CS 141: Intermediate Data Structures and Algorithms

Discussion - Week 2, Winter 2018

TA information

- First/last name: Tin Vu
- E-mail: tvu032@ucr.edu (Please add the prefix [CS141] in your email's subject).
- Office hours: WCH 110

Thursday 1:00 PM - 3:00 PM

What should we do in discussion class?

- Review lecture’s content.
- Discuss about assignment.
- Do additional works.

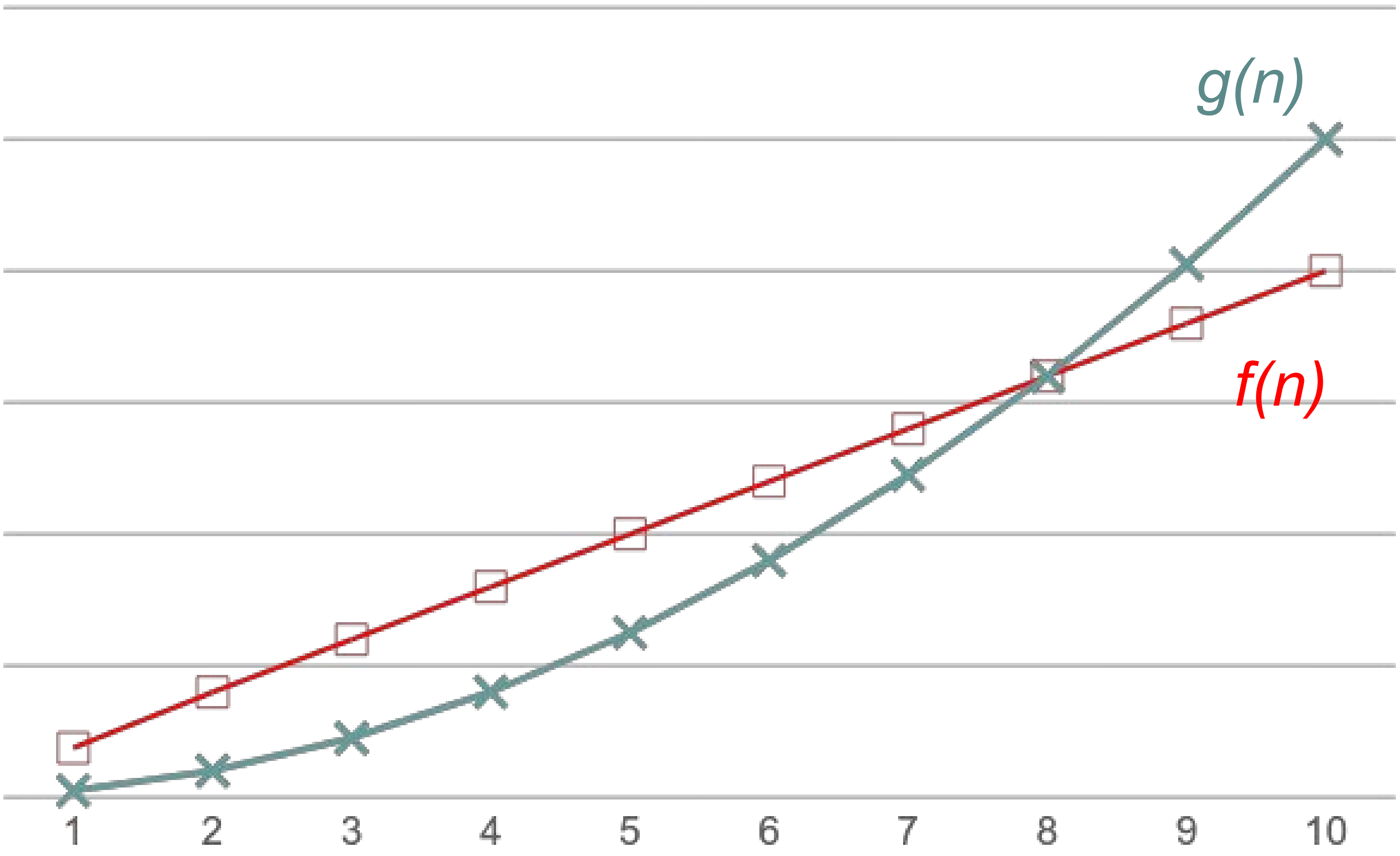
…

Do you have any other recommendation?

