CS2102 Analysis and Design of Algorithms

Divide and Conquer

M.Sc. Bryan Gonzales Vega
bgonzales.vega@gmail.com

University of Engineering and Technology

Lecture Content

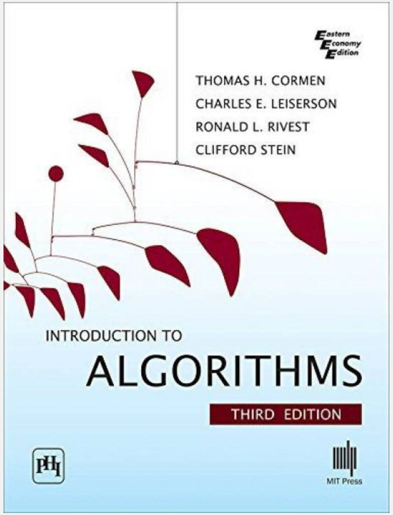
1. Recursion Tree
2. Master Method
3. Substitution Method

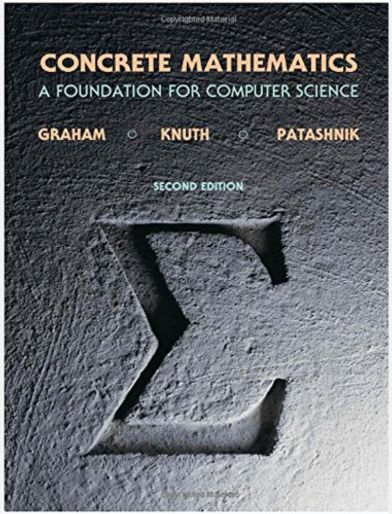
Introduction to Algorithms
[Cormen et al., 2009]

- Chap 02: Getting started
- Chap 03: Growth of functions
- Chap 04: Divide and conquer

Concrete Mathematics
[Graham et al., 1989]

