Average-Case.
4. Các ký pháp Big-Oh, Big-Theta, Big-Omega.

Tham số quyết định và phép toán cơ sở

- Chiều dài của một list.
- Bậc của một đa thức hoặc số lượng các hệ số.
- Số lượng bits(b) trong biểu diễn nhị phân của n.

$$b = \log_2 n + 1$$

Tham số quyết định và phép toán cơ sở

- Để đo được độ hiệu quả của thuật toán thì phải có một phép đo không dựa trên các yếu tố không liên quan.
- Một cách tiếp cận khả thi là **đếm số lần thực thi** của mỗi phép toán trong một giải thuật

⟹ **Phép toán cơ sở:** phép toán mà đóng góp thời gian chạy nhiều nhất trong một giải thuật.

- Điển hình là các phép toán số học: +, -, *, /, … Thứ tự thời gian để thực hiện các phép toán số học là: /, * , (+, -).

Tham số quyết định và phép toán cơ sở

INSERTION-SORT(A)	*cost*	*times*
for $j \leftarrow 2$ **to** $length[A]$	c_1	n
do $key \leftarrow A[j]$	c_2	$n-1$
$\triangleright$ Insert $A[j]$ into the sorted		
$\triangleright$ sequence $A[1\,..\,j-1]$.	0	$n-1$
$i \leftarrow j-1$	c_4	$n-1$
while $i > 0$ and $A[i] > key$	c_5	$\sum_{j=2}^{n} t_j$
do $A[i+1] \leftarrow A[i]$	c_6	$\sum_{j=2}^{n}(t_j-1)$
$i \leftarrow i-1$	c_7	$\sum_{j=2}^{n}(t_j-1)$
$A[i+1] \leftarrow key$	c_8	$n-1$

Tham số quyết định và phép toán cơ sở

Problem	**Tham số quyết định**	**Phép toán cơ sở**
Tìm giá trị *k* trong mảng có *n* phần tử	Số lượng phần tử có trong mảng: *n*	Phép so sánh
Phép nhân 2 ma trận	Số chiều của ma trận hoặc số lượng các phần tử	Phép nhân
Kiểm tra số *n* có phải là số nguyên tố	*n*’size = số chữ số (biễu diễn nhị phân)	Phép chia

Tham số quyết định và phép toán cơ sở

- Giả sử $\boldsymbol{c_{op}}$ là thời gian thực thi của một phép toán cơ sở.
- $\boldsymbol{C(n)}$ là số lần thực hiện phép toán cơ sở.

$\Longrightarrow$ Công thức ước lượng thời gian $T(n)$ thực hiện giải thuật:

$$\boldsymbol{T(n) \approx c_{op}C(n)}$$

Tham số quyết định và phép toán cơ sở

Giả sử: $C(n) = \frac{1}{2}n(n-1)$

Thời gian gian chạy giải thuật sẽ lâu hơn bao nhiêu nếu kích cỡ input lớn gấp đôi?

$$C(n) = \frac{1}{2}n(n-1) = \frac{1}{2}n^2 - \frac{1}{2}n \approx \frac{1}{2}n^2$$

$$\Rightarrow \frac{T(2n)}{T(n)} \approx \frac{c_{op}C(2n)}{c_{op}C(n)} \approx \frac{\frac{1}{2}(2n)^2}{\frac{1}{2}n^2} = 4$$

Tỉ suất tăng là gì?

- Kích thước đầu vào nhỏ thông thường được tính một cách ngay lập tức, do đó chúng ta chỉ quan tâm thuật toán hoạt động ra sao khi $n \to \infty$.
- Thực tế, với n là giá trị nhỏ thì phần lớn các giải thuật đều cho ra thời gian đều như nhau. Chỉ khi $n \to \infty$ thì sự khác biệt ngày càng rõ.

Tỉ suất tăng là gì?

n	$log_2 n$	n	$nlog_2 n$	n^2	n^3	2^n	$n!$
10	3.3	10^1	$3.3 \cdot 10^1$	10^2	10^3	10^3	$3.6 \cdot 10^6$
10^2	6.6	10^2	$6.6 \cdot 10^2$	10^4	10^6	$1.3 \cdot 10^{30}$	$9.3 \cdot 10^{157}$
10^3	10	10^3	$1.0 \cdot 10^4$	10^6	10^9		
10^4	13	10^4	$1.3 \cdot 10^5$	10^8	10^{12}		
10^5	17	10^5	$1.7 \cdot 10^6$	10^{10}	10^{15}		
10^6	20	10^6	$2.0 \cdot 10^7$	10^{12}	10^{18}		

Worst-case, Best-case, and Average-Case.

