به نام خدا

مرور رشد توابع $O , \Omega , \Theta , o , \omega$

زمان الگوریتم به صورت تابعی از سایز مسئله بیان می‌شود.

زمان الگوریتم: جنس این زمان، واحد آن؟

ثانیه second نیست

بلکه تعداد دستورات پایه‌ای است که اجرا می‌شود

زمان الگوریتم $T(n) = 2n^2 + n$

سایز مساله

سایز مساله ؟

مثال 1) n عدد را میخواهیم باهم جمع بزنیم.

مثال 2) 2 عدد n بیتی را میخواهیم جمع بزنیم.

در مثال 1 سایز مساله $= n =$ تعداد اعداد

در مثال 2 سایز مساله $= n =$ تعداد بیت

در مسایل مختلف سایز مساله چه خواهد بود؟

در واقع سایز مساله = تعداد بیت‌های ورودی
(ورودی در مبنای ۲ کدگذاری می‌شود)

در مثال 1) n عدد داریم (به طور ضمنی فرض
شده است که هر عدد در یک word ذخیره می‌شود که سایز
هر word (تعداد بیت‌ها) ثابت است مثلاً ۸ یا ۱۶ یا ۳۲
یا ۶۴ بیت است)

سایز مساله = $64n$ = $n \times 64$ = تعداد بیت‌ها
تعداد بیت‌های ورودی

مثال 2)

