CSE 373/L-5/20.02.2024/

⊛ O - notation

$O(g(n)) = \{ f(n)$: there exist $\boxed{\text{positive constants } c \text{ and } n_0}$ such that

$$0 \le f(n) \le c\, g(n) \text{ for all } n \ge n_0 \}$$

⇒ we say that "$f(n)$ is big-O of $g(n)$"

As n increases, $f(n)$ grows no faster than $g(n)$

- $g(n)$ asymptotically upper bound for $f(n)$

⊛ O-notation Example:

$$7n^3 + 100n^2 - 20n + 6$$

- Highest-order term: $7n^3$
- function's rate of growth: n^3
- function grows no faster than n^3

⇒ $O(n^3)$, $O(n^4)$, $O(n^5)$ ...

⇒ $O(n^c)$ for any constant $c \ge 3$

$\circledast$ $4n^2 + 100n + 500 = O(n^2)$ Prove it

$\Rightarrow$ we need to find positive constant c and n_0 such that

$$4n^2 + 100n + 500 \leq cn^2 \text{ for all } n \geq n_0$$

$$\Rightarrow 4 + \frac{100}{n} + \frac{500}{n^2} \leq c$$

Here,

$$\left.\begin{array}{l} n_0 = 1 \; ; \; c = 604 \\ n_0 = 10 \; ; \; c = 19 \\ n_0 = 100 \; ; \; c = 5.05 \end{array}\right\} \text{Positive constant}$$

Proved

$\circledast$ $n^2 + n = O(n^3)$

$$\Rightarrow n^2 + n \leq cn^3 \quad ; \quad n \geq n_0$$

We know,

$$a \leq b \; ; \quad n^a \leq n^b \quad ; \; n \geq 1$$

if $n \geq 1$,

$$n \leq n^3 \;\&\; n^2 \leq n^3$$

Therefore,

$$n^2 + n \leq n^3 + n^3 = 2n^3$$

$$\Rightarrow n^2 + n \leq 2n^3$$

Here,

$$n_0 = 1 \; ; \; c = 2$$

$$\Rightarrow n^2 + n \leq cn^3$$

$$\frac{1}{n} + \frac{1}{n^2} \leq c$$

Here,

$$n_0 = 1 \; ; \; c = 2$$

$n^3 - 100n^2 \neq O(n^2)$

$\Rightarrow$ Let,

$n^3 - 100n^2 = O(n^2)$ ...(i)

$\Rightarrow n^3 - 100n^2 \leq cn^2 \quad ; n \geq n_0$

$\Rightarrow n - 100 \leq c$

Here

$n_0 = 1 \; ; \; c = -99 \Rightarrow$ negative

Therefore (i) is not true.

$\therefore n^3 - 100n^2 \neq O(n^2)$ Proved)

Ω- notation

$\Omega(g(n)) = \{ f(n)$: there exist positive constant c and n_0 such that $0 \leq cg(n) \leq f(n)$ for all $n \geq n_0 \}$

- we say that " $f(n)$ is big-omega of $g(n)$"
- as n increases, $f(n)$ grows no slower than $g(n)$
- $g(n)$ is ~~asymto~~ asymptotically lower bound for $f(n)$

Ω- notation Example:

$\Rightarrow 7n^3 + 100n^2 - 20n + 6$

- highest order term: $7n^3$
- function rate of growth: n^3
- the function grows no slower than n^3

$= \Omega(n^3), \Omega(n^2), \Omega(n), \Omega(n^{0.5}) \ldots$

$= \Omega(n^c)$ for any constant $c \leq 3$

* $4n^2 + 100n + 500 = \Omega(n^2)$ Prove it

$\Rightarrow$ we need to find positive constants c and n_0 such that

$$4n^2 + 100n + 500 \geq cn^2 \text{ for all } n \geq n_0$$

$$\Rightarrow 4 + \frac{100}{n} + \frac{500}{n^2} \geq c$$

for any positive number $c \geq 4$

$\Rightarrow c = 4$

$\boxed{\text{Slide} - 12, 13}$

* Θ - notation

$\Theta(g(n)) = \{ f(n)$: there exist positive constants c_1, c_2, and n_0 such that