Worst-case: trường hợp xấu nhất (hiệu quả thấp nhất)

$$C_{worst}(n) = n$$

- Giải thuật chạy lâu nhất trong số các input phù hợp.
- Cách xác định: Phân tích giải thuật để xem loại input nào cho ra số lần đếm $C(n)$ là lớn nhất giữa những input phù hợp.

Worst-case, Best-case, and Average-Case.

Best-case: trường hợp tốt nhất (hiệu quả nhất)

$$C_{best}(n) = 1$$

- Giải thuật chạy nhanh nhất trong số các input phù hợp.
- Cách xác định: Phân tích thuật toán để biết loại input nào cho ra số lần đếm $C(n)$ là nhỏ nhất giữa những input phù hợp.

Worst-case, Best-case, and Average-Case.

Average-case = trung bình cộng

- Chạy giải thuật nhiều lần bằng những input cùng size *n* (dùng một số hàm phân phối để tạo ra các input đó).
- Tính tổng thời gian chạy và chia cho số lần thử.

Các ký pháp

Ký hiệu trong phần này:

- $t(n)$: thời gian chạy của giải thuật.(thông thường là được chỉ ra qua phép đếm C(n)).
- $g(n)$: là hàm dùng để so sánh với $t(n)$.

Các ký pháp

Ký pháp Big-Oh:

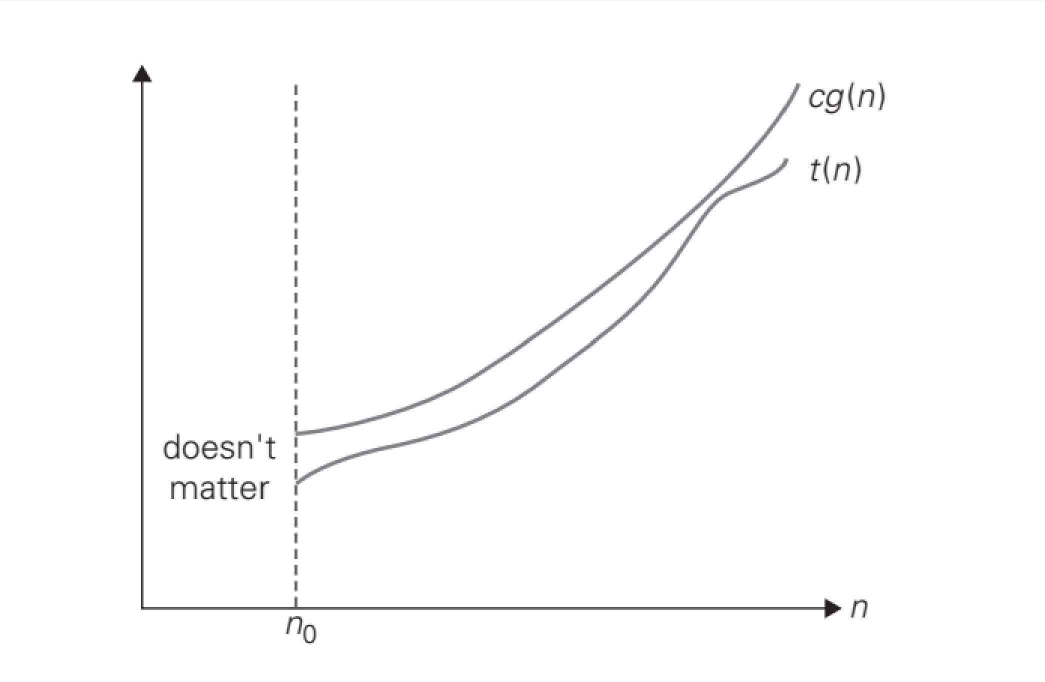
- $O\big(g(n)\big)$ là tập của tất cả các hàm có tỉ suất tăng thấp hơn hoặc cùng với g(n) (với bội số không đổi và $n \to \infty$).
- Ví dụ:

$$n \in O(n^2) \qquad 100n + 5 \in O(n^2) \qquad \frac{1}{2}n(n-1) \in O(n^2)$$

Các ký pháp

Định nghĩa toán học: $t(n) \in O(g(n))$

$$\Leftrightarrow t(n) \le cg(n) \qquad (\exists c > 0, n \ge n_0)$$



Các ký pháp

Chứng minh: $100n + 5 \in O(n^2)$

$$100n + 5 \leq 100n + n \ (\forall \mathrm{n} \geq 5) = 101n \leq 100n^2$$

Chứng minh xong với $c = 101, n_0 = 5$.

Thực tế là định nghĩa phía trên cho ta rất nhiều cách chọn khác nhau với hai hằng số c và n_0.