2 عدد داریم و هر عدد n بیت است

سایز داده $= 2 \times n = 2n$

تعداد بیت‌ها

ورودی

$$(C \times n)^{c'} = c^{c'} \times n^{c'} = c'' \times n^{c'}$$

مرور O , Ω , Θ , o , ω

مقدار دقیق و ضرایب تابع زمان اجرای الگوریتم ها برای ما اهمیت ندارد

بلکه رشد این توابع برای ما مهم است

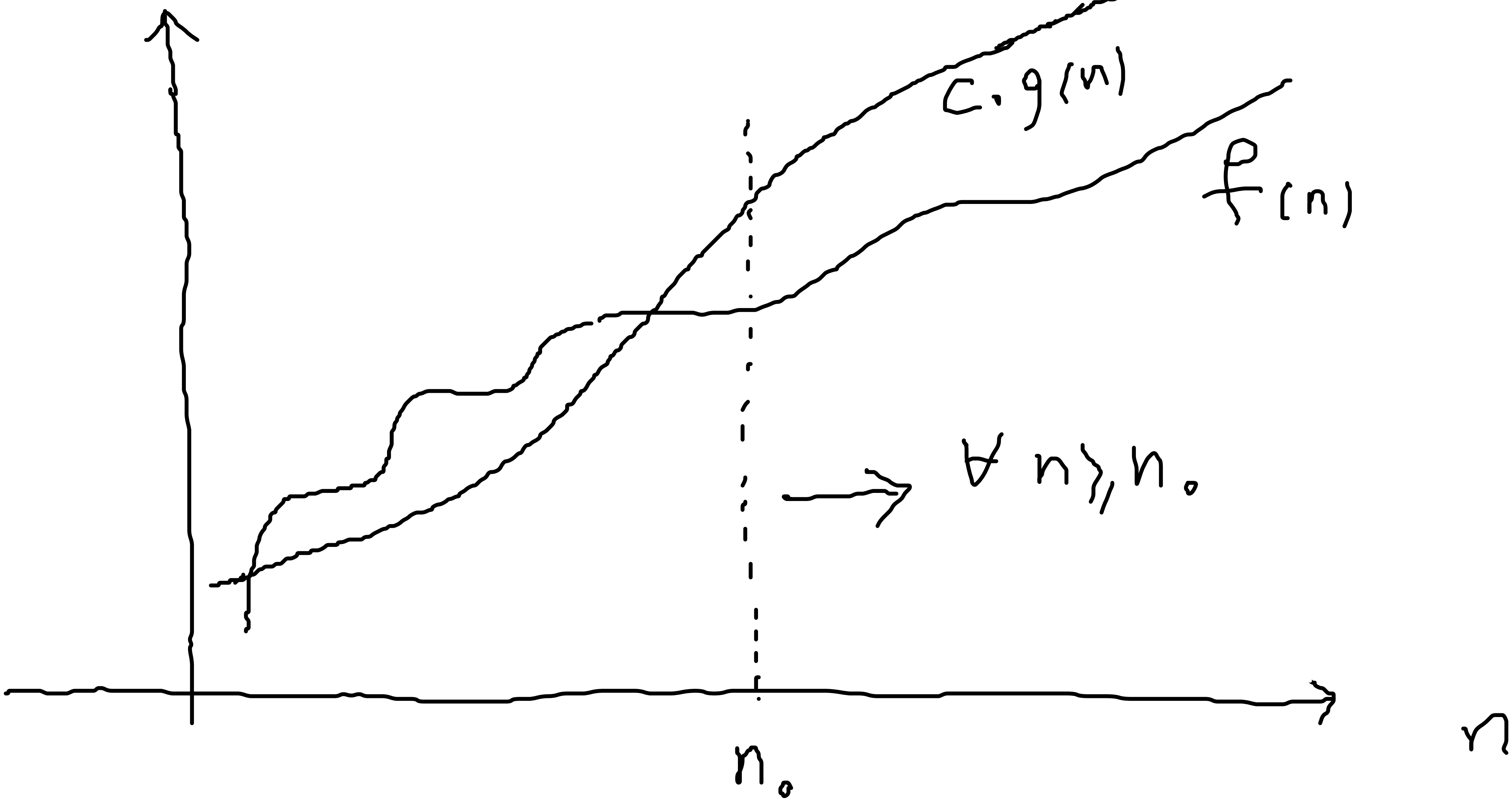
تابع زمان اجرا

n

سایز مسأله

$f(n) \in O(g(n)) \iff \exists\, c > 0,\ n_0 > 0$

$\forall n \geq n_0 \quad f(n) \leq c \cdot g(n)$



رشد تابع f(n) کمتر یا مساوی رشد تابع g(n) است

$$\lim_{n\to\infty} \frac{g(n)}{f(n)} = \begin{cases} c \neq 0 & \text{ثابت} \\ 0 \\ \infty \end{cases}$$

رشد $g(n)$
کمتر یا مساوی
$f(n)$ است

مثال

$$f(n) = 2n^2 + n$$

$$\in O(n^2)$$

$$\in O(n^3)$$

$$\notin O(n)$$

$O(n^2) =$ مجموعه تمام توابعی که رشد آنها از n^2 کمتر یا مساوی است.