$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$ for all $n \geq n_0$

- We say that, " $f(n)$ is big-theta of $g(n)$"
- As n increases, $f(n)$ grows at the same rate as $g(n)$
- $g(n)$ is asymptotically tight bond for $f(n)$

c	n
$\log_2 n$	$n\log_2 n$
$\sqrt{n}$	$n^{1.1}$
n	n^2
$n\log_2 n$	n^3
	n^4
	2^n

* Θ - notation example:

$\Rightarrow 7n^3 + 100n^2 - 20n + 6$

- highest order term : $7n^3$
- function rate of growth : n^3
- the function is both $O(n^3)$ and $\Omega(n^3)$

$\Rightarrow \Theta(n^3)$

$\boxed{\text{Example, Slide} - 16 -}$ $\boxed{\text{Slide} - 18 - 25}$

L-6 / 25.02.2024

Chapter-4

Divide and Conquer

* Three method

(i) Recursion Tree Method

(ii) Substitution method

(iii) Master Method

*

$T(1) = 4$

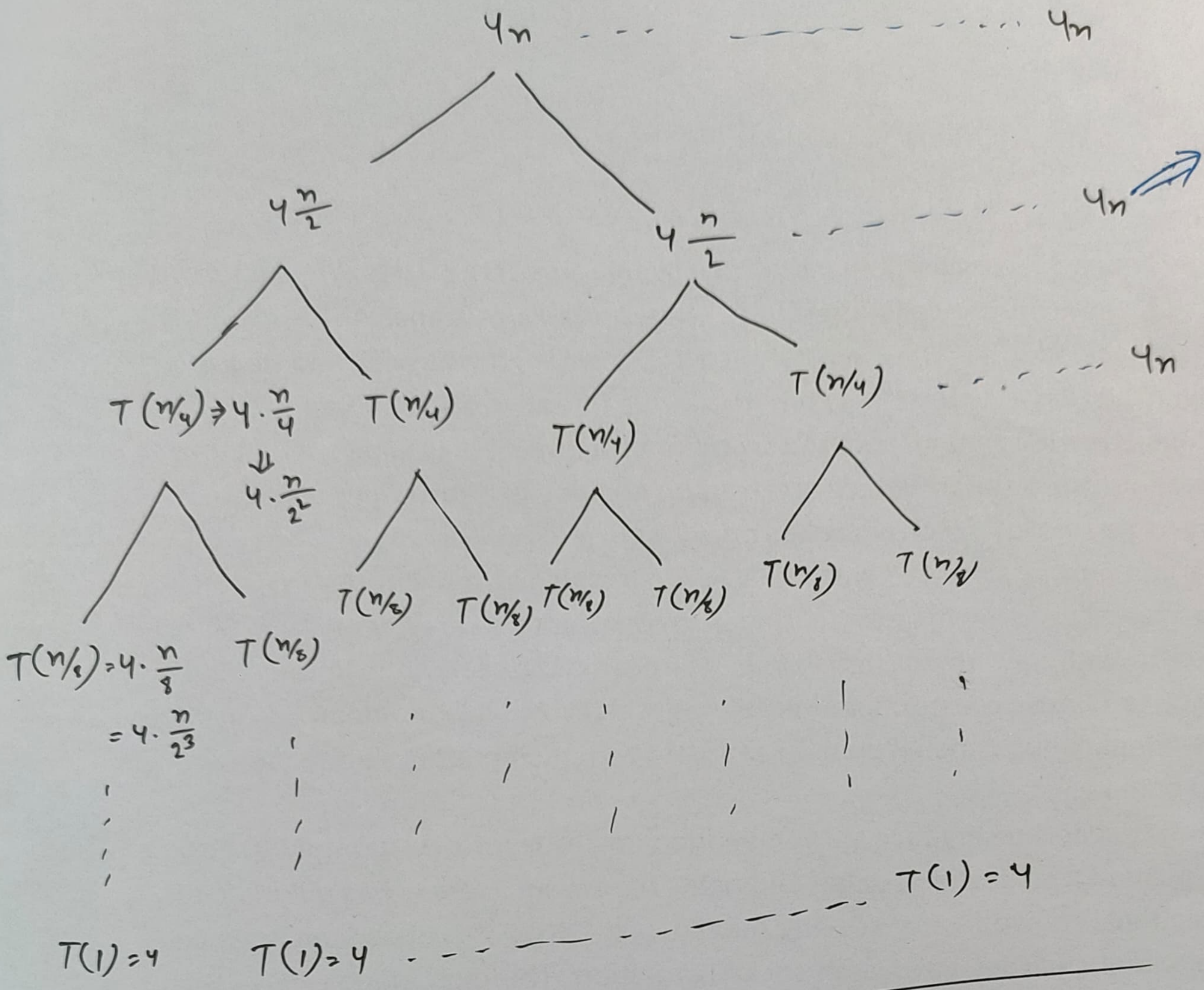
$T(n) = 2T(n/2) + 4n$

$T(n/2) = 2T(n/4) + 4\frac{n}{2}$

$T(n/4) = 2T(n/8) + 4 \cdot \frac{n}{4}$

$T(n/8) = 2T(n/16) + 4 \cdot \frac{n}{8}$

Call	#nodes
$T(n)$	$1 \Rightarrow 2^0$
$T(n/2)$	$2 \Rightarrow 2^1$
$T(n/4)$	$4 \Rightarrow 2^2$
$T(n/8)$	$8 \Rightarrow 2^3$
$T(n/2^i)$	2^i