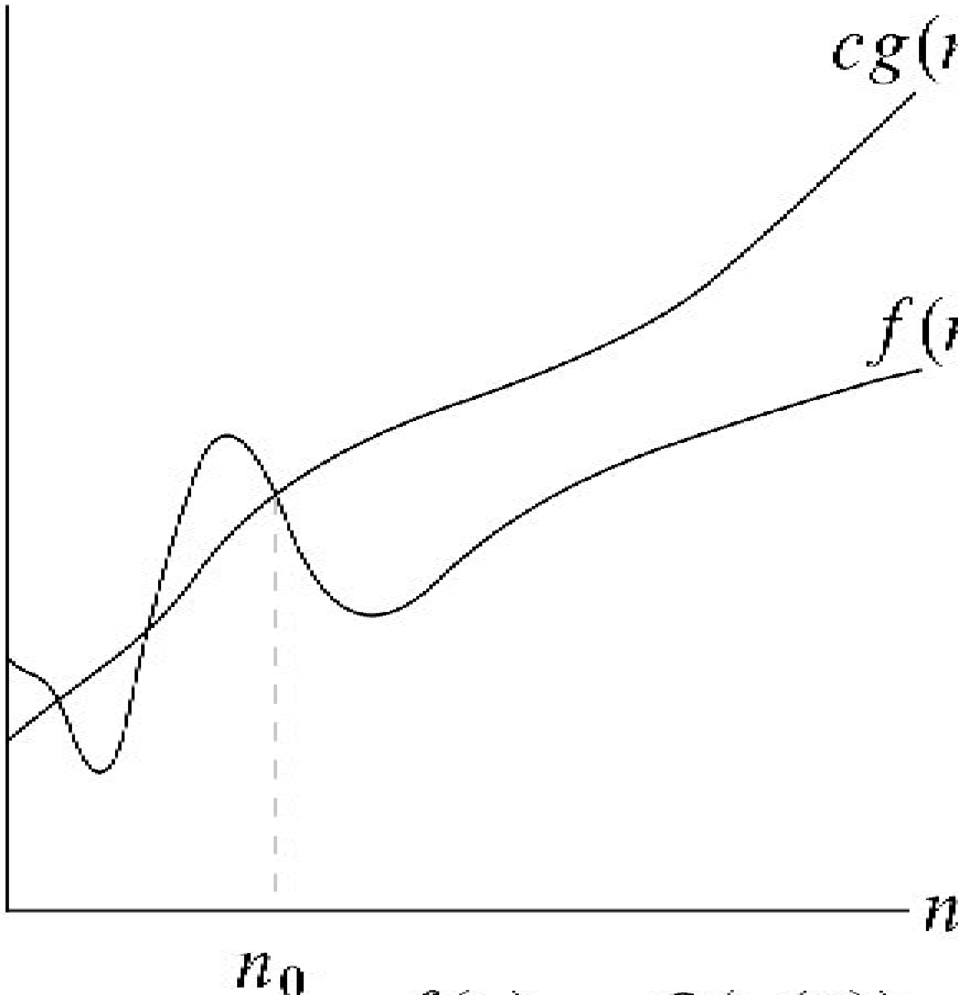
Analysis of Algorithms

- Analyzing Algorithms
  - Algorithm correctness.
  - Algorithm performance:
    - **Runtime analysis.**
    - Space analysis.

