CSE 326 Data Structures

CSE 326 : Dave Bacon

Asymptotic Analysis, Part II

Logistics

- Project 1 – Reverse a sound file
  - Due Wed January 10, 2007 at electronically at midnight
  - Look for (possible) new problem tonight (announced on mailing list) Announce on mailing list
  - Hard copy handed in Thursday in Section
- Homework 1 now online
  - Due Fri January 12, 2007 at beginning of lecture
- Reading (assume you finished Chapter 1,2,3)
  - Chapter 4 : Section 1,2, and 3 (trees, binary trees)
  - Chapter 6 : Priority Queues [next lecture]
- My Offices swichted to Tue 12:30-1:30 CSE 460

  4-5

Asymptotic Analysis

- Asymptotic analysis looks at the *order* of the running time of the algorithm
  - A valuable tool when the input gets "large"
  - Ignores the *effects of different machines* or *different implementations* of the same algorithm

- Intuitively, to find the asymptotic runtime, throw away the constants and low-order terms
  - Linear search is $T(n) = 3n + 2 \in \mathbf{O}(n)$
  - Binary search is $T(n) = 4 \log_2 n + 4 \in \mathbf{O}(\log n)$

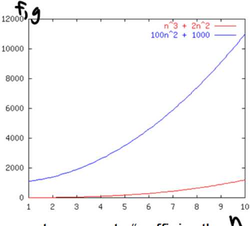
BIG O TIRES

Remember: the fastest algorithm has the slowest growing function for its runtime

Order Notation: Intuition

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

$g(n) = 100n^2 + 1000$



Although not yet apparent, as n gets "sufficiently large", $f(n)$ will be "greater than or equal to" $g(n)$

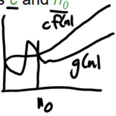
Order Notation: Definition

$5n = O(n)$

O(f(*n*)) : <u>a set or class of functions</u>

$5n \in O(n)$

g(*n*) ∈ O(f(*n*)) iff there exist consts c and n_0 such that:



$g(n) \le c\, f(n)$ for all $n \ge n_0$

Example: $g(n) = 1000n$ vs. $f(n) = n^2$

Is $g(n) \in O(f(n))$? $g(n) \in O(n^2)$

Pick: n0 = 1000, c = 1 $(1001)^2$

$1000n \le cn^2$ $n \ge n_0$

$1000n \le n^2$ $n_0 = 1000$

Notation Notes

Note: Sometimes, you'll see the notation:

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

This is equivalent to:

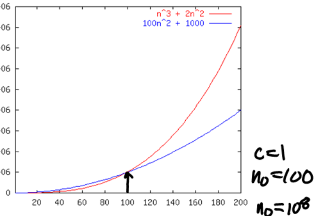
$g(n) \in O(f(n))$.

However: The notation ← Doom

$O(f(n)) = g(n)$ is meaningless!

(in other words big-O is not symmetric)

Order Notation: Example



$100n^2 + 1000 \leq 5\ (n^3 + 2n^2)$ for all $n \geq 19$

c=5
n0=19

So $f(n) \in O(\ g(n)\ )$

Big-O: Common Names

- $O(1)$ constant
- $O(\log_2 n)$ logaritmic
- $O(n)$ linear
- $O(n \log_2 n)$ log linear — fred
- $O(n^2)$ quadratic
- $O(n^3)$ cubic
- $O(n^k)$ (k is a constant) ← n^5, n^8, etc polynomial
- $O(c^n)$ (c is a constant > 1) ← exponential $\left(\frac{1}{2}\right)^n$

Log?

TRUE

$\log_k n \in O(\log_2 n)$?

$\log_A B = \frac{\log_C B}{\log_C A}$

$\log_k n = \frac{\log_2 n}{\log_2 k}$

$n_0 = 1$

$c = \frac{1}{\log_2 k}$

$\log_k n = \frac{\log_2 n}{\log_2 k} \leq c \log_2 n$

$\log_2 n^2 \in O(\log_2 n)$?

$\log AB = \log A + \log B$

$\log_2 n^2 = 2 \log_2 n \in O(\log_2 n)$

Definition of Order Notation

- Upper bound: $T(n) = O(f(n))$ Big-O $O(n^2)$

  Exist constants c and n' such that

  $T(n) \le c\, f(n)$ for all $n \ge n'$

- Lower bound: $T(n) = \Omega(g(n))$ Omega $\Omega(n^2)$

  Exist constants c and n' such that

  $T(n) \ge c\, g(n)$ for all $n \ge n'$

- Tight bound: $T(n) = \theta(f(n))$ Theta

  When both hold:

  $T(n) = O(f(n))$

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

Style

Reduce to most important term (drop constants)

$$n^3 + 105 \log n + 137 \log_{137} n$$

$$\in O(n^3 + 105 \log n)$$

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

$$5n^5 \in O(n^5)$$

Meet the Family

- $O(f(n))$ is the set of all functions asymptotically less than or equal to $f(n)$ $\leq$
  - $o(f(n))$ is the set of all functions asymptotically strictly less than $f(n)$ $<$
- $\Omega(f(n))$ is the set of all functions asymptotically greater than or equal to $f(n)$ $\geq$
  - $\omega(f(n))$ is the set of all functions asymptotically strictly greater than $f(n)$ $>$
- $\theta(f(n))$ is the set of all functions asymptotically equal to $f(n)$ $g(n) \in \theta(f(n))$ $g = f$

$f(n) = n$

$g(n) = 5n$

$5n \in O(n)$

$c = 5$

$5n \in \Omega(n)$

$c = 1$

$5n \leq 5n$ (c)

$5n \geq 1n$

$5 \geq 1$

Meet the Family, Formally

- $g(n) \in O(f(n))$ iff
  There exist c and n_0 such that $g(n) \le c\, f(n)$ for all $n \ge n_0$
  - $g(n) \in o(f(n))$ iff
    There exists a n_0 such that $g(n) < c\, f(n)$ for all c and $n \ge n_0$
    Equivalent to: $\lim_{n\to\infty} g(n)/f(n) = 0$
- $g(n) \in \Omega(f(n))$ iff
  There exist c and n_0 such that $g(n) \ge c\, f(n)$ for all $n \ge n_0$
  - $g(n) \in \omega(f(n))$ iff
    There exists a n_0 such that $g(n) > c\, f(n)$ for all c and $n \ge n_0$
    Equivalent to: $\lim_{n\to\infty} g(n)/f(n) = \infty$
- $g(n) \in \theta(f(n))$ iff
  $g(n) \in O(f(n))$ and $g(n) \in \Omega(f(n))$

Big-Omega et al. Intuitively

Asymptotic Notation	Mathematics Relation
O	$\leq$
Ω	$\geq$
θ	$=$
o	$<$
ω	$>$

Big O, Big Omega

- If $f(n) = \Omega(g(n))$ then $g(n) = O(f(n))$

$\exists$ constants, n_0, c, s.t.

$$g(n) \le c f(n) \quad \text{for } n \ge n_0$$

$\exists$ constants n_0', c', s.t

$$f(n) \ge c' g(n) \quad n \ge n_0'$$

$$\frac{f(n)}{c'} \ge g(n) \rightarrow g(n) \le \frac{1}{c'} f(n) \quad n \ge n_0'$$

$$c = \frac{1}{c'} \quad n_0 = n_0'$$

n^2 vs $n \log n$
n vs $\log n$

True or False?

$n \log n$ vs n
$\log n$ vs c

$< O(n^2)$

- (a) 3 n^2 + 10 n log n = O(n log n) F
- (b) 3 n^2 + 10 n log n = Omega(n^2) T
- (c) 3 n^2 + 10 n log n = Theta(n^2) T
- (d) n log n + n/2 = O(n) F
- (e) 10 SQRT(n) + log n = O(n) T
- (f) SQRT(n) + log n = O(log n) F
- (g) SQRT(n) + log n = Theta(log n) F
- (h) SQRT(n) + log n = Theta(n) Ω F
- (i) 2 SQRT(n) + log n = Theta(SQRT(n)) T
- (j) SQRT(n) + log n = Omega(1) T
- (k) SQRT(n) + log n = Omega(log n) T
- (l) SQRT(n) + log n = Omega(n) F

$n^{\frac{1}{2}}$ vs n
n vs n^2
$n^{\frac{1}{2}}$ vs $\log n$
n vs $\log^2 n$

Pros and Cons of Asymptotic Analysis

Pros

Con

← large data needed →

quick & dirty

separates
alg.
architecture

Which Function Grows Faster?

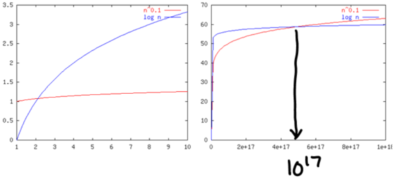
$n^{0.1}$ vs. $\log n$

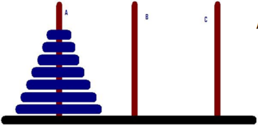
$n^{\frac{1}{10}}$ $\log n$

n vs $\log^{10} n$

Which Function Grows Faster?

$n^{0.1}$ vs. **log n**





ALGORITHM TH (n, A,B,C)

1. if n $\leq$ 0 **then return**

2. TH (n-1, A,C,B)

3. A ==> B

4. TH (n-1, C,B,A)

end

Office Hours

- Dave Bacon, Tu ~~4:00-5:00~~ 12:30-1:30, CSE 460
- Ruth Anderson, M 3:30-4:30, CSE 360

- Ethan Phelps-Goodman, Th 10:30-11:30, CSE 218
- Jonah Cohen, W 1:30-2:30, TBA

- David Wu, W 4:00-5:00 (in lab CSE 002/003), Th 3:30-4:30 (in CSE 218)