Total $\Rightarrow 4n(\lg n + 1)$

$= 4n \lg n + 4n$

Let,

$n = 2^i$

$\Rightarrow \log_2 n = i \log_2 2$

$\Rightarrow \frac{n}{2^i} = 1$

$\therefore \sum_{i=0}^{\lg n} 4n = 4n \sum_{i=0}^{\lg n} = 4n(\lg n + 1)$

$= 4n \lg n + 4n$

$= O(n \lg n)$

$T(1) = c$

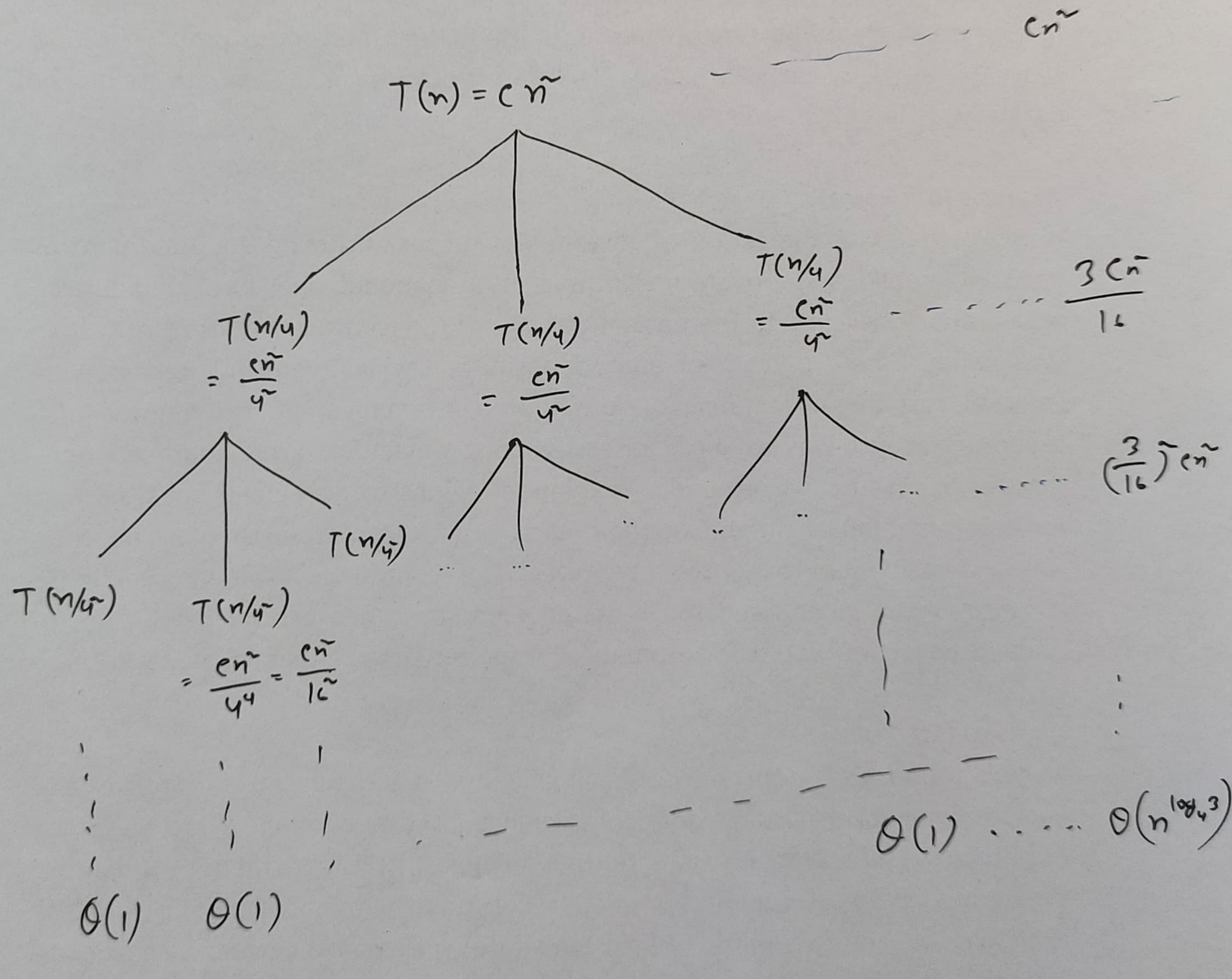
$T(n) = 3T(n/4) + \Theta(n^2)$

$T(n/4) = 3T(n/4^2) + \Theta\left(\frac{n^2}{4^2}\right)$

$T(n/4^2) = 3T(n/4^3) + \Theta\left(\frac{n^2}{4^4}\right)$

$T(n/4^3) = 3T\left(\frac{n}{4^4}\right) + \Theta\left(\frac{n^2}{4^6}\right)$

Call	# nodes
$T(n)$	1
$T(n/4)$	3
$T(n/4^2)$	$9 \Rightarrow 3^2$
$T(n/4^i)$	3^i



Let, $n = 4^i$

$\log_2 n = i \log_2 4$

$\log_4 n = i \log_4 4$

$i = \log_4 n$

$\therefore T(n/4^i) \Rightarrow 3^i = 3^{\log_4 n}$

$= n^{\log_4 3}$

$$\Rightarrow \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$< \sum_{i=0}^{\infty} \left(\frac{3}{16}\right)^i cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$= \frac{1}{1-\left(\frac{3}{16}\right)} cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$= \frac{16}{13} cn^2 + \underbrace{\Theta\left(n^{\log_4 3}\right)}_{O(n^{0.8}) < O(n^2)}$$