Growth of Functions



O-notation

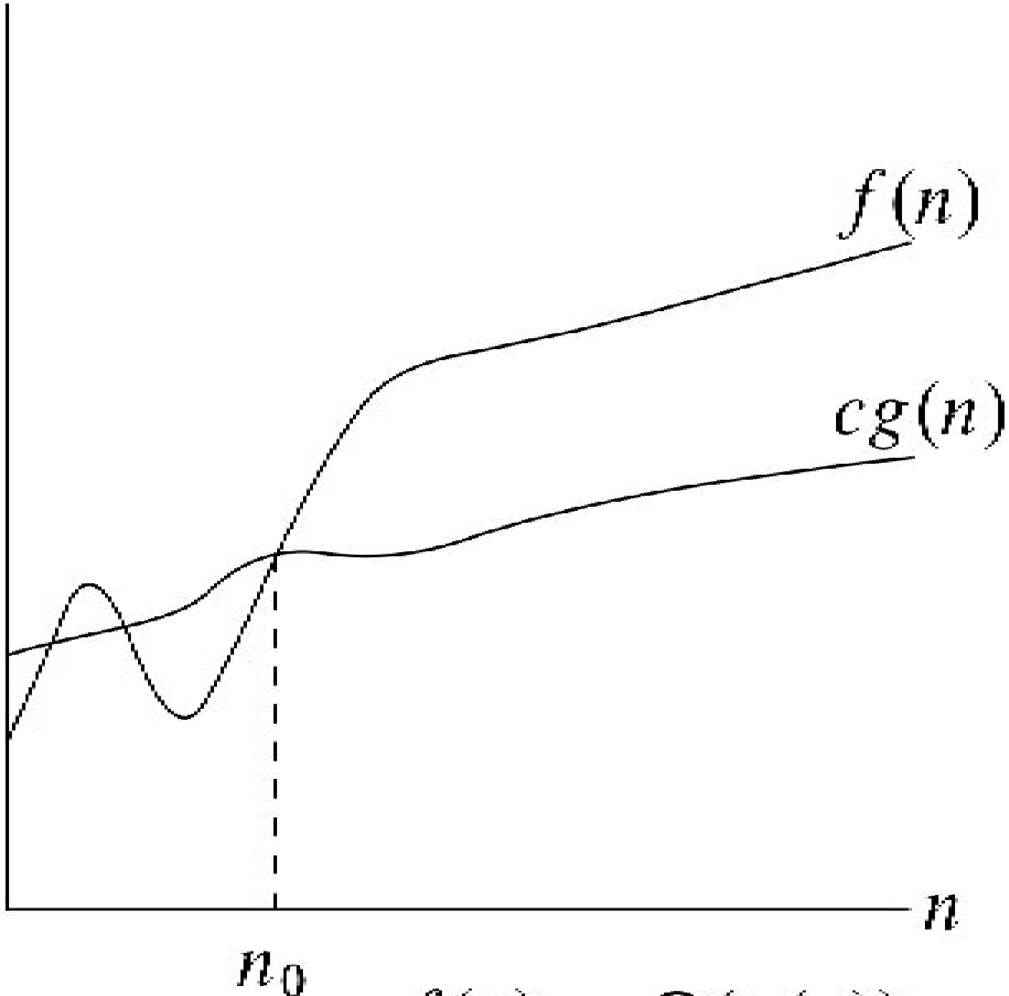


$f(n) = O(g(n))$

$\exists c > 0, n_0 > 0$
$0 \leq f(n) \leq cg(n)$
$n \geq n_0$

g(n) is an asymptotic upper-bound for f(n)

