Detail: when do we not recurse anymore?
when subarray has size 1

Running time for input of size n is T(n)
p=1, r=n

Mergesort(A,p,r)

if p<r then $q = \left\lfloor \frac{p+r}{2} \right\rfloor$ — $\Theta(1)$

Mergesort(A,p,q) — $T(\left\lceil \frac{n}{2} \right\rceil)$

Mergesort(A,q+1,r) — $T(\left\lfloor \frac{n}{2} \right\rfloor)$

Merge(A,p,q,r) — $\Theta(n)$

Running time for an input of size n is T(n)

$$T(n)=T(\left\lceil \frac{n}{2} \right\rceil) + T(\left\lfloor \frac{n}{2} \right\rfloor) + \Theta(n) + \Theta(1)$$

we will see later that this is $= 2\,T(n/2) + \Theta(n) = \Theta(n \lg n)$

Designing Algorithms

Insertion sort uses incremental approach

into a sorted A[1..j-1], insert 1 element A[j]

Today -- divide-and-conquer approach

Divide the problem into a number of subproblems

Conquer solve each subproblem recursively

Combine solutions to subproblems into a solution to original problem

Merge sort

Divide: divide n-element array A into 2 subarrays of n/2 elements each

Conquer: recursively sort the two subarrays

Combine: merge 2 sorted subarrays into 1 sorted array

Example: A = 5 2 4 6 1 3 2 6

1:

2: 2 4 5 6 1 2 3 6

3: A = Merge (2 4 5 6) (1 2 3 6) = 1 2 2 3 4 5 6 6

Intuition: look at the highest degree term, disregard other terms and the constant in front of the highest degree term

$1/2\ n^2 + 3n$ "becomes" n^2 $1/2\ n^2 - 3n$ "becomes" n^2

$1/2\ n^2 \lg n - 3n$ "becomes" $n^2 \lg n$ $5n^{2.5}$ "becomes" $n^{2.5}$ (not n^2)

Which of O, Θ, Ω should be put in place of the question mark?

$n^2 = ?(n)$ $n^2 = ?(n^3)$ $n^2 = ?(n^2)$

$n^2 \lg n = ?(n)$ $n^2 \lg n = ?(n^2)$ $n^2 \lg n = ?(n^3)$

A word on functions

for any constants a,b and c>1, as n approaches ∞ $\lg^a n \leq n^b \leq c^n$

even when a=10000 and b=0.00001

or b=10000 and c=1.00001

Thus, $\lg^a n = O(n^b)$ and $n^b = O(c^n)$

or equivalently, $\Omega(\lg^a n) = n^b$ and $\Omega(n^b) = c^n$

Example (formal): want to show that $1/2\ n^2 + 3n = O(n^2)$

$f(n)=1/2\ n^2 + 3n \qquad g(n)=n^2$

to show desired result, need c and n_0 such that $0 \le \frac{1}{2}n^2 + 3n \le cn^2$

try c=1

$\frac{1}{2}n^2 + 3n \le n^2 \qquad 3n \le \frac{1}{2}n^2 \qquad 6 \le n \qquad$ i.e. $n_0 = 6$

Example (formal): want to show that $1/2\ n^2 - 3n = \Theta(n^2)$

$f(n)=1/2\ n^2 - 3n \qquad g(n)=n^2$

to show desired result, need c_1, c_2 and n_0 such that

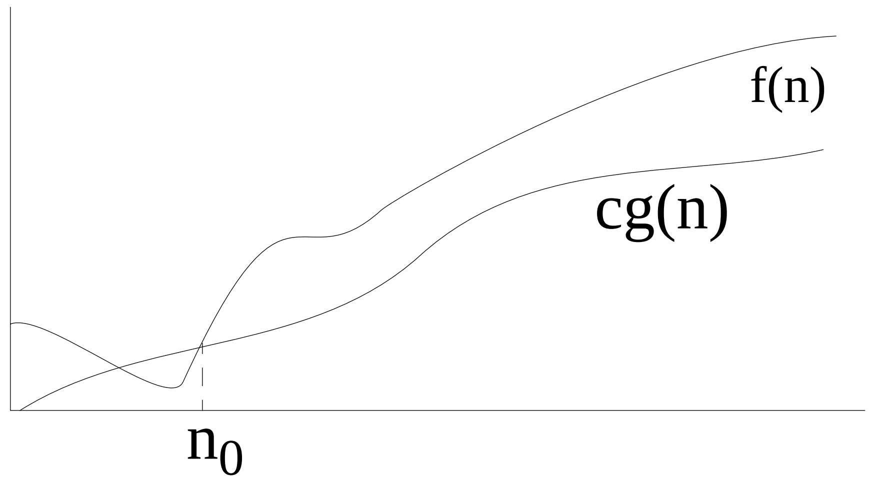
$$0 \le c_1 n^2 \le \frac{1}{2}n^2 - 3n \le c_2 n^2$$

dividing by n^2, we get $0 \le c_1 \le \frac{1}{2} - \frac{3}{n} \le c_2$

holds for $10 \le n$ and $c_2=1$ (right inequality)

holds for $10 \le n$ and $c_1= 0.2$ (left inequality) f(n) = Ω(g(n)) if there exist constants c and n_0 such that