- Chap 01: Recurrent problems
- Chap 02: Sums



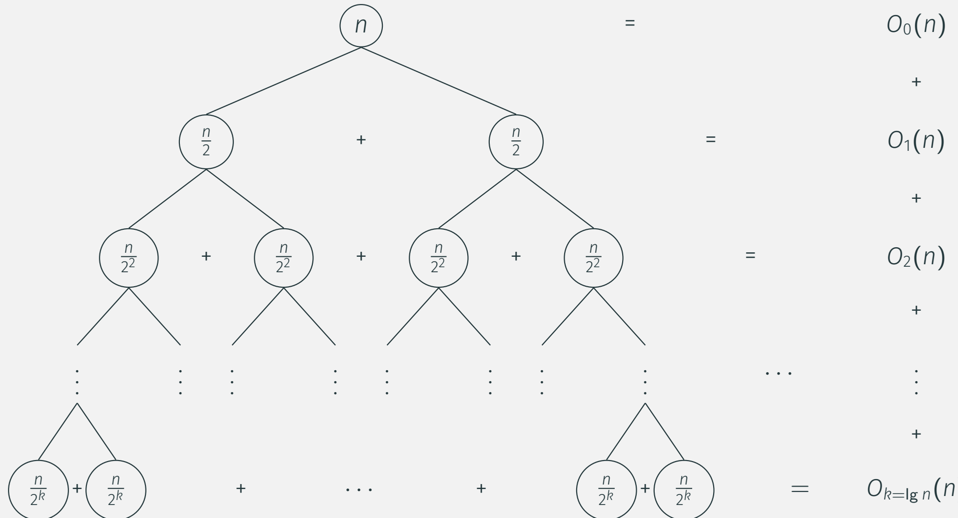


Recursion Tree

Merge Sort Recurrence

$$T(n) = 2T(n/2) + \Theta(n)$$

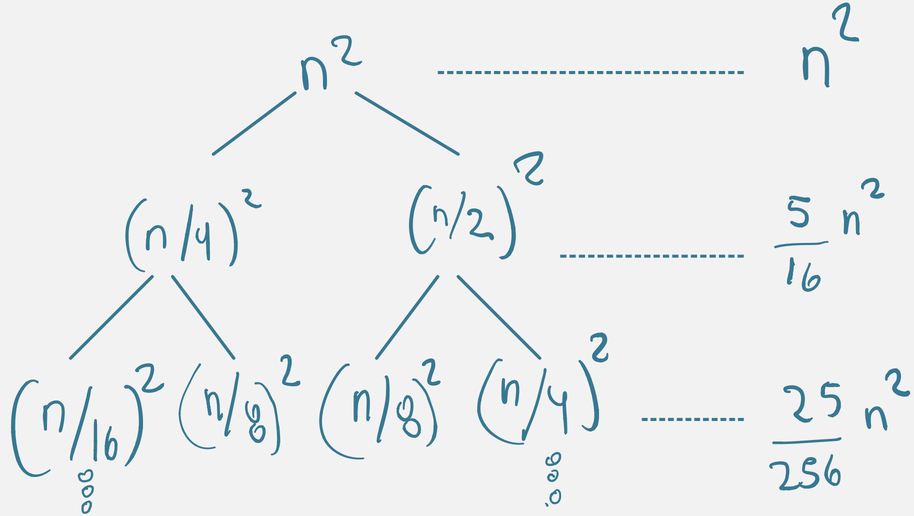
- Recursion tree method allow us to have an sketch of the behavior of the recurrence at hands.
- It helps us to propose a good assumption when using the substitution method.
- If we sketch a generalized recurrence we can deduce the master theorem/method.



$$\Downarrow$$

$$O\left(\sum_{i=0}^{k} 2^i \cdot \frac{n}{2^i}\right) = O\left(\sum_{i=0}^{k} n\right) = O(k \cdot n) \Leftrightarrow O(n \cdot \lg n)$$

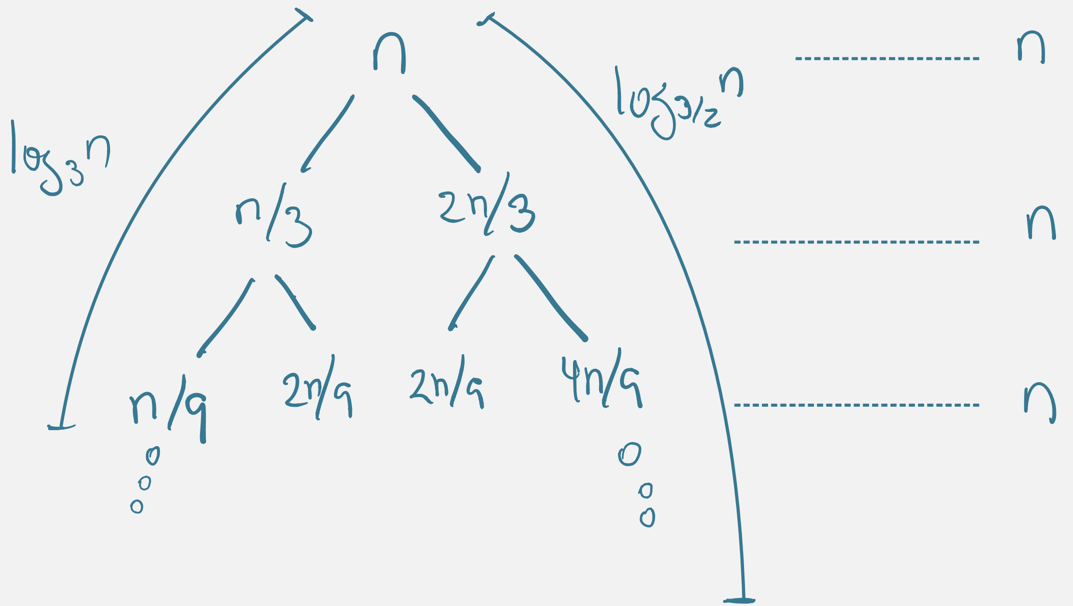
$$T(n) = T(n/4) + T(n/2) + n^2 \Rightarrow \Theta(n^2)$$



$$T(n) = n^2\left(1 + \frac{5}{16} + \dots \frac{5^k}{16^k} + \dots\right) \leq 2n^2 = \Theta(n^2)$$

Cont'd

$$T(n) = T(n/3) + T(2n/3) + n$$



$$O(n \log_{3/2} n),\ \Omega(n \log_3 n) = \Theta(n \log n) \;//$$

Cont'd

$$T(n) = 2T(n/2) + n^2$$

Dominates

n^2 .... n^2

$(n/2)^2$ $(n/2)^2$ .... $n^2/2$

$(n/4)^2$ $(n/4)^2$ $(n/4)^2$ $(n/4)^2$ .... $n^2/4$

...

