CSE 15
Discrete Mathematics

Lecture 13 – Growth of Functions

Announcement

- HW #6 out
  - Due **5pm** 10/30 (Tues) with 1 extra day of re-submission.
  - **Write NEATLY!!!**
- Reading assignment
  - Ch. 4.1 – 4.3 of textbook

Some Important Points about Big-O Notation

- If one pair of witnesses is found, then there are infinitely many pairs.
- E.g., we can always make the k or the C larger and still maintain the inequality $|f(x)| \leq C|g(x)|$.
  - Any pair C' and k' where $C < C'$ and $k < k'$ is also a pair of witnesses since $|f(x)| \leq C|g(x)| \leq C'|g(x)|$ whenever $x > k' > k$.
- You may see "$f(x) = O(g(x))$" instead of "$f(x)$ is $O(g(x))$."
- It is ok to write $f(x) \in O(g(x))$, because $O(g(x))$ represents the set of functions that are $O(g(x))$.
- Usually, we will drop the absolute value sign since we will always deal with functions that take on positive values.

Using the Definition of Big-*O* Notation

Example: Show that $f(x) = x^2 + 2x + 1$ is $O(x^2)$.

Solution: Since when $x > 1$, $x < x^2$ and $1 < x^2$

$$0 \leq x^2 + 2x + 1 \leq x^2 + 2x^2 + x^2 = 4x^2$$

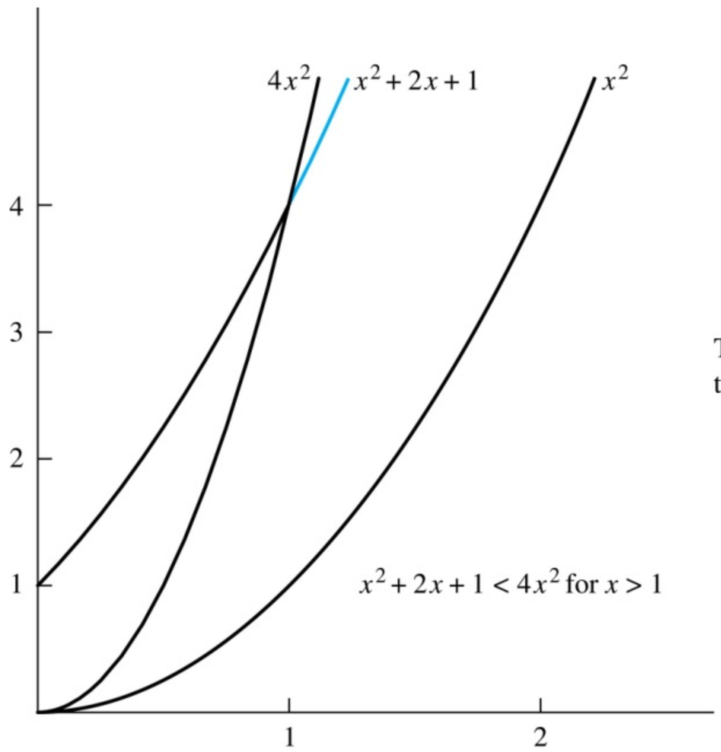
- Can take $C = 4$ and $k = 1$ as witnesses to show that

  $f(x)$ is $O(x^2)$ (see graph on next slide)

- Alternatively, when $x > 2$, we have $2x \leq x^2$ and $1 < x^2$. Hence, $0 \leq x^2 + 2x + 1 \leq x^2 + x^2 + x^2 = 3x^2$ when $x > 2$.
  - Can take $C = 3$ and $k = 2$ as witnesses instead.

Illustration of Big-*O* Notation

$f(x) = x^2 + 2x + 1$ is $O(x^2)$



The part of the graph of $f(x) = x^2 + 2x + 1$ that satisfies $f(x) < 4x^2$ is shown in blue.