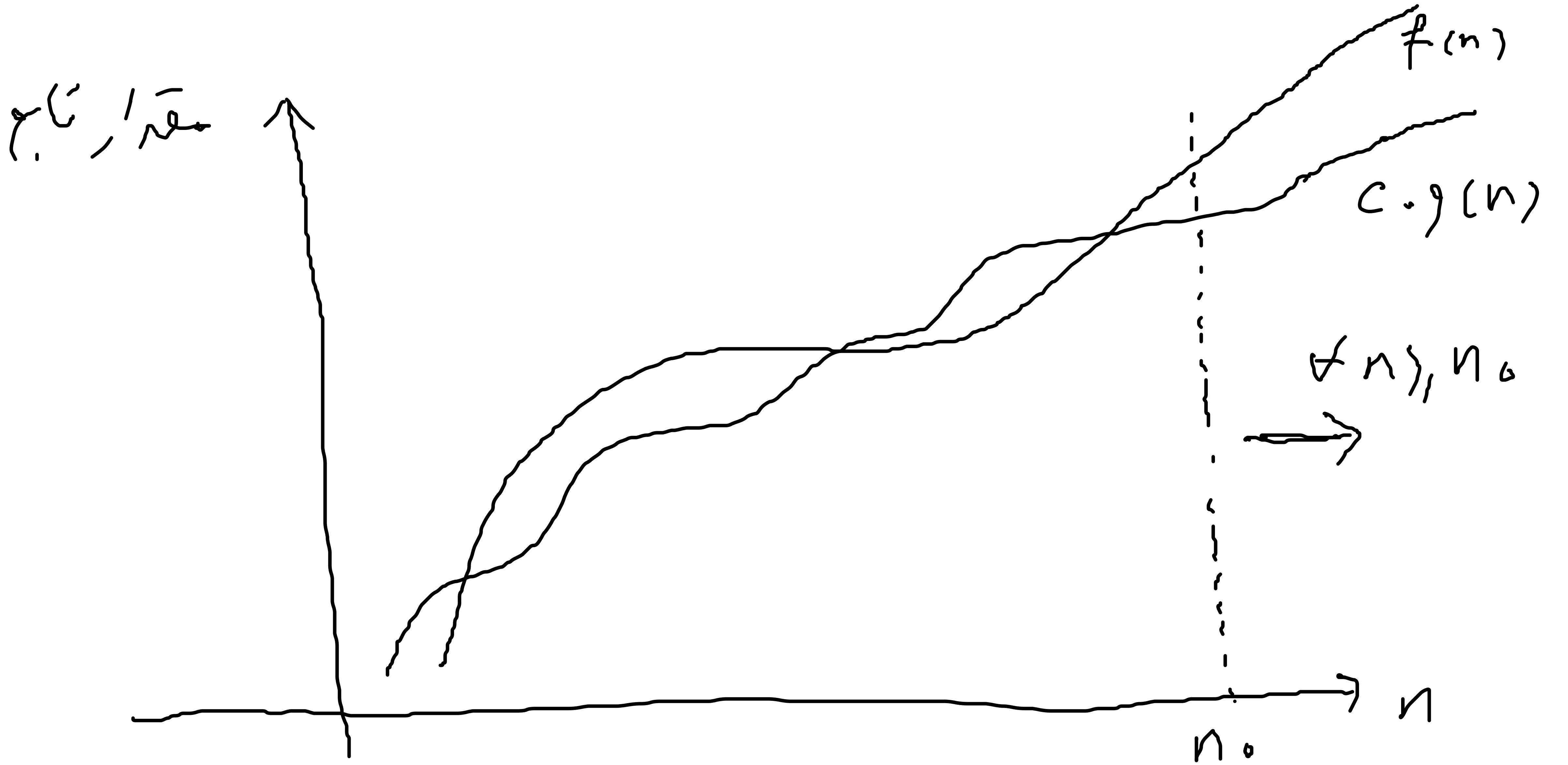
$n \in O(n^2)$

$n^2 \in O(n^2)$

$n^3 \notin O(n^2)$

$$f(n) \in \Omega(g(n)) \iff \exists\, c>0,\ n_0>0$$

$$\forall n \geq n_0 \quad f(n) \geq c \cdot g(n)$$



سایز مسئله

رشد تابع $g(n)$ حد پایینی رشد تابع
$f(n)$ است

رشد تابع $f(n)$ بیشتر یا مساوی رشد تابع $g(n)$ است

$$\lim_{n\to\infty} \frac{g(n)}{f(n)} = \begin{cases} c \neq 0 \\ \text{یا} \\ 0 \end{cases}$$

$$f(n) = 2n^2 + n$$

$$\in \Omega(n)$$

$$\in \Omega(n^2)$$

$$\notin \Omega(n^3)$$

$\Omega(n^2) =$ مجموعه همه توابعی که رشد آنها بیشتر یا مساوی n^2 است.

