403: Algorithms and Data Structures

Asymptotic Notation

Fall 2016

UAlbany

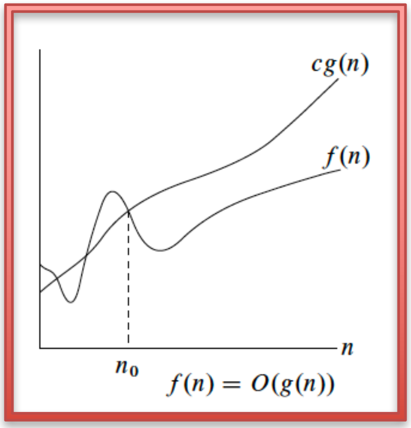
Computer Science

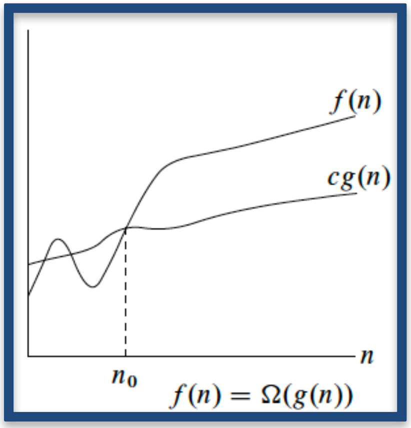
Assumptions

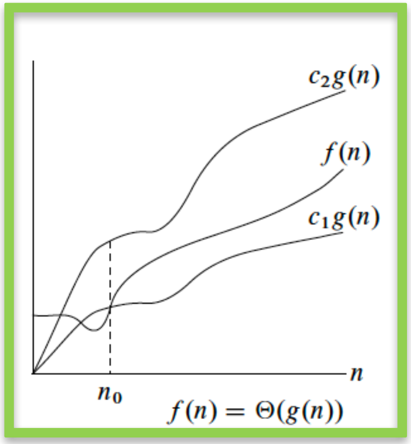
- All functions take non-negative values
- All functions are defined on non-negative integers
- Only such functions will be of interest

Overview of Asymptotic Notation

Notation	Informal statement	Formal definition	
$f = O(g)$	f grows no faster than g	$\exists n_0 > 0, c > 0$	s.t. $f(n) \leq cg(n), \forall n \geq n_0$
$f = \Omega(g)$	f grows at least as fast as g	$\exists n_0 > 0, c > 0$	s.t. $f(n) \geq cg(n), \forall n \geq n_0$
$f = \Theta(g)$	f grows at the same rate as g	$\exists n_0 > 0, c_1 > 0, c_2 > 0$	s.t. $c_1 g(n) \leq f(n) \leq c_2 g(n), \forall n \geq n_0$
$f = o(g)$	f grows *slower* than g	$\forall c > 0\ \exists n_0 > 0$	s.t. $f(n) \leq cg(n), \forall n \geq n_0$
$f = \omega(g)$	f grows *faster* than g	$\forall c > 0\ \exists n_0 > 0$	s.t. $f(n) \geq cg(n), \forall n \geq n_0$







Overview of Asymptotic Notation

Notation	Informal statement	Formal definition	
$g = O(f)$	g grows no faster than f	$\exists n_0 > 0, c > 0$	s.t. $g(n) \leq cf(n), \forall n \geq n_0$
$g = \Omega(f)$	g grows at least as fast as f	$\exists n_0 > 0, c > 0$	s.t. $g(n) \geq cf(n), \forall n \geq n_0$
$g = \Theta(f)$	g grows at the same rate as f	$\exists n_0 > 0, c_1 > 0, c_2 > 0$	s.t. $c_2 f(n) \leq g(n) \leq c_1 f(n), \forall n \geq n_0$
$g = o(f)$	g grows *slower* than f	$\forall c > 0\ \exists n_0 > 0$	s.t. $g(n) \leq cf(n), \forall n \geq n_0$
$g = \omega(f)$	g grows *faster* than f	$\forall c > 0\ \exists n_0 > 0$	s.t. $g(n) \geq cf(n), \forall n \geq n_0$

- o() is used to denote O() bound that is not asymptotically tight
  - E.g. we can verify that **$2n = O(n^2)$**, but it is not tight
  - We can denote this as **$2n = o(n^2)$**
  - …however **$2n^2$ is not $o(n^2)$**

Properties and Relations

Notation	Informal statement
$f = O(g)$	f grows no faster than g
$f = \Omega(g)$	f grows at least as fast as g
$f = \Theta(g)$	f grows at the same rate as g
$f = o(g)$	f grows *slower* than g
$f = \omega(g)$	f grows *faster* than g

- Θ *is a shorthand:* $f = O(g)$ and $f = \Omega(g) \iff f = \Theta(g)$

- o *is weaker than* O*:* $f = o(g) \Rightarrow f = O(g)$
- ω *is weaker than* Ω*:* $f = \omega(g) \Rightarrow f = \Omega(g)$

- *Sum:* $f_1 = O(g_1)$ and $f_2 = O(g_2) \Rightarrow f_1 + f_2 = O(g_1 + g_2)$ (same for other notations)
- *Product:* $f_1 = O(g_1)$ and $f_2 = O(g_2) \Rightarrow f_1 \times f_2 = O(g_1 \times g_2)$ (same for other notations)

- *Transitivity:* $f = O(g)$ and $g = O(h) \Rightarrow f = O(h)$ (same for other notations)

- *Symmetry* Θ*:* $f = \Theta(g) \iff g = \Theta(f)$
- *Transpose symmetry* O *and* Ω*:* $f = O(g) \iff g = \Omega(f)$
- *Transpose symmetry* o *and* ω*:* $f = o(g) \iff g = \omega(f)$

Read the rest of Chapter 3

- Comparing the growth of common functions

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	n^2	$n!$	$(\lg n)!$
$(\frac{3}{2})^n$	n^3	$\lg^2 n$	$\lg(n!)$	2^{2^n}	$n^{1/\lg n}$
$\ln \ln n$	$\lg^* n$	$n \cdot 2^n$	$n^{\lg \lg n}$	$\ln n$	1
$2^{\lg n}$	$(\lg n)^{\lg n}$	e^n	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$\lg^*(\lg n)$	$2^{\sqrt{2 \lg n}}$	n	2^n	$n \lg n$	$2^{2^{n+1}}$

Announcements

- Read through Chapter 3
- HW1 solutions available on BB