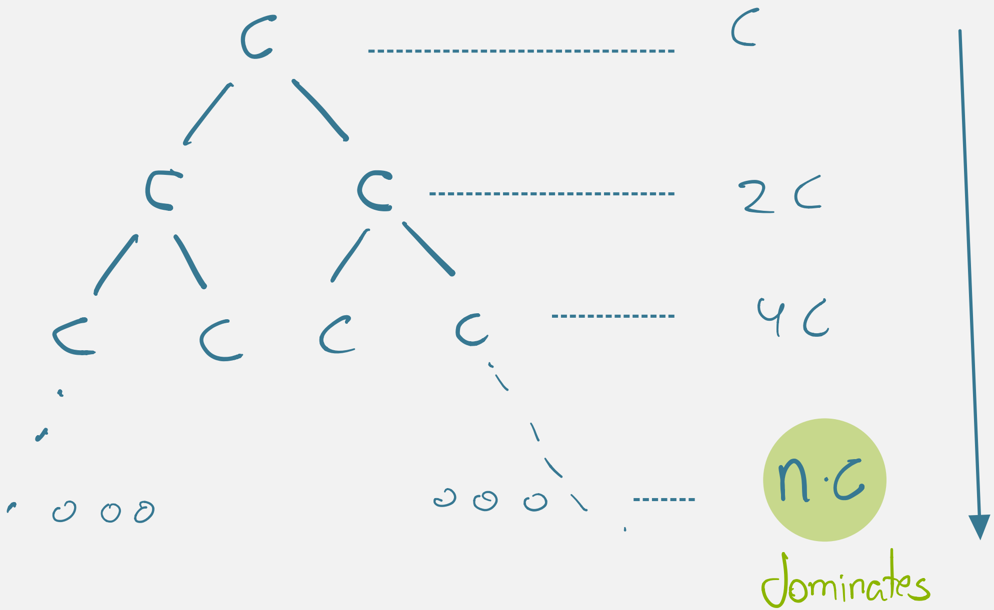
$$T(n) \le n^2 \sum_{i=0}^{k} \frac{1}{2^i}$$

$$\le 2n^2$$

$$\Theta(n^2)$$

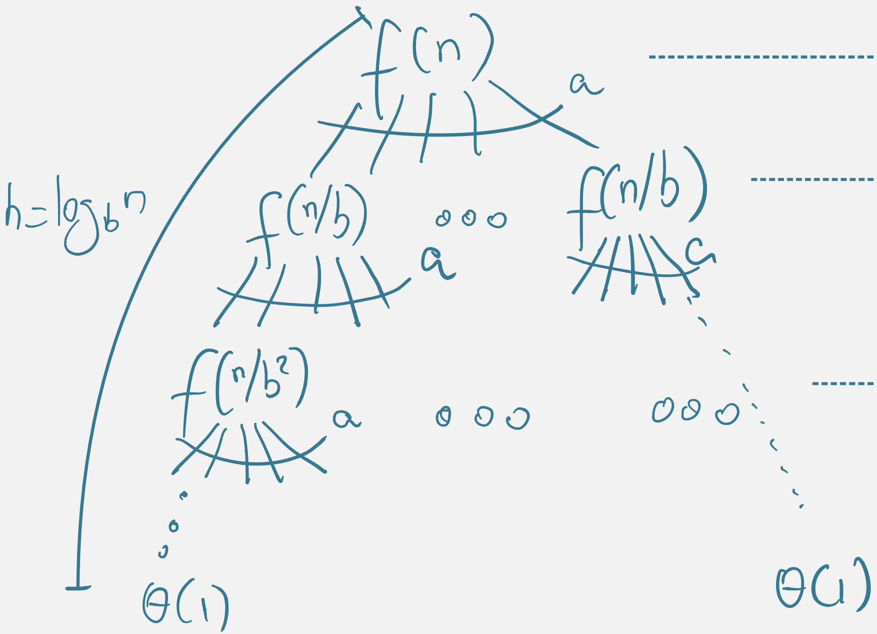
Cont'd

$$T(n) = 2T(n/2) + \Theta(1)$$



Cont'd

$$T(n) = aT(n/b) + f(n)$$



$f(n)$

$a f(n/b)$

$a^2 f(n/b^2)$

$\Theta(n^{\log_b a})$

$f(n)$ dominates

or

$f(n)$, $n^{\log_b a}$ are equivalent

or

$n^{\log_b a}$ dominates

Master Method

Let $T(n)$ be a monotonically increasing function that satisfies:

