- Turn in your group's worksheet
- Sit wherever you want for lecture

Big O

Def Let $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$. We say that f is $O(g)$ if $\exists\, c > 0, n_0 \geq 0$ s.t.

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

We also write $f = O(g)$ to mean f is $O(g)$.

Why O? "order" of a function.

ex $f(n) = 3n^2 + 2$ is $O(n^2)$.

Proof We must give $c > 0$, $n_0 \geq 0$ s.t.

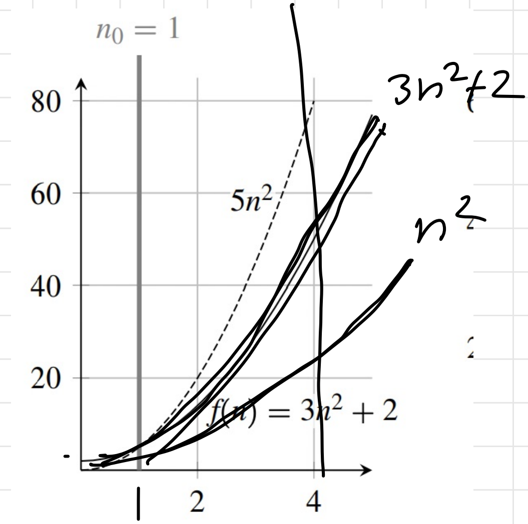
$$\forall n \geq n_0 : f(n) \leq c \cdot n^2.$$

Note that $\forall n \geq 1$, $2n^2 \geq 2$, so

$\rightarrow$ $\forall n \geq 1 : f(n) = 3n^2 + 2 \leq 3n^2 + 2n^2 = 5n^2.$

So we can choose $c = 5$, $n_0 = 1$

and we have $\forall n \geq \underbrace{1}_{n_0} : f(n) \leq \underbrace{5}_{c} \cdot n^2$



(a) $f(n)$ is $O(n^2)$ with $c = 5$ and $n_0 = 1$.

$f(n) \le 5 \cdot n^2$

$\underline{3n^2+2} = \underline{O(n^2)}$

$\forall n \ge n_0 \;\; 3n^2+2 \le cn^2$

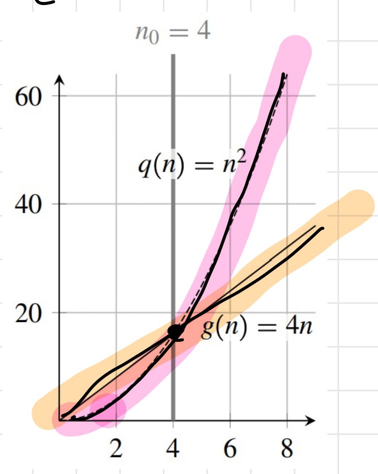
$n_0 = 4$

<u>ex</u> $g(n) = 4n$ is $O(n^2)$. T

<u>Proof</u> We must give $c > 0$, $n_0 \ge 0$ s.t.

$4n \le c \cdot n^2$ for all $n \ge n_0$.

Note that $4n$ and n^2 cross exactly one time, at $n=4$. So we can pick $n_0 = 4$, $c = 1$.



$\forall n \ge 4 : 4n \le 1 \cdot n^2$

n_0 c

Q Is $4n = O(4n^2)$? ~~z~~ yes

Is $3n^2 + 2 = O(\frac{1}{2}n^2)$ yes

we prefer $3n^2+2 = O(n^2)$ — the simplest form in big O.

$n^2 = O(n^3)$
but $n^2 \neq n^3$

another ex

n^3 is not $O(n^2)$. T

Proof we WTS $\forall c > 0, n_0 \geq 0, \exists n \geq n_0$:

$$\underbrace{n^3}_{f(n)} > c \cdot \underbrace{n^2}_{g(n)}.$$

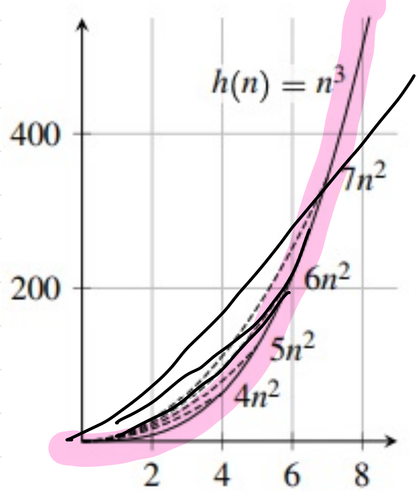
To do this, we show how to construct such an n for any choice of c, n_0.

Let $c > 0$ and $n_0 \geq 0$. We need $n \geq n_0$ s.t. $n^3 > cn^2$. Let $n = c+1$.

$$n^3 = (c+1)^3,$$

which is greater than $cn^2 = c(c+1)^2$, since $c > 0$. So we have $n^3 > cn^2$.
But if $n_0 > c+1$, we can't set $n = c+1$, because we need $n \geq n_0$. So choose $n = \max(n_0, c+1)$, and the inequality still holds.