Ví dụ: $100n + 5 \leq 100n + 5n \ (\forall \mathrm{n} \geq 1) = 105n$

$$c = 105, n_0 = 1$$

Các ký pháp

Ký pháp Big-Omega (Ω):

- $\Omega(g(n))$ là tập của tất cả các hàm có tỉ suất tăng cao hơn hoặc bằng với g(n) (với bội số không đổi và $n \to \infty$).
- Ví dụ:

$$n^3 \in \Omega(n^2) \qquad \frac{1}{2}n(n-1) \in \Omega(n^2) \qquad \text{but} 100n + 5 \in \Omega(n^2)$$

Các ký pháp

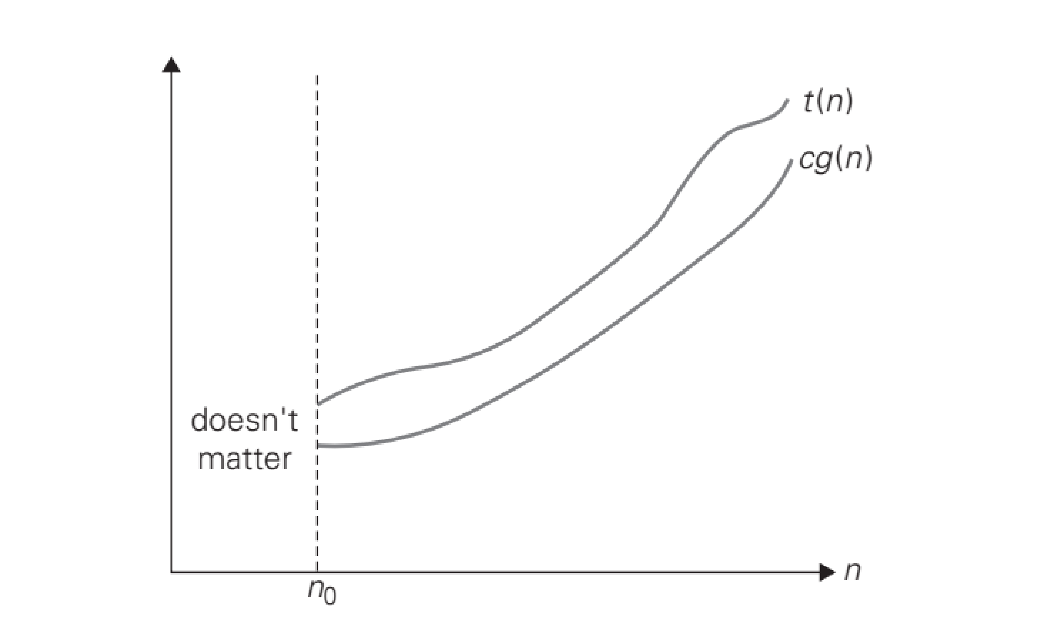
Định nghĩa toán học: $t(n) \in \Omega(g(n))$

$\Leftrightarrow t(n) \geq cg(n) \qquad (\exists c > 0, n \geq n_0)$

Ví dụ chứng minh: $n^3 \in \Omega(n^2)$

$$n^3 \geq n^2 \quad (\forall \mathrm{n} \geq 0)$$

Ở đây chúng ta chọn: $c = 1, n_0 = 0$



Các ký pháp

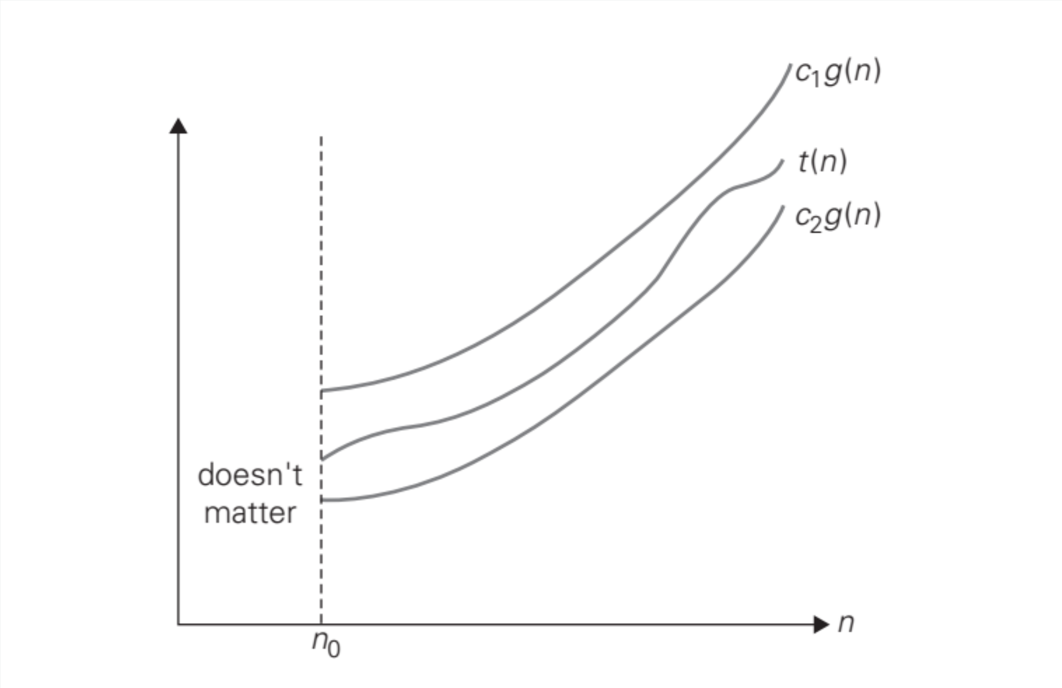
Ký pháp Big-Theta (Θ):