Big-*O* Notation

- Both $f(x) = x^2 + 2x + 1$ and $g(x) = x^2$ are such that $f(x)$ is $O(g(x))$ and $g(x)$ is $O(f(x))$. We say that the two functions are of the *same order*. (More on this later)
- If $f(x)$ is $O(g(x))$ and *h(x)* is larger than *g(x)* for all positive real numbers, then $f(x)$ is $O(h(x))$.
  - Note that if $|f(x)| \leq C|g(x)|$ for *x* > *k* and if $|h(x)| > |g(x)|$ for all *x*, then $|f(x)| \leq C|h(x)|$ if *x* > *k*. Hence, $f(x)$ is $O(h(x))$.
- The goal is to select the function *g*(*x*) in *O(g(x))* as small as possible (up to multiplication by a constant, of course).

Big-*O* Estimates for Polynomials

Example: Let $f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_o$ where $a_0, a_1, \ldots, a_n$ are real numbers with $a_n \neq 0$.

Then $f(x)$ is $O(x^n)$.

Proof: $|f(x)| = |a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x^1 + a_0|$

$\leq |a_n| x^n + |a_{n-1}| x^{n-1} + \cdots + |a_1| x^1 + |a_0|$

Uses triangle inequality, an exercise in Section 1.8.

$= x^n (|a_n| + |a_{n-1}| /x + \cdots + |a_1|/x^{n-1} + |a_0|/ x^n)$

Assuming $x > 1$ $\leq x^n (|a_n| + |a_{n-1}| + \cdots + |a_1| + |a_0|)$

- Take $C = |a_n| + |a_{n-1}| + \cdots + |a_1| + |a_0|$ and $k = 1$. Then $f(x)$ is $O(x^n)$.
- **The leading term $a_n x^n$ of a polynomial dominates its growth.**

Big-*O* Estimates for some Important Functions

Example: Use big-*O* notation to estimate the sum of the first *n* positive integers.

Solution: $1 + 2 + \cdots + n \ \leq \ n + n + \cdots n \ = \ n^2$

$$1 + 2 + \ldots + n \text{ is } O(n^2) \text{ taking } C = 1 \text{ and } k = 1.$$

Example: Use big-*O* notation to estimate the factorial function

Solution: $f(n) = n! = 1 \times 2 \times \cdots \times n$.

$$n! = 1 \times 2 \times \cdots \times n \ \leq \ n \times n \times \cdots \times n \ = \ n^n$$