We have shown how to produce an $n \geq n_0$ s.t. $n^3 > cn^2$ for any $c > 0$, $n_0 \geq 0$, so $n^3 \neq O(n^2)$.



(c) No value of c has $n^3 < cn^2$ for all large n.

$$2n^3 + n^2 = O(n^3)$$

Common Distinct Functions

name	example f(n)	generic f(n)	tightest O(·)
constant	$f(n) = 10$	$f(n) = c$, $c \in \mathbb{R}^{\geq 0}$	$O(1)$
log	$3 \log_2 n$	$c \log_b n$, $b \in \mathbb{R}^{\geq 1}$, $c \in \mathbb{R}^{\geq 0}$	$O(\log n)$
linear	$5n$	cn, $c \in \mathbb{R}^{\geq 0}$	$O(n)$
n log n	$8n \log_{10} n$	$cn \log_b n$	$O(n \log n)$
quadratic	$3n^2 + n$	deg-2 polynomial	$O(n^2)$

cubic $\qquad$ deg-3 polynomial $\qquad O(n^3)$

$\vdots$

deg k polynomial $\qquad O(n^k)$

exponential $\qquad 2^n \qquad O(2^n)$

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

$n! = O(n^n)$

$n \cdot (n-1) \cdot (n-2) \cdots 1$

what do we mean when we say "distinct"?

- later on the list $\neq O$(earlier on list)

  $2^n \neq O(n^{100})$

  $3n^2 \neq O(n \log n)$

- earlier on list $= O$(later on list)

  $n^{100} = O(2^n)$

  $n^{100} = O(n^{100})$

can I write $f(n) = O(\sin(n))$?
for any $f(n)$?

Other asymptotic relations

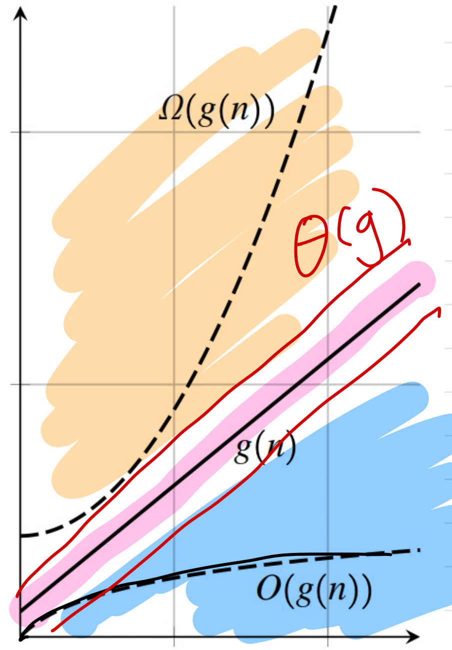
Big Omega (Ω) - f grows no slower than g

f is $\Omega(g)$ if $\exists d > 0, n_0 \geq 0$ s.t.

$\forall n \geq n_0 : f(n) \geq d \cdot g(n)$

Big Theta (Θ)

f is $\Theta(g)$ if $f = O(g)$ and $f = \Omega(g)$



Properties of Big O

Lemma 6.2 Asymptotic Equivalence of max and sum

$f(n) = O(g(n) + h(n)) \iff f(n) = O(\max(g(n), h(n)))$

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

$f(n) = O(\max(n^2, n))$

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

This lemma tells us that we can drop lower order terms.

Proof Because lemma 6.2 is $\iff$, we prove each separately.

($\Rightarrow$) WTS $f(n) = O(g(n) + h(n)) \Rightarrow$
$f(n) = O(\max(g(n), h(n)))$.

Assume $f(n) = O(g(n) + h(n))$.
WTS $f(n) = O(\max(g(n), h(n)))$. $\leftarrow$

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

$$f(n) \leq c \cdot (g(n) + h(n))$$
$$\leq c \cdot (\max(g(n), h(n)) + h(n))$$
$$\leq c \cdot (\max(g(n), h(n)) + \max(g(n), h(n)))$$
$$= c \cdot 2 \cdot \max(g(n), h(n))$$

Choose $c' = 2c$ and $n_0' = n_0$

goal: find some $c' > 0$, $n_0' \geq 0$ s.t.
$\forall n \geq n_0 : f(n) \leq c' \cdot \max(g(n), h(n))$

$\square$

($\Leftarrow$) in book

Lemma 6.5 Asymptotics of polynomials

Let $f(n) = \sum_{i=0}^{k} a_i n^i = a_0 n^0 + a_1 n^1 + a_2 n^2 + \dots a_k n^k$

be a deg-k polynomial. Then $f(n) = O(n^k)$.

Lemma 6.3 Transitivity of big O

$$\begin{tikzcd} f(n) \arrow[r] \arrow[rr, bend right] & g(n) \arrow[r] & h(n) \end{tikzcd}$$

If $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

More interesting properties in book

To measure the runtime of an algorithm, we:

① give $f(n)$ counting worst case # of primitive operations on input of size n

② find simplest $g(n)$ s.t. $f(n) = \Theta(g(n))$

we often say big O

That $g(n)$ is runtime of the algorithm.