Ω-notation

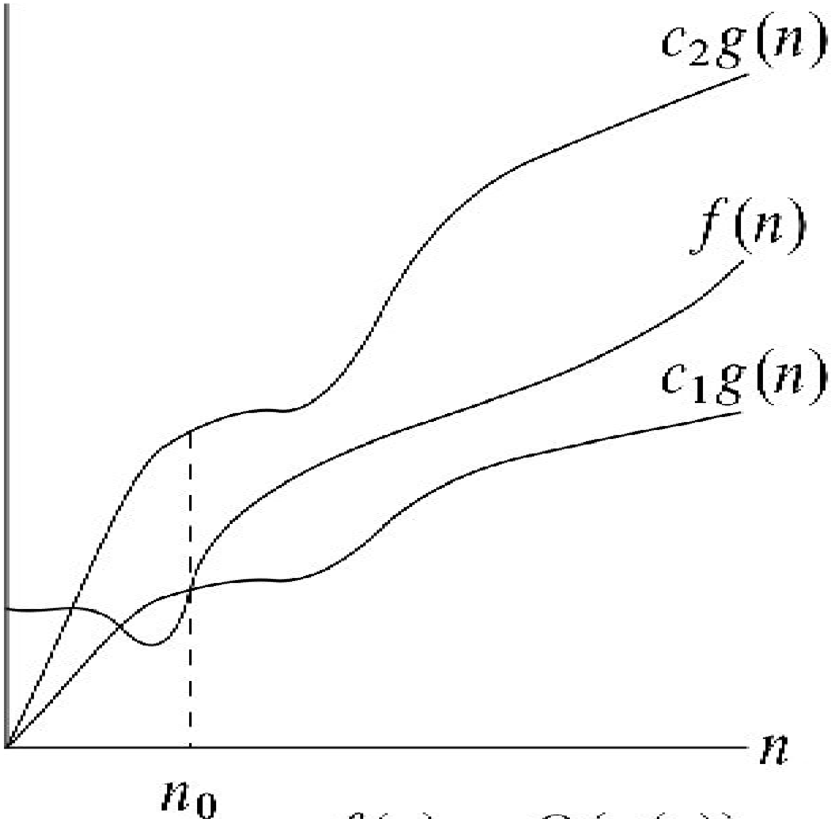


$f(n) = \Omega(g(n))$

$\exists c > 0, n_0 > 0$
$0 \leq cg(n) \leq f(n)$
$n \geq n_0$

g(n) is an asymptotic lower-bound for f(n)

Θ-notation

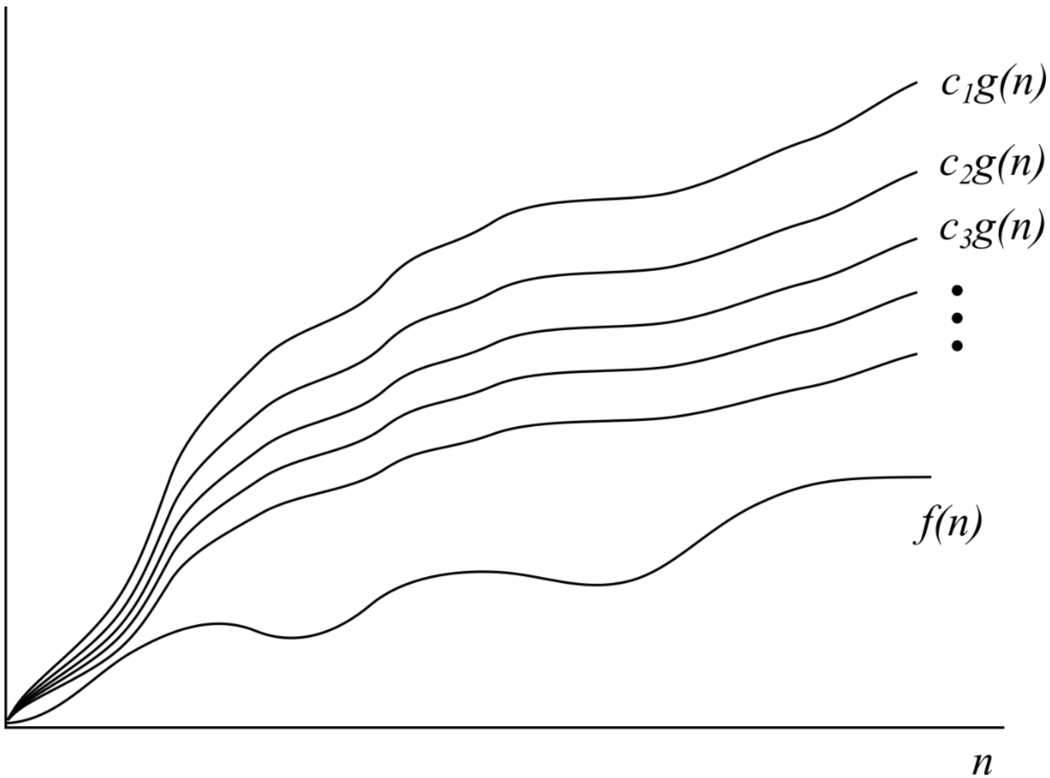


$f(n) = \Theta(g(n))$

$\exists c_1, c_2 > 0, n_0 > 0$
$0 \le c_1 g(n) \le f(n) \le c_2 g(n)$
$n \ge n_0$

g(n) is an asymptotic tight-bound for f(n)