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

⊛

$$T(1) = c$$

$$T(n) = T(n/3) + T\left(\frac{2n}{3}\right) + cn$$

$$T(n/3) = T(n/9) + T\left(\frac{2n}{9}\right) + c\frac{n}{3}$$

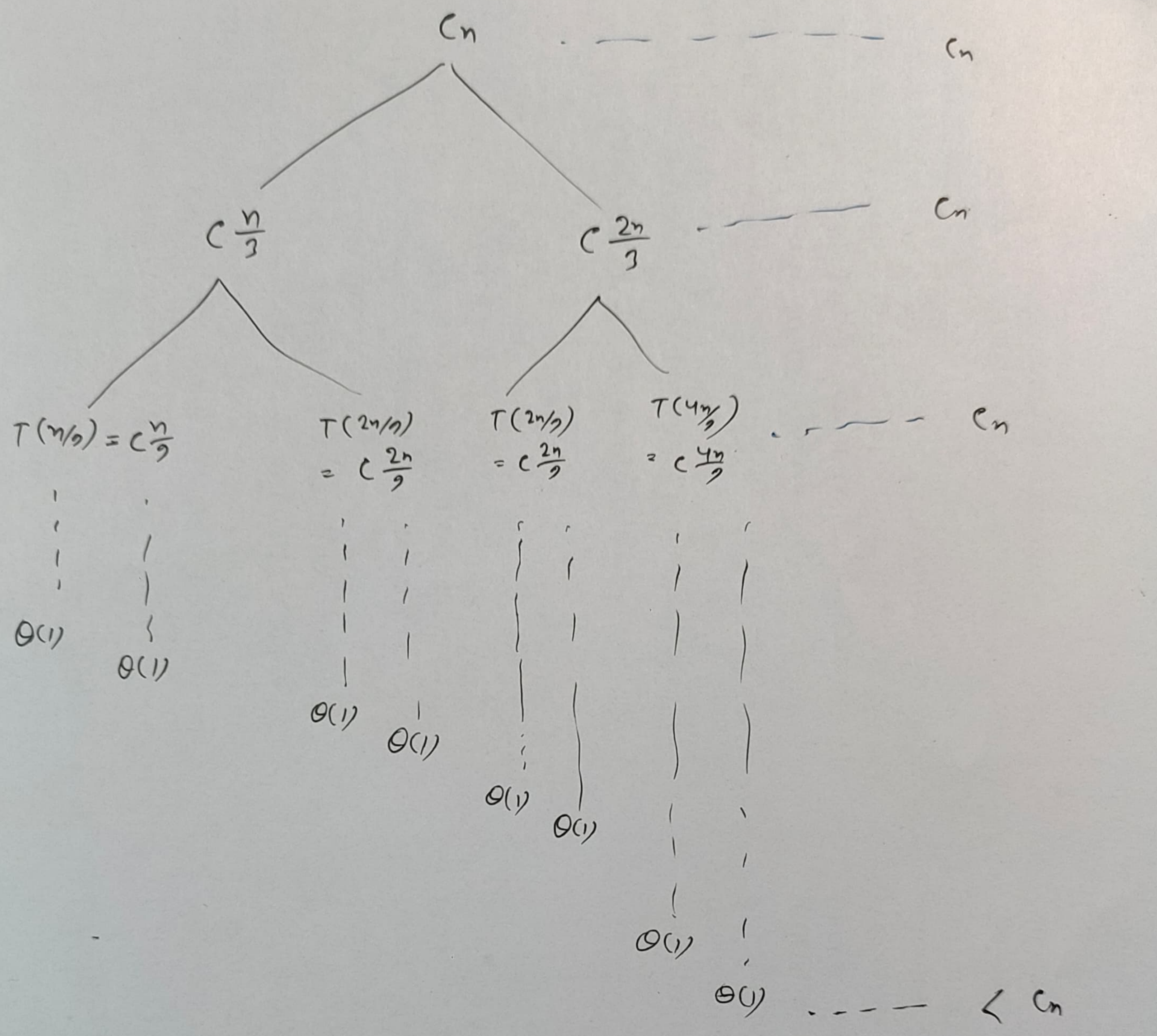
$$T\left(\frac{2n}{3}\right) = T\left(\frac{2n}{9}\right) + T\left(\frac{4n}{9}\right) + c\frac{2n}{3}$$

$$T\left(\frac{n}{9}\right) = T\left(\frac{n}{27}\right) + T\left(\frac{2n}{27}\right) + c\frac{n}{9}$$

$$T\left(\frac{2n}{9}\right) = T\left(\frac{2n}{27}\right) + T\left(\frac{4n}{27}\right) + c\frac{2n}{9}$$

$$T\left(\frac{4n}{9}\right) = T\left(\frac{4n}{27}\right) + T\left(\frac{8n}{27}\right) + c\frac{4n}{9}$$

call	# nodes
$T(n)$	1
$T(n/3)$	2
...	...


cn
cn
$c\frac{n}{3}$
$c\frac{2n}{3}$
cn
$T(n/9) = c\frac{n}{9}$
$T(2n/9) = c\frac{2n}{9}$
$T(2n/9) = c\frac{2n}{9}$
$T(4n/9) = c\frac{4n}{9}$
cn
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$\Theta(1)$
$< cn$

Total $O(n \lg n)$