$$n! \text{ is } O(n^n) \text{ taking } C = 1 \text{ and } k = 1.$$

Continued →

Big-*O* Estimates for some Important Functions

Example: Use big-*O* notation to estimate $\log(n!)$

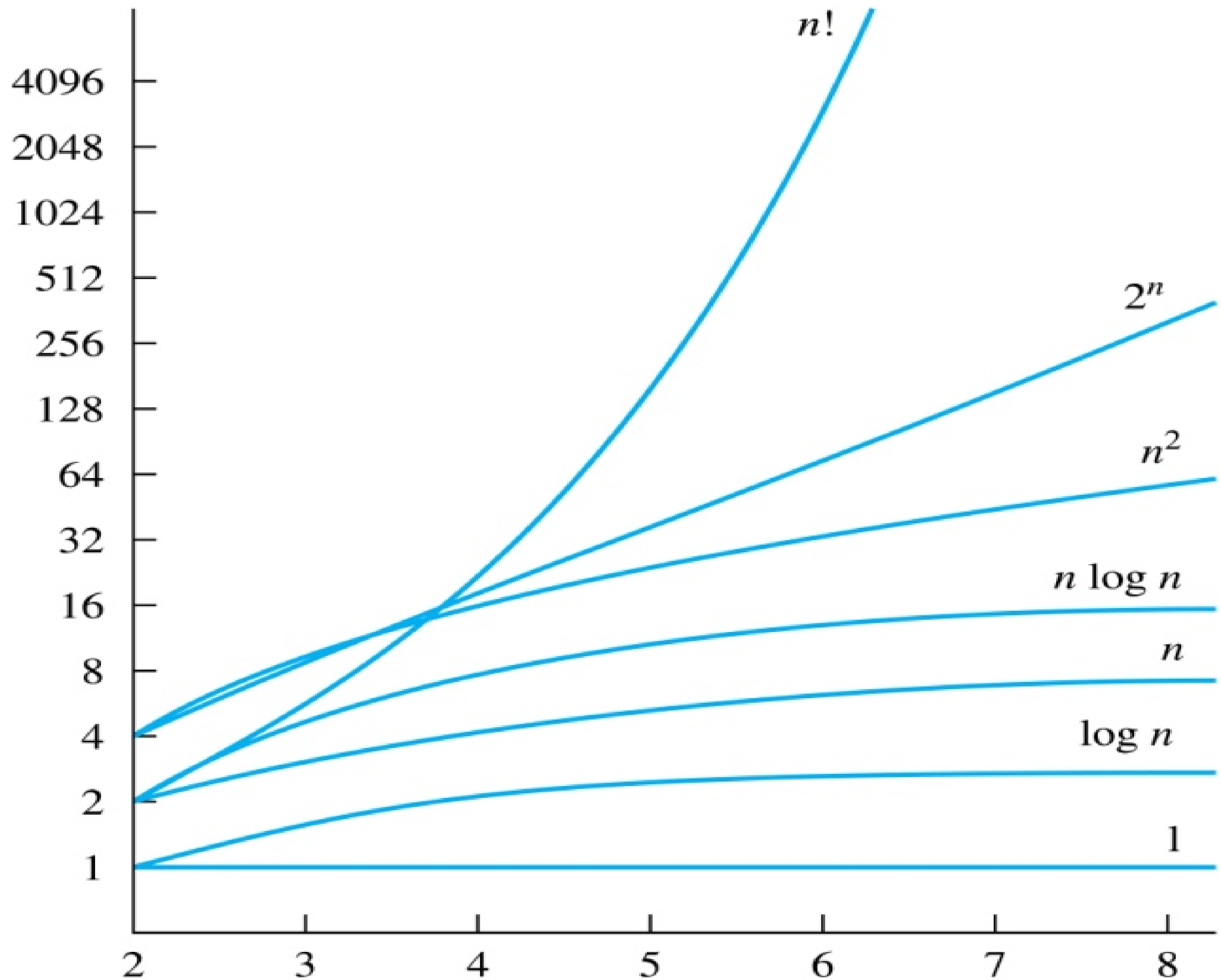
Solution: Given that $n! \leq n^n$ (previous slide)

then $\log(n!) \leq n \cdot \log(n)$.

Hence, $\log(n!)$ is $O(n \cdot \log(n))$ taking $C = 1$ and $k = 1$.

Display of Growth of Functions

Note the difference in behavior of functions as *n* gets larger



Combinations of Functions

- If $f_1(x)$ is $O(g_1(x))$ and $f_2(x)$ is $O(g_2(x))$ then
  $$(f_1 + f_2)(x) \text{ is } O(\max(|g_1(x)|, |g_2(x)|)).$$
  - See next slide for proof

- If $f_1(x)$ and $f_2(x)$ are both $O(g(x))$ then
  $$(f_1 + f_2)(x) \text{ is } O(g(x)).$$
  - See text for argument
  -

- If $f_1(x)$ is $O(g_1(x))$ and $f_2(x)$ is $O(g_2(x))$ then
  $$(f_1 f_2)(x) \text{ is } O(g_1(x)g_2(x)).$$
  - See text for argument

Combinations of Functions

- If $f_1(x)$ is $O(g_1(x))$ and $f_2(x)$ is $O(g_2(x))$ then
  $(f_1 + f_2)(x)$ is $O(\max(|g_1(x)|,|g_2(x)|))$.
  - By the definition of big-O notation, there are constants C_1, C_2, k_1, k_2 such that $|f_1(x)| \leq C_1|g_1(x)|$ when $x > k_1$ and $|f_2(x)| \leq C_2|g_2(x)|$ when $x > k_2$.

    $|(f_1 + f_2)(x)| = |f_1(x) + f_2(x)|$

    $\leq |f_1(x)| + |f_2(x)|$ by the triangle inequality $|a + b| \leq |a| + |b|$

    $|f_1(x)| + |f_2(x)| \leq C_1|g_1(x)| + C_2|g_2(x)|$

    $\leq C_1|g(x)| + C_2|g(x)|$ where $g(x) = \max(|g_1(x)|,|g_2(x)|)$

    $= (C_1 + C_2)|g(x)|$

    $= C|g(x)|$ where $C = C_1 + C_2$
  - Therefore $|(f_1 + f_2)(x)| \leq C|g(x)|$ whenever $x > k$, where $k = \max(k_1, k_2)$.

