

Properties and Application

Properties and Application

Asymptotic Notation ...

Mike Jacobson (University of Calgary)

• provides information about the *relative rates of growth* of a pair of functions (of a single integer or real variable)

Computer Science 331

- ignores or hides other details, including
 - behaviour on *small* inputs results are most meaningful when inputs are extremely *large*
 - multiplicative constants and lower-order terms which can be implementation or platform-dependent anyway
- permits classification of algorithms into classes (eg. linear, quadratic, polynomial, exponential, etc...)
- is useful for giving the kinds of bounds on running times of algorithms that we will study in this course

Types of Asymptotic Notation Big-Oh Notation

Computer Science 331

Big-Oh Notation

Mike Jacobson (University of Calgary)

Definition: Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$.

 $f \in O(g)$:

There exist constants c > 0 and $N_0 \ge 0$ such that

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

for all $n \geq N_0$.

Intuition:

- growth rate of f is at most (same as or less than) that of g
- Eg. $4n + 3 \in O(n)$ definition is satisfied using c = 5 and $N_0 = 3$
- sometimes written f = O(g) (also OK)

Lecture #7

Example: $4n^2 + 2 \in O(n^2)$



Types of Asymptotic Notation Big-Oh Notation

Proof that $4n^2 + 2 \in O(n^2)$

Theorem 1

 $4n^2 + 2 \in O(n^2)$

Proof.

Let $f(n) = 4n^2 + 2$ and $g(n) = n^2$. Then: • $f(n) = 4n^2 + 2 \le 4n^2 + n^2 = 5n^2$ whenever $n^2 \ge 2$ • $n^2 \ge 2$ holds if $n \ge \sqrt{2} \approx 1.414$ • $f(n) \le cg(n)$ for all $n \ge N_0$ when c = 5 and $N_0 = 2$. By definition, $f \in O(g)$ as claimed.

Note: this proof is *constructive* in that it determines the appropriate constants. Also OK to find constants by any means and simply prove that they satisfy the definition.

Computer Science 33

Mike Jacobson (University of Calgary)

Lecture #

Types of Asymptotic Notation Big-Omega Notation

Example: $n^2 \in \Omega(4n^2 + 2)$



Big-Omega Notation

Mike Jacobson (University of Calgary

Definition: Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$.

$f\in \Omega(g)$:

There exist constants c > 0 and $N_0 \ge 0$ such that

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

Computer Science 331

Types of Asymptotic Notation Big-Omega Notation

for all $n \geq N_0$.

Intuition:

- growth rate of f is at least (the same as or greater than) that of g
- $4n + 3 \in \Omega(n)$ definition is satisfied using $c = N_0 = 1$

Types of Asymptotic Notation Big-Omega Notation

Transpose Symmetry

Theorem 2

Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$. Then $f \in O(g)$ if and only if $g \in \Omega(f)$.

Proof.

If $f \in O(g)$:

- by defn $\exists c \in \mathbb{R}^{>0}$ and $N_0 \in \mathbb{R}^{\geq 0}$ such that $f(n) \leq cg(n)$ for all $n \geq N_0$.
- implies $cg(n) \ge f(n)$ for all $n \ge N_0$
- implies $g(n) \ge (1/c)f(n)$ for all $n \ge N_0$
- as $c\in \mathbb{R}^{>0},$ so is 1/c, so $g\in \Omega(f)$ by definition

```
If g \in \Omega(f), \ldots (exercise!)
```

Mike Jacobson (University of Calgary)

Computer Science 33

Types of Asymptotic Notation Big-Theta Notation

Example: $4n^2 + 2 \in \Theta(n^2)$



Types of Asymptotic Notation Big-Theta Notation

Big-Theta Notation

Definition: Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$.

 $f \in \Theta(g)$:

There exist constants $c_L, c_U > 0$ and $N_0 \ge 0$ such that

$$c_Lg(n) \leq f(n) \leq c_U \cdot g(n)$$

for all $n \geq N_0$.

Intuition:

- f has the same growth rate as g
- $4n + 3 \in \Theta(n)$ definition is satisfied using $c_L = 1, c_U = 5, N_0 = 3$

Mike Jacobson (University of Calgary)

Computer Science 331

Lecture #7 10 /

Types of Asymptotic Notation Big-Theta Notation

An Equivalent Definition

Theorem 3
Suppose $f,g:\mathbb{R}^{\geq 0} o\mathbb{R}^{\geq 0}.$ Then $f\in \Theta(g)$ if and only if
$f \in O(g)$ and $f \in \Omega(g)$

Exercise: Prove that the two definitions of " $f \in \Theta(g)$ " are *equivalent*.

How To Solve This:

• Work from the definitions, as in previous example!

Types of Asymptotic Notation Big-Theta Notation

A Common Mistake

People sometimes write "f is O(g)" (which is yet another way to write " $f \in O(g)$ " or "f = O(g)") when they actually mean " $f \in \Theta(g)$."

Please note that if $f \in O(g)$ then it is *not* necessarily true that $f \in \Theta(g)$ as well

• For example, as functions of $n, n \in O(n^2)$ but $n \notin \Theta(n^2)$.

So: If you want people to understand that " $f \in \Theta(g)$ " then this is what you should write!

Little-oh Notation

Definition: Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$.

 $f \in o(g)$:

For every constant c>0 there exists a constant $N_0\geq 0$ such that

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

for all $n \geq N_0$.

Intuition:

• f grows strictly slower than g

Big-Oh versus Little-Oh: notice how the constant *c* is quantified!



Example: $x \in \omega(\sqrt{x})$



Useful Properties

Suppose $f, g : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$.

Useful properties:

- $f \in o(g) \Rightarrow f \in O(g)$
- $f \in \omega(g) \Rightarrow f \in \Omega(g)$
- Transpose Symmetry: $f \in o(g) \iff g \in \omega(f)$
- Limit Test: $f \in o(g) \iff \lim_{x \to +\infty} \frac{f(x)}{g(x)} = 0$
- Limit Test:

Mike Jacobson (University of Calgary)

$$f \in \omega(g) \iff \lim_{x \to +\infty} \frac{f(x)}{g(x)} = +\infty$$

Useful Properties and Functions

Computer Science 331

Some Standard Functions

Mike Jacobson (University of Calgary

Polynomial (degree d): $p(n) = a_d n^d + a_{d-1} n^{d-1} + \dots + a_1 n + a_0$ • $p(n) \in \Theta(n^d)$

Exponentials: a^n , $a \in \mathbb{R}^{\geq 0}$ (increasing if a > 1)

• if a > 1, then $a^n \in \omega(p(n))$ for every polynomial p(n)

Logarithms: $\log_a n, a \in \mathbb{R}^{\geq 0}$

• $(\log_a n)^k \in o(p(n))$ whenever a > 1, $k \in \mathbb{R}^{\geq 0}$, and p(n) is a polynomial with degree at least one

Recommended Reading

Recommended Reading

Please read Section 4.1 and 4.2 of the textbook.

Chapter 3 of Cormen, Leiserson, Rivest and Stein's *Introduction to Algorithms* is also highly recommended.

Computer Science 331

Especially Useful in Introduction to Algorithms:

- Additional Properties and Exercises (pp. 49-50)
- Standard Notation and Common Functions (Section 3.2):
 - Floors and Ceilings
 - Modular Arithmetic
 - Standard Functions: Polynomials, Exponentials, Logarithms, and Their Properties

Lecture #7

17 / 20