Big O, Ω, Θ

Joshua D. Guttman

Worcester Polytechnic Institute

30 October 2012

Orders of Growth: Big O

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in O(g)$$

means, for some N_0 and multiplicative constant c,
for every $n > N_0$

$$f(n) \leq c \cdot g(n)$$

Orders of Growth: Big O

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in O(g)$$

means, for some N_0 and multiplicative constant c,
for every $n > N_0$

$$f(n) \leq c \cdot g(n)$$

"g eventually bounds f, to within a multiplicative constant"

Orders of Growth: Ω

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Omega(g)$$

means, for some N_0 and multiplicative constant c,
for every $n > N_0$

$$c \cdot f(n) \geq g(n)$$

Orders of Growth: Ω

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Omega(g)$$

means, for some N_0 and multiplicative constant c,
for every $n > N_0$

$$c \cdot f(n) \geq g(n)$$

"f eventually bounds g, to within a multiplicative constant"

Orders of Growth: Ω

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Omega(g)$$

means, for some N_0 and multiplicative constant c,
for every $n > N_0$

$$c \cdot f(n) \geq g(n)$$

"f eventually bounds g, to within a multiplicative constant"

Ω is just the converse of O

$$f \in \Omega(g) \quad \text{means} \quad g \in O(f)$$

Orders of Growth: Θ

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Theta(g)$$

means, for some N_0 and multiplicative constants c_0, c_1,
for every $n > N_0$

$$f(n) \leq c_0 \cdot g(n) \quad \text{and} \quad g(n) \leq c_1 \cdot f(n)$$

Orders of Growth: Θ

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Theta(g)$$

means, for some N_0 and multiplicative constants c_0, c_1,
for every $n > N_0$

$$f(n) \leq c_0 \cdot g(n) \quad \text{and} \quad g(n) \leq c_1 \cdot f(n)$$

"f and g eventually agree, to within multiplicative constants"

Orders of Growth: Θ

If f, g are functions $\mathbb{R}^+ \to \mathbb{R}^+$, then

$$f \in \Theta(g)$$

means, for some N_0 and multiplicative constants c_0, c_1,
for every $n > N_0$

$$f(n) \leq c_0 \cdot g(n) \quad \text{and} \quad g(n) \leq c_1 \cdot f(n)$$

"f and g eventually agree, to within multiplicative constants"

$$f \in \Theta(g) \quad \text{is just} \quad f \in O(g) \text{ and } g \in O(f)$$

School multiplication

Input size: Number of digits in m and n (or bits)

Example: Input size 6

			2	3	8
			6	5	2
– –	– –	– –	– –	– –	– –
			4	7	6
	1	1	9	0	
1	4	2	8		
– –	– –	– –	– –	– –	– –
1	5	5	1	7	6

Some Properties of O, Θ (1)

O, Θ Reflexive:

$$f \in O(f), \Theta(f)$$

Some Properties of O, Θ (1)

O, Θ Reflexive:

$$f \in O(f), \Theta(f)$$

O, Θ Transitive:

$$f \in O(g) \quad \text{and} \quad g \in O(h) \quad \text{implies} \quad f \in O(h)$$

$$f \in \Theta(g) \quad \text{and} \quad g \in \Theta(h) \quad \text{implies} \quad f \in \Theta(h)$$

Some Properties of O, Θ (1)

O, Θ Reflexive:

$$f \in O(f), \Theta(f)$$

O, Θ Transitive:

$$f \in O(g) \quad \text{and} \quad g \in O(h) \quad \text{implies} \quad f \in O(h)$$

$$f \in \Theta(g) \quad \text{and} \quad g \in \Theta(h) \quad \text{implies} \quad f \in \Theta(h)$$

Θ Symmetric:

$$f \in \Theta(g) \quad \text{implies} \quad g \in \Theta(f)$$

Some properties of O, Θ (2)

O, Θ Additive

$$f, g \in O(h) \quad \text{implies} \quad f(x) + g(x) \in O(h(x))$$

$$f, g \in \Theta(h) \quad \text{implies} \quad f(x) + g(x) \in \Theta(h(x))$$

$$f(x) + g(x) \in \Theta(\max(f(x), g(x)))$$

Some properties of O, Θ (2)

O, Θ Additive

$$f, g \in O(h) \quad \text{implies} \quad f(x) + g(x) \in O(h(x))$$

$$f, g \in \Theta(h) \quad \text{implies} \quad f(x) + g(x) \in \Theta(h(x))$$

$$f(x) + g(x) \in \Theta(\max(f(x), g(x)))$$

O, Θ Preserved under scalar multiplication

$$f \in O(g) \quad \text{implies} \quad c \cdot f \in O(g)$$

$$f \in \Theta(g) \quad \text{implies} \quad c \cdot f \in \Theta(g)$$

Θ Equivalence Classes

- Θ gathers functions into equivalence classes

  $\Theta(g)$ gathers all the fns
  that grow about as fast as g

- Every fn f belongs to one class

  namely $\Theta(f)$

- If two classes overlap, they're identical

  $f \in \Theta(g)$ *and* $f \in \Theta(h)$
  implies $\Theta(h) = \Theta(g)$

Θ Equivalence Classes: Upper and Lower Bounds

Define $\max(f, g) = h_{\max}$ and
$\min(f, g) = h_{\min}$
one value at a time:

Θ Equivalence Classes: Upper and Lower Bounds

Define $\max(f, g) = h_{\max}$ and
$\min(f, g) = h_{\min}$

one value at a time:

$$h_{\max}(n) = \max(f(n), g(n))$$

$$h_{\min}(n) = \min(f(n), g(n))$$

Θ Equivalence Classes: Upper and Lower Bounds

Define $\max(f, g) = h_{\max}$ and
$\min(f, g) = h_{\min}$
one value at a time:

$$h_{\max}(n) = \max(f(n), g(n))$$

$$h_{\min}(n) = \min(f(n), g(n))$$

So

$$f, g \in \Omega(h_{\min}) \quad \text{and} \quad f, g \in O(h_{\max})$$

Θ Equivalence Classes: Upper and Lower Bounds

Define $\max(f, g) = h_{\max}$ and
$\min(f, g) = h_{\min}$
one value at a time:

$$h_{\max}(n) = \max(f(n), g(n))$$

$$h_{\min}(n) = \min(f(n), g(n))$$

So

$$f, g \in \Omega(h_{\min}) \quad \text{and} \quad f, g \in O(h_{\max})$$

$\Theta(h_{\min})$ is a greatest lower bound for $\Theta(f), \Theta(g)$
$\Theta(h_{\max})$ is a least upper bound for $\Theta(f), \Theta(g)$

Θ Equivalence Classes are Dense

Suppose that

$$f \in O(g) \quad \text{but} \quad f \notin \Theta(g)$$

Define $\text{mid}(f, g) = h$ one value at a time:

Θ Equivalence Classes are Dense

Suppose that

$$f \in O(g) \quad \text{but} \quad f \notin \Theta(g)$$

Define $\text{mid}(f, g) = h$ one value at a time:

$$h(n) = (f(n) \cdot g(n))^{1/2}$$

Θ Equivalence Classes are Dense

Suppose that

$$f \in O(g) \quad \text{but} \quad f \notin \Theta(g)$$

Define $\text{mid}(f, g) = h$ one value at a time:

$$h(n) = (f(n) \cdot g(n))^{1/2}$$

Then

$$f \in O(h) \quad \text{and} \quad h \in O(g)$$

But

$$f \notin \Theta(h) \quad \text{and} \quad h \notin \Theta(g)$$

Proof: $f \in O(h)$ and $h \in O(g)$

1. Suppose for all $x > N_0$, $f(x) \leq c \cdot g(x)$
2. Multiplying through by $f(x)$: $(f(x))^2 \leq c \cdot f(x) \cdot g(x)$
   taking square roots,

   $$f(x) \leq \sqrt{c} \cdot (f(x) \cdot g(x))^{1/2} = \sqrt{c} \cdot h(x)$$

   So $f \in O(h)$
3. Multiplying through by $g(x)$: $f(x) \cdot g(x) \leq c \cdot (g(x))^2$
   taking square roots,

   $$h(x) = (f(x) \cdot g(x))^{1/2} \leq \sqrt{c} \cdot g(x)$$

   So $h \in O(g)$

Proof: $f \notin \Theta(h)$

1. If $f \in \Theta(h)$, then for $x > N_0$:

$$(f(x) \cdot g(x))^{1/2} \leq c \cdot f(x)$$

2. Squaring:

$$f(x) \cdot g(x) \leq c^2 \cdot (f(x))^2$$

3. Dividing through:

$$g(x) \leq c^2 \cdot f(x)$$

4. Hence $g \in O(f)$, contrary to assumption

Proof: $h \notin \Theta(g)$

1. If $h \in \Theta(g)$, then for $x > N_0$:

$$g(x) \leq c \cdot (f(x) \cdot g(x))^{1/2}$$

2. Squaring,

$$(g(x))^2 \leq c^2 \cdot (f(x) \cdot g(x))$$

3. Dividing through

$$g(x) \leq c^2 \cdot f(x)$$

4. Hence $g \in O(f)$, contrary to assumption

Some Orders of Growth

$O(1)$ Vector ref

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search
$O(n)$	Linear search

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search
$O(n)$	Linear search
$O(n \log n)$	Merge sort

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search
$O(n)$	Linear search
$O(n \log n)$	Merge sort
$O(n^2)$	Insertion sort

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search
$O(n)$	Linear search
$O(n \log n)$	Merge sort
$O(n^2)$	Insertion sort
$\dots$	
$O(2^n)$	Largest independent set in a graph of size n

Some Orders of Growth

$O(1)$	Vector ref
$O(\log n)$	Binary search
$O(n)$	Linear search
$O(n \log n)$	Merge sort
$O(n^2)$	Insertion sort
$\dots$	
$O(2^n)$	Largest independent set in a graph of size n
$O(n!)$	Generate all permutations of n

Find the Order of Growth

An Exercise

- Multiply m, n by the school method
- Given an alg. $\mathcal{A}$, an input i, and a bound k, check if $\mathcal{A}$ halts on i before step k
- Factor n by trial division
- Given n and a claimed prime factorization, check correctness
- Given a formula, search for a proof of it in set theory

Has $\mathcal{A}$ terminated on i by step k?

Input size:	Total width w (in bits) of $\mathcal{A}$ and integers i, k
Worst case strategy:	Let k use most input bits, $k \sim 2^w$
Effect:	Often have to wait 2^w steps to see
So:	Exponential

Big-O, Big-Θ

- $f \in \Theta(g)$ clusters functions into equivalence classes
- Big-O determines a partial ordering
  - Greatest lower bounds and least upper bounds exist
  - Ordering is dense
- Big-O ordering: implementation-independent estimate of cost