$$T(n) = aT(n/b) + f(n)$$
$$T(1) = c$$

where $a \geq 1, b \geq 2, c > 0$, then

Master Theorem [1]

$$T(n) = \begin{cases} \Theta(n^{\log_b a}), & f(n) = \mathcal{O}(n^{\log_b a - \epsilon}), \epsilon > 0 \\ \Theta(n^{\log_b a} \log n), & f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)) & f(n) = \Omega(n^{\log_b a + \epsilon}), \epsilon > 0 \end{cases} \quad (1)$$

[1] Master Method is deduced from the recurrence tree method

If $f(n) \in \Theta(n^d)$, $d \geq 0$:

- $\Theta(n^{\log_b a})$, if $d < \log_b a$
- $\Theta(n^{\log_b a} \log n)$, if $d = \log_b a$
- $\Theta(n^d)$, if $d > \log_b a$

Let $T(n)$ be a monotonically increasing function that satisfies:

$$T(n) = aT(n/b) + f(n)$$
$$T(1) = c$$

where $a \geq 1, b \geq 2, c > 0$, then

Master Theorem [1]

$$T(n) = \begin{cases} \Theta(n^{\log_b a}), & f(n) = \mathcal{O}(n^{\log_b a - \epsilon}), \epsilon > 0 \\ \Theta(n^{\log_b a} \log n), & f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)) & f(n) = \Omega(n^{\log_b a + \epsilon}), \epsilon > 0 \end{cases} \tag{1}$$

$T(n) = 4T(n/2) + n$ $\Theta(n^2)$

$T(n) = 4T(n/2) + n^2$ $\Theta(n^2 \log n)$

$T(n) = 4T(n/2) + n^3$ $\Theta(n^3)$

$T(n) = 2T(n/3) + n \log n$ $\Theta(n \log n)$

$T(n) = 9T(n/3) + n$ $\Theta(n^2)$

$T(n) = T(2n/3) + 1$ $\Theta(1 \cdot \log n)$

$T(n) = 2T(n/2) + n^4$ $\Theta(n^4)$

$T(n) = 3T(n/4) + n \log n$ $\Theta(n \log n)$

$T(n) = 4T(n/2) + n^2\sqrt{n}$ $\Theta(n^{2.5})$

$T(n) = 4T(n/3) + n \log n$ $\Theta(n^{\log_3 4})$

[1] Master Method is deduced from the recurrence tree method

Substitution Method

Given $T(n) = 4T(n/2) + n$
1. Guess solution of recurrence
2. Try mathematical induction
3. Close induction to prove initial guess

$T(n) = \mathcal{O}(n^2)$ **guess**

$T(k) \leq ck^2, k < n$ **assume**

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &\leq 4c(n/2)^2 + n \\ &\leq 4cn^2/4 + n \\ &\leq cn^2 + n \\ &\leq cn^2 - (-n) \end{aligned}$$

doesn't complete the induction

$T(n) = \mathcal{O}(n^2)$ **guess**

$T(k) \leq c_1k^2 - c_2k, k < n$ **assume**

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &\leq 4[c_1(n/2)^2 - c_2(n/2)] + n \\ &\leq 4c_1(n^2/4) - 4c_2(n/2) + n \\ &\leq c_1n^2 - 2c_2n + n \\ &\leq c_1n^2 - c_2n - n(c_2 - 1) \end{aligned}$$

completes induction when $c_2 \geq 1$

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009).
Introduction to algorithms.
MIT press.

Graham, R., Graham, R., Knuth, D., Knuth, D., and Patashnik, O. (1989).
Concrete Mathematics: A Foundation for Computer Science.
A foundation for computer science. Addison-Wesley.

Cont’d