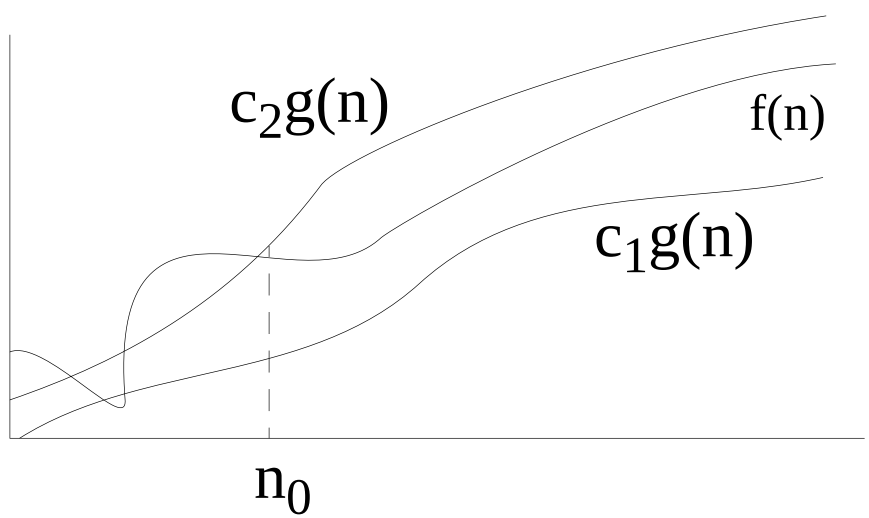
$$0 \le cg(n) \le f(n) \qquad \text{for all } n_0 \le n$$



Think of Ω as
a lower bound function

f(n) = Θ(g(n)) if there exist constants c_1, c_2 and n_0 such that

$$0 \le c_1 g(n) \le f(n) \le c_2 g(n) \quad \text{for all } n_0 \le n$$



Think of Θ as
both upper and lower
bound function

Lecture 2: Growth of Functions

Want to determine the order of growth for the worst-case running time

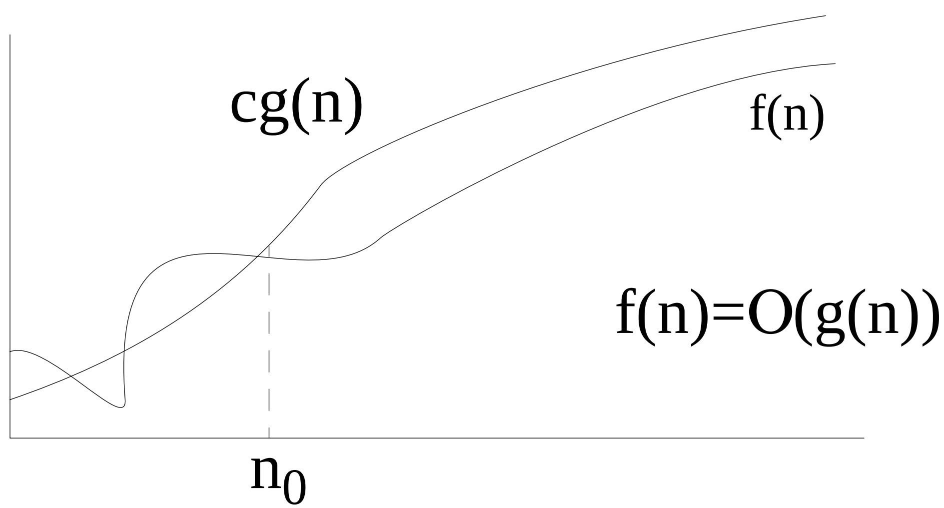
Ex: T(n)= $\left(\frac{c_4}{2}+\frac{c_5}{2}+\frac{c_6}{2}\right)n^2+\left(c_1+c_2+c_3+\frac{c_4}{2}-\frac{c_5}{2}-\frac{c_6}{2}+c_7\right)n$

$$-(c_2+c_3+c_4+c_7)$$

<u>Growth of Functions</u> (formal definitions)

f(n) = O(g(n)) if there exist constants c and n_0 such that

$$0 \le f(n) \le cg(n) \qquad \text{for all } n_0 \le n$$



Think of O as
upper bound function