Asymptotic notation: O, Ω, Θ

time change of today's office hour: 4-5pm

Note Title 1/19/2016

- summary of running time analysis (using insertion sort as an example)

worst-case analysis:

1. The longest running time for any input of size n: the worst case
   e.g. 5, 3, 2, 1, 0 for insertion sort

2. The upperbound on the running time for any input of size n
3. The worst case happens often: eg. database search: failed to find a match
4. The average case is often roughly as bad as the worst case
   e.g. Insertion sort: roughly half elements $\le$ key $\quad t_j = \frac{j}{2}$
   .... $\ge$ key

* Caveate about estimating running time of a function/sub-routine
  e.g. matrix multiplication in matlab

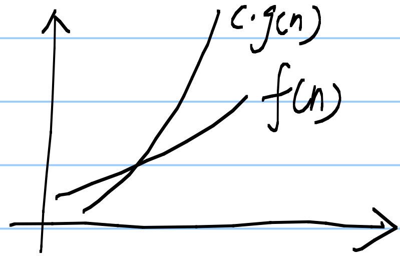
simplifications / approximations

	n	$\frac{3}{2}n^2$	$\frac{3}{2}n^2 + \frac{7}{2}n - 4$	% difference
array size	10	150	181	17% $= \frac{181-150}{180}$
	50	3,750	3,921	4.4%
	100	15,000	15,436	2.3%
	500	375,000	376,746	0.5%

highest order term finally dominates the output

Asymptotic notations: O Ω Θ

- $f(n) = O(g(n))$ iff there exist a constant $c > 0$ and a value n_0 s.t. for any $n \geq n_0$, $f(n) \leq c \cdot g(n)$



e.g. $n = O(n^2)$ $n = O(n)$ $2n = O(n)$

e.g1. Prove $n^2+7n+5=O(n^2)$

$c \cdot n^2 \geq n^2+7n+5$
when $n \geq n_0$

method1: proof: $f(n)=n^2+7n+5$

$\leq n^2+7n^2+5n^2=13n^2$ when $n\geq 1$

Thus, let $c=13$, $n_0=1$ we have

$f(n)\leq 13n^2$ when $n\geq 1$ ∎

note: $n^2\geq n$ when $n\geq 1$

method2: $f(n)\leq c\cdot g(n) \iff \underline{f(n)-c\,g(n)\leq 0}$ when $n\geq n_0$

proof: Let $c=2$ and $n_0=8$ consider that

$n^2+7n+5-2n^2=0$ when $n\approx 7.65$

The function $n^2+7n+5-2n^2$ is monotonically decreasing when $n\geq 8$

because the derivative of difference

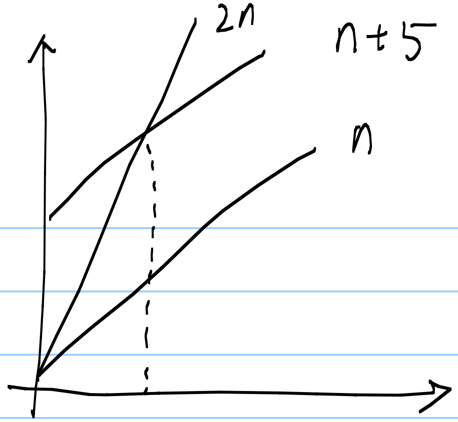
$$(n^2+7n+5-2n^2)' = (-n^2+7n+5)' = -2n+7 < 0 \text{ when } n \geq 8$$

Thus $n^2+7n+5-2n^2 \leq 0$ when $n \geq 8$

$$n^2+7n+5 \leq 2n^2$$

$n = O(n^2)$

$f(n) = O(g(n))$



$f(n) = n+5$

$g(n) = n$

$n+5 = O(n)$ Choose $c=2$.

$2n = n+5 \Rightarrow n=5$

In-class exercises:

prove $3n^2 + 2n + 5 = O(n^2+n)$

$c \cdot (n^2+n) \geq 3n^2 + 2n + 5$

when $n \geq n_0$

Proof: Let $C=3$, $n_0=5$,

$$\begin{aligned} 3(n^2+n) &= 3n^2 + 3n \\ &= 3n^2 + 2n + n \\ &\geq 3n^2 + 2n + 5 \quad \text{when } n \geq 5 \end{aligned}$$

$\blacksquare$

Let $c=10$, $n_0=1$

$$10(n^2+n) = 10n^2+10n$$
$$= 3n^2+2n^2+5n^2+10n$$
$$\geq 3n^2+2n+5+0$$

Invalid proof: the highest order term of n^2+n is n^2

- - - - - - - - - - . . $3n^2+2n+5$ is n^2

Thus, - - - - .