$$n \notin \Omega(n^2)$$

$$n^2 \cancel{\notin} \in \Omega(n^2)$$

$$n^3 \in \Omega(n^2)$$

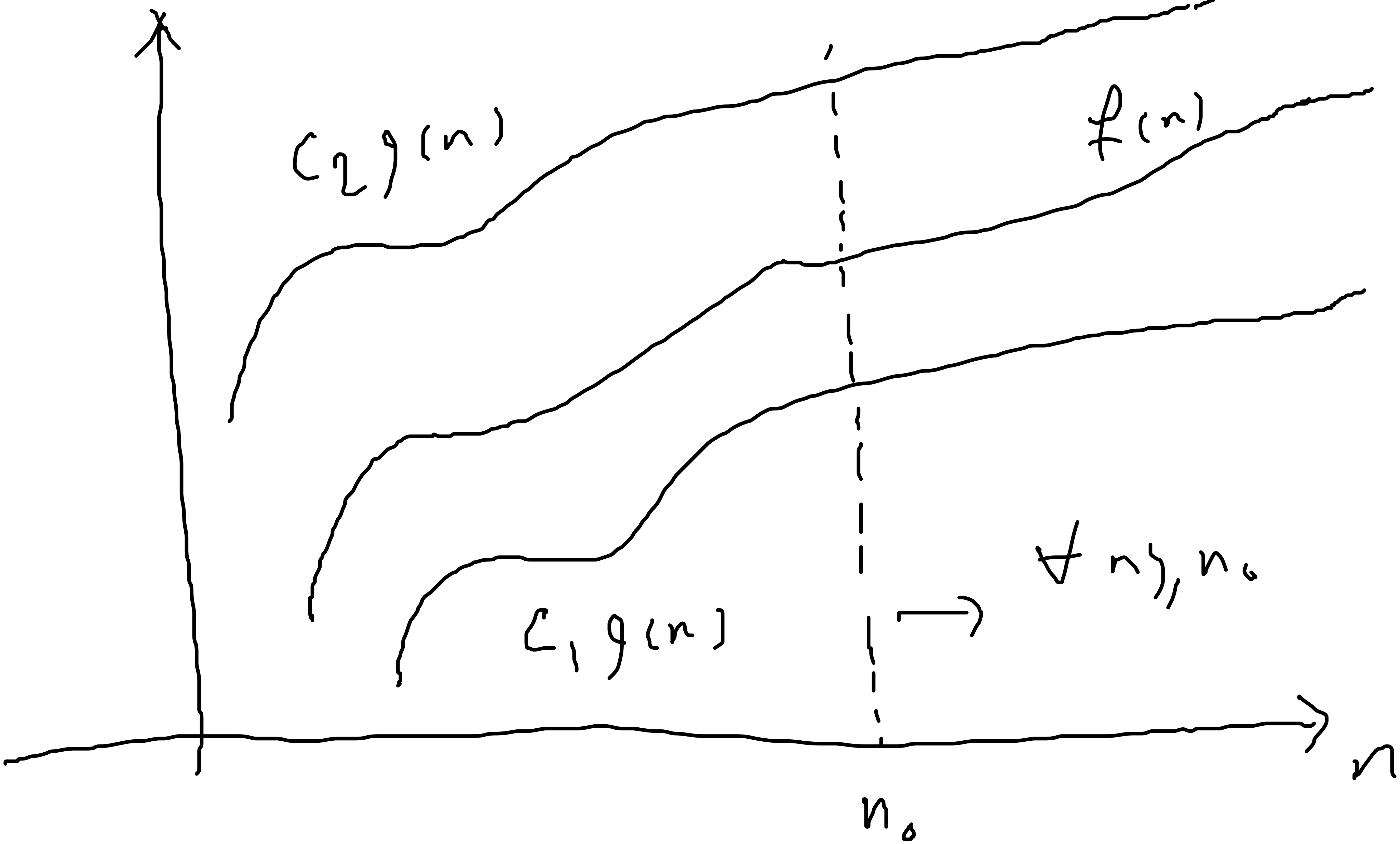
کاربرد O : برای تضمین در مورد حد بالای رشد تابع زمان اجرای <u>الگوریتم</u> است.
(~~...~~ یعنی تضمین میدهیم که از یک حدی بدتر (بیشتر) نیست)