- $\Theta(g(n))$ là tập của tất cả các hàm có tỉ suất tăng bằng với g(n) (với bội số không đổi và $n \to \infty$).
- Do đó mọi hàm bậc 2 "$an^2 + bn + c$", với mọi $a > 0$ đều thuộc $\Theta(n^2)$.

Các ký pháp

Định nghĩa toán học: $t(n) \in \Theta(g(n))$

$$\Leftrightarrow c_2 g(n) \le t(n) \le c_1 g(n) \qquad (\exists c_1, c_2 > 0, n \ge n_0)$$



Các ký pháp

Chứng minh: $\frac{1}{2}n(n-1) \in \Theta(n^2)$

Chứng minh cận trên, bất đẳng thức vế phải:

$$\frac{1}{2}n(n-1) = \frac{1}{2}n^2 - \frac{1}{2}n \le \frac{1}{2}n^2 \qquad (\forall n \ge 0)$$

Chứng minh cận dưới, bất đẳng thức vế trái:

$$\frac{1}{2}n(n-1) = \frac{1}{2}n^2 - \frac{1}{2}n \ge \frac{1}{2}n^2 - \frac{1}{2}n\frac{1}{2}n \ (\forall n \ge 2) = \frac{1}{4}n^2$$

Do đó, chúng ta có thể chọn $c_2 = \frac{1}{4}, c_1 = \frac{1}{2}\ và\ n_0 = 2.$

Các ký pháp

THEOREM If $t_1(n) \in O(g_1(n))$ and $t_2(n) \in O(g_2(n))$, then

$$t_1(n) + t_2(n) \in O(\max\{g_1(n), g_2(n)\}).$$

(The analogous assertions are true for the Ω and Θ notations as well.)

PROOF The proof extends to orders of growth the following simple fact about four arbitrary real numbers a_1, b_1, a_2, b_2: if $a_1 \le b_1$ and $a_2 \le b_2$, then $a_1 + a_2 \le 2\max\{b_1, b_2\}$.

Since $t_1(n) \in O(g_1(n))$, there exist some positive constant c_1 and some nonnegative integer n_1 such that

$$t_1(n) \le c_1 g_1(n) \quad \text{for all } n \ge n_1.$$

Similarly, since $t_2(n) \in O(g_2(n))$,

$$t_2(n) \le c_2 g_2(n) \quad \text{for all } n \ge n_2.$$

Let us denote $c_3 = \max\{c_1, c_2\}$ and consider $n \ge \max\{n_1, n_2\}$ so that we can use both inequalities. Adding them yields the following:

$$\begin{aligned} t_1(n) + t_2(n) &\le c_1 g_1(n) + c_2 g_2(n) \\ &\le c_3 g_1(n) + c_3 g_2(n) = c_3[g_1(n) + g_2(n)] \\ &\le c_3 2\max\{g_1(n), g_2(n)\}. \end{aligned}$$

Hence, $t_1(n) + t_2(n) \in O(\max\{g_1(n), g_2(n)\})$, with the constants c and n_0 required by the O definition being $2c_3 = 2\max\{c_1, c_2\}$ and $\max\{n_1, n_2\}$, respectively. ■

Dùng lim để so sánh tỉ số tăng.

$$\lim_{n\to\infty}\frac{t(n)}{g(n)} = \begin{cases} 0 \\ c \\ \infty \end{cases}$$

Trong đó:

- $lim = 0$: $t(n)$ có tỉ số tăng nhỏ hơn g(n).
- $lim = c$: $t(n)$ có cùng tỉ số tăng với g(n).
- $lim = \infty$: $t(n)$ có tỉ số tăng lớn hơn g(n).