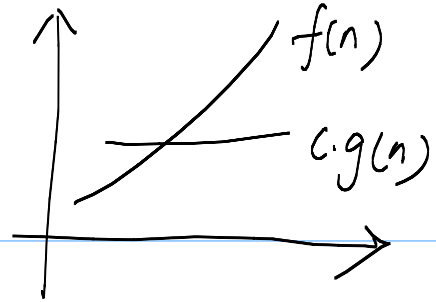
— Asymptotic lowerbound

$f(n)=\Omega(g(n))$ iff there exist a constant $c>0$ and a value n_0

s.t. $c \cdot g(n) \leq f(n)$ when $n \geq n_0$

e.g $n^2=\Omega(n)$ $n^3=\Omega(n^2)$

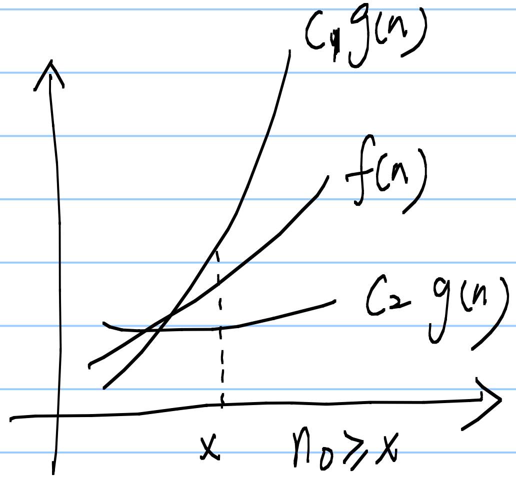
$n^2 = \Omega(5n^2)$



- $f(n) = \Theta(g(n))$ iff $f(n)$ is both $O(g(n))$ and $\Omega(g(n))$

$f(n) = O(g(n))$

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



e.g. $2n^2 = \Theta(n^2)$

$2n^2 = \Theta(100n^2)$

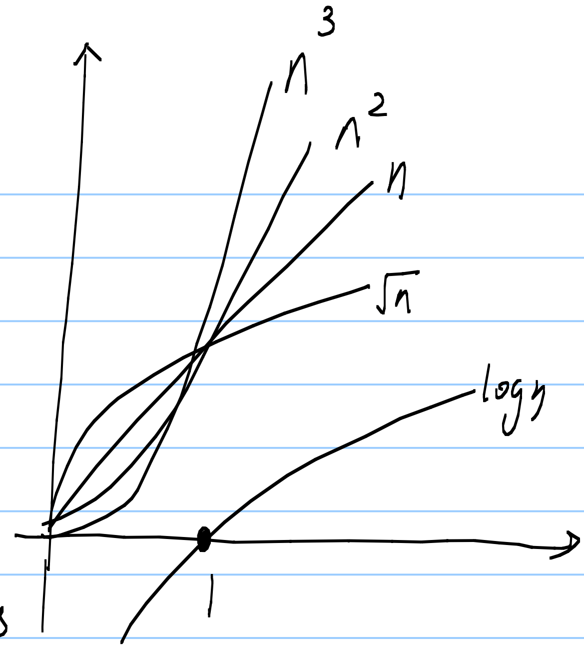
$n^3 = \Omega(n)$ $\quad n^3 = \Omega(n^2)$

$n^3 = \Omega(1)$ $\quad 1000n^2 = O(n^3)$

$2n^2 = O(n^4)$ $\quad 2n^2 = O(2^n)$

Facts you can use in your proof.

1. growth rate of different functions



n^k: polynomial function

$n, n^2, n^3, n^4 \ldots$

2^n: exponential function a^n

$n^3 \geq n^2 \geq n \geq \sqrt{n}$ when $n \geq 1$

order of growth of some commonly used functions

$1 \quad \log n \quad n \quad n\log n \quad n^2 \quad n^3 \quad 2^n$

2. derivative

monotonicity: a function $f(n)$ is monotonically increasing if $x \leq y$ implies $f(x) \leq f(y)$

$f(n) \leq c \cdot g(n) \Longleftrightarrow \quad c \cdot g(n) - f(n) \geqslant 0$

if $c \cdot g(x) - f(x) = 0$ when $n = x$

You need to show $c \cdot g(n) - f(n)$ is monotonically increasing in order to say $c\, g(n) - f(n) \geq 0$ when $n \geqslant x$

$(c \cdot g(n) - f(n))'$