کاربرد Ω : برای بیان دشواری <u>مساله ها</u>

زمان حل مساله $A \in \Omega(n^2)$

$$f(n) \in \Theta(g(n)) \iff \exists\, c_1 > 0,\ c_2 > 0,\ n_0 > 0$$

$$\forall n \geq n_0 \quad c_1 g(n) \leq f(n) \leq c_2 g(n)$$



رشد $f(n)$ مساوی رشد $g(n)$ است

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

$$f(n) = 2n^2 + n$$

$$\in \Theta(n^2)$$

$$\in \Theta(5n^2 + 2n)$$

$$\notin \Theta(n)$$

$$\notin \Theta(n^3)$$

$\Theta(n^2) =$ مجموعه همه توابعی که رشد آنها
<u>مساوی</u> n^2 است.

$$2n^2+3 \in \Theta(n^2)$$

$$5n^2+10 \in \Theta(n^2)$$

$$n \notin \Theta(n^2)$$

$$n^3 \notin \Theta(n^2)$$

$f(n) \in \Theta(g(n)) \iff f(n) \in O(g(n))$

and

$f(n) \in \Omega(g(n))$

$f(n) \in o(g(n)) \iff \underline{\forall} c>0 \quad \exists n_0>0 \mid$

$\forall n \geq n_0 \quad f(n) < c \cdot g(n)$

رشد $f(n)$ <u>کمتر</u> از رشد $g(n)$ است

رشد $g(n)$ حد بالای اکید رشد $f(n)$ است

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

$f(n) = 2n^2 + n$

$\notin o(n^2)$

$\in o(n^3)$

$\notin o(n)$

کاربرد ۵ : آیا الگوریتمی وجود دارد که

زمان آن $O(n^2)$

یعنی الگوریتم هایی داریم که زمان آن $\Theta(n^2)$ است

می خواهیم ببینیم الگوریتمی که زمان آن کمتر از

n^2 (رشد آن کمتر از رشد n^2) باشد

وجود دارد ؟

$$f(n) \in \omega(g(n)) \iff \forall\, c>0 \quad \exists\, n_0>0$$

$$\forall\, n \geq n_0 \quad f(n) \geq c \cdot g(n)$$

رشد $g(n)$ کمتر از رشد $f(n)$ است

($f(n)$ بیشتر است)

رشد $g(n)$ حد پایین اکید $f(n)$ است

$$f(n) = 2n^2 + n$$

$$\in \omega(n)$$

$$\notin \omega(n^2)$$

$$\notin \omega(n^3)$$

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

$\forall \; a, b \in \mathbb{R}$

$a < b$ ۱

$a = b$ ۲ سه حالت

داریم

$a > b$ ۳

برای هر دو تابع $f(n)$ و $g(n)$

$f(n) \in o(g(n))$ ۱

$f(n) \in \Theta(g(n))$ ۲

$f(n) \in \omega(g(n))$ ۳

$$f(n) = n$$

$$g(n) = n^{1+\sin n}$$

این دو تابع نسبت به هم هیچ یک از

۳ حالت صفحه قبل را

ندارند.

رشد هر تابع چندجمله‌ای از رشد هر تابع

چندلگاریتمی بیشتر است:

$$f(n) = n^{c} \in \omega\left(g(n) = \lg^{c'} n\right)$$

چندجمله‌ای

چندلگاریتمی

$$c, c' > 0$$

رشد هر تابع نمایی از رشد هر تابع چندجمله‌ای
بیشتر است :

$$f(n) = a^n \in \omega(g(n) = n^c)$$

$$c > 0 \quad , a > 1$$

$$\log_a^n = \underline{\underline{\log_a^b}} \times \log_b^n$$

عدد ثابت

$$\log_a^n \in \Theta(\log_b^n)$$

$$\log_a^{n^b} = b \log_a^n \qquad a^{\log_a^n} = n$$

$$a^{\log_b^n} = n^{\log_b^a}$$