Big-Omega Notation

Ω is the upper case version of the lower case Greek letter ω.

Definition: Let *f* and *g* be functions from the set of integers or the set of real numbers to the set of real numbers. We say that $f(x)$ is $\Omega(g(x))$

if there are constants *C* and *k* such that

$|f(x)| \geq C|g(x)|$ when $x > k$.

- We say that "*f*(*x*) is big-Omega of *g*(*x*)."
- **Big-*O* gives an upper bound** on the growth of a function, while **Big-Omega gives a lower bound**. Big-Omega tells us that a function grows at least as fast as another.
- *f*(*x*) is $\Omega(g(x))$ if and only if *g*(*x*) is *O*(*f*(*x*)). This follows from the definitions. See the text for details.

Big-Omega Notation

Example: Show that $f(x) = 8x^3 + 5x^2 + 7$ is $\Omega(g(x))$
where $g(x) = x^3$.

Solution: $f(x) = 8x^3 + 5x^2 + 7 \geq 8x^3$ for all positive real numbers *x*.

- Is it also the case that $g(x) = x^3$ is $O(8x^3 + 5x^2 + 7)$?

Big-Theta Notation

Θ is the upper case version of the lower case Greek letter θ.

- **Definition**: Let *f* and *g* be functions from the set of integers or the set of real numbers to the set of real numbers. The function $f(x)$ is $\Theta(g(x))$ if $f(x)$ is $O(g(x))$ and $f(x)$ is $\Omega(g(x))$.

- $f(x)$ is $\Theta(g(x))$ if and only if there exists constants C_1, C_2 and k such that $C_1 g(x) < f(x) < C_2 g(x)$ if $x > k$. This follows from the definitions of big-*O* and big-Omega.

Big Theta Notation

Example: Show that the sum of the first n positive integers is $\Theta(n^2)$.

Solution: Let $f(n) = 1 + 2 + \cdots + n$.

- We have already shown that $f(n)$ is $O(n^2)$.
- To show that $f(n)$ is $\Omega(n^2)$, we need a positive constant C such that $f(n) > Cn^2$ for sufficiently large n. Summing only the terms greater than $n/2$ we obtain the inequality

$$
\begin{aligned}
1 + 2 + \cdots + n &\geq \lceil n/2 \rceil + (\lceil n/2 \rceil + 1) + \cdots + n \\
&\geq \lceil n/2 \rceil + \lceil n/2 \rceil + \cdots + \lceil n/2 \rceil \\
&= (n - \lceil n/2 \rceil + 1) \lceil n/2 \rceil \\
&\geq (n/2)(n/2) = n^2/4
\end{aligned}
$$

- Taking $C = ¼$, $f(n) > Cn^2$ for all positive integers n. Hence, $f(n)$ is $\Omega(n^2)$, and we can conclude that $f(n)$ is $\Theta(n^2)$.

Big-Theta Notation

Example: Show that $f(x) = 3x^2 + 8x \log x$ is $\Theta(x^2)$.

Solution:

- $3x^2 + 8x \log x \leq 11x^2$ for $x > 1$,
  since $0 \leq 8x \log x \leq 8x^2$.
  - Hence, $3x^2 + 8x \log x$ is $O(x^2)$.
- x^2 is clearly $O(3x^2 + 8x \log x)$ $\leftarrow$ O should be OMEGA
- Hence, $3x^2 + 8x \log x$ is $\Theta(x^2)$.

Big-Theta Estimates for Polynomials

Theorem: Let $f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_o$ where $a_0, a_1, \ldots, a_n$ are real numbers with $a_n \neq 0$.
Then $f(x)$ is of order x^n (or $\Theta(x^n)$).
(The proof is an exercise.)

Example:

The polynomial $f(x) = 8x^5 + 5x^2 + 10$ is order of x^5 (or $\Theta(x^5)$).

The polynomial $f(x) = 8x^{199} + 7x^{100} + x^{99} + 5x^2 + 25$ is order of x^{199} (or $\Theta(x^{199})$).