o-notation

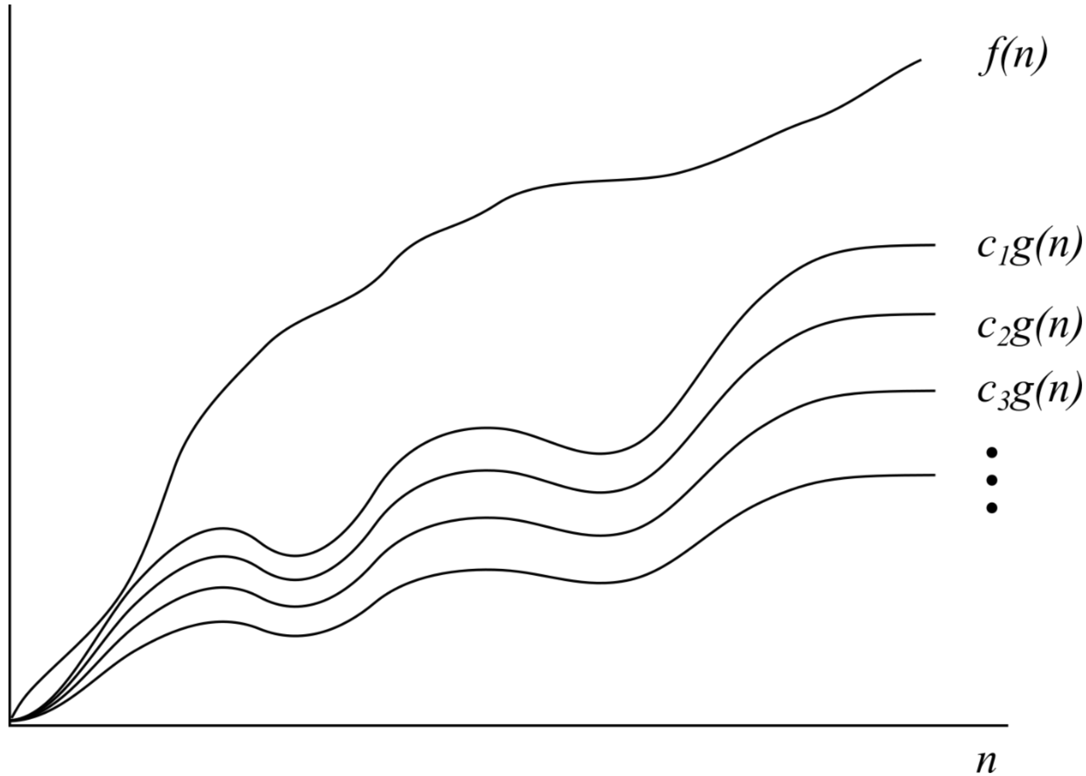


$$f(n) = o(g(n))$$

$$\forall c > 0$$
$$\exists n_0 > 0$$
$$0 \leq f(n) \leq cg(n)$$
$$n \geq n_0$$

g(n) is a non-tight asymptotic upper-bound for f(n)

ω-notation



$$f(n) = \omega(g(n))$$

$\forall c > 0$

$\exists n_0 > 0$

$0 \leq cg(n) \leq f(n)$

$n \geq n_0$

g(n) is a non-tight asymptotic lower-bound for f(n)

Discussion question

Is the following statement true or false?

$$2^n = \Theta(3^n)$$

Simple Rules

- We can omit constants
- We can omit lower order terms
- $\Theta(an^2+bn+c)$ becomes $\Theta(n^2)$
- $\Theta(c1)$ and $\Theta(c2)$ become $\Theta(1)$
- $\Theta(\log_{k1} n)$ and $\Theta(\log_{k2} n)$ become $\Theta(\log n)$
- $\Theta(\log(n^k))$ becomes $\Theta(\log n)$
- $\log^{k1}(n) = o(n^{k2})$ for any positive constants $k1$ and $k2$

Popular Classes of Functions

Constant: $f(n) = \Theta(1)$

Logarithmic: $f(n) = \Theta(\lg(n))$

Sublinear: $f(n) = o(n)$

Linear: $f(n) = \Theta(n)$

Super-linear: $f(n) = \omega(n)$

Quadratic: $f(n) = \Theta(n^2)$

Polynomial: $f(n) = \Theta(n^k)$; k is a constant

Exponential: $f(n) = \Theta(k^n)$; k is a constant

Comparing Two Functions

- $lim_{n \to \infty} \frac{f(n)}{g(n)}$
- 0: f(n) = o(g(n))
- c > 0: f(n) = Θ(g(n))
- ∞: f(n) = ω(g(n))

Discussion question

Solve a part of problem 4 - assignment 1.