CSE 201 1-21-20

Exercises

- show 1^{st} PMI $\Longrightarrow$ 2^{nd} PMI (hw2 #6)

- show 2^{nd} PMI $\Longrightarrow$ 1^{st} PMI (easy)

Hint on #6

1^{ST} PMI: for any $P: \mathbb{Z}^+ \to \{T, F\}$,

$$\Big(P(1) \wedge [\forall n > 1 : P(n-1) \to P(n)]\Big) \to \forall n \geq 1 : P(n)$$

2nd PMI: for any $Q : \mathbb{Z}^+ \to \{T, F\}$

$$\Big(Q(1) \wedge [\forall n > 1 : Q(1) \wedge \dots \wedge Q(n-1) \to Q(n)] \Big) \to \forall n \geq 1 : Q(n)$$

Proof: (1st PMI $\Rightarrow$ 2nd PMI)

Assume 1st PMI. Let $Q(n)$ be any Prop. fcn. $\mathbb{Z}^+ \to \{T, F\}$

Assume

(1) $Q(1)$

and

(2) $\forall n > 1 : Q(1) \wedge \dots \wedge Q(n-1) \to Q(n)$.

we must show

(3) $\forall n \geq 1 : Q(n)$

Define $P(n)$ by

$$P(n) = \cdots\cdots Q(\cdot)\cdots\cdots$$

$\vdots$

If P is defined correctly, then

$$P(1) = Q(1)$$

$$P(2) = ?$$

$$P(3) = ?$$

$$\vdots$$

$$P(n) = ?$$

we have

$$\begin{tikzcd}
 & \text{WOP} \arrow[dl, shift right, "\text{Thm 2, p. 10}"'] \arrow[dr, shift left, "\text{Thm 1, p. 10}"] & \\
2^{nd}\text{ PMI} \arrow[ur, shift right, "\text{Thm 3, p. 11}"'] \arrow[rr, shift right, "\text{exercise}"'] & & 1^{st}\text{ PMI} \arrow[ul, shift left, "\text{exercise}"] \arrow[ll, shift right, "\text{hw2 \#6}"']
\end{tikzcd}$$

Thus WOP iff 1^{st} PMI iff 2^{nd} PMI.

Recurrence Relations

3 methods

- substitution
- Recursion tree / Iteration
- Master theorem

Example

$$T(n) = \begin{cases} 2 & 1 \le n < 3 \\ 3T\left(\left\lfloor \frac{n}{3} \right\rfloor\right) + n & n \ge 3 \end{cases}$$

Guess: $T(n) = O(n \log n)$

must find c, n_0 s.t.

$T(n)$

$\forall n \geq n_0$: $\boxed{T(n) \leq c\, n \log n}$

notice

$n=1$: $T(1) \leq c\,1 \cdot \log 1 = 0$, $2 \leq 0$ ✗

$n=2$: $T(2) \leq c\,2 \log 2$, $2 \leq c \cdot 2 \log 2$

Pick: $n_0 = 2$ as lowest base case.

notation: n_1 is highest base case.

mimic induction step:

let $n > n_1$. Assume for all k in range $2 \leq k < n$ that

$$T(k) \leq c \cdot k \cdot \underline{\log} k$$

we must show:

$$T(n) \leq c \cdot n \, \underline{\log} n$$

so

$$T(n) = 3\,T\left(\left\lfloor \frac{n}{3} \right\rfloor\right) + n$$

$$\leq 3 \cdot c \cdot \left\lfloor \frac{n}{3} \right\rfloor \cdot \underline{\log} \left\lfloor \frac{n}{3} \right\rfloor + n$$

By ind. hyp. with $k = \left\lfloor \frac{n}{3} \right\rfloor$.

$$\leq \cancel{3}c \cdot \frac{n}{\cancel{3}} \cdot \log\left(\frac{n}{3}\right) + n \quad \left\{ \begin{array}{l} \text{since} \\ \lfloor x \rfloor \leq x \end{array} \right.$$

$$= cn(\log n - \log 3) + n$$

(Pick $\log = \log_3$)

$$= cn(\log_3 n - 1) + n$$

$$= cn\log n - cn + n \underset{\text{Want}}{\leq} cn\log n$$

Want: $-cn + n \leq 0$

i.e. $n \leq cn$

i.e. $1 \leq c$ ← constraint on c

Also need $2 \le \lfloor \frac{n}{3} \rfloor < n$ for all $n > n_1$.

note $2 \le \lfloor \frac{n}{3} \rfloor < n$ is equiv. to $2 \le \frac{n}{3} < n$, so we must have

$$n > n_1 \longrightarrow n \ge 6$$

Pick $\boxed{n_1 = 5}$

check Base cases:

$n = 2$ ✓

$n = 3$: $T(3) \le c\, 3 \log_3(3) = 3c$

$\therefore 9 \le 3c \quad \therefore c \ge 3$

$n=4$: $T(4) \le c \cdot 4 \log_3(4)$

$\therefore 10 \le 4c \log_3(4) \quad \therefore c \ge \dfrac{5}{2\log_3(4)}$

$n=5$: $T(5) \le c \cdot 5 \log_3(5)$

$\therefore 11 \le 5c \log_3(5) \quad \therefore c \ge \dfrac{11}{5\log_3(5)}$

Note $\quad 3 > \dfrac{5}{2\log_3 4} > \dfrac{11}{5\log_3 5}$

so pick $\boxed{c = 3}$.

Exercise

Show that $T(n) \leq 3n\log_3(n)$ for all $n \geq 2$. (use base cases $n = 2, 3, 4, 5$.)

conclusion: $T(n) = O(n \log n)$

in fact $T(n) = \Theta(n \log n)$

Exercise

Show $T(n) = \Omega(n \log n)$. i.e. find c, n_0 s.t.

$$\forall n \geq n_0 : T(n) \geq c n \log_3(n)$$

Ex.

$$T(n) = \begin{cases} 1 & n=1 \\ 2T(\lfloor \frac{n}{2} \rfloor)+1 & n \geq 2 \end{cases}$$

Guess: $T(n) = O(n)$.

must find Pos. c, n_0 st.

$$\forall n \geq n_0 : T(n) \leq cn$$

mimic ind. step:

let $n > n_0$. Assume for all k in range $n_0 \leq k < n$ that

$$T(k) \leq ck$$

must show $T(n) \le cn$.

Then

$$T(n) = 2T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + 1$$

$$\le 2 \cdot c \left\lfloor \frac{n}{2} \right\rfloor + 1 \qquad \left\{ \begin{array}{l} \text{by ind. hyp.} \\ \text{with } k = \left\lfloor \frac{n}{2} \right\rfloor \end{array} \right.$$

$$\le \cancel{2} \cdot c \left(\frac{n}{\cancel{2}}\right) + 1$$

$$= cn + 1 \underbrace{\le}_{\text{want}} cn$$

i.e. we want $1 \le 0$ ✗

same guess: $T(n) = O(n)$

This time find positive c, b, and n_0

such that

$$\forall n \geq n_0 : T(n) \leq cn - b$$

<u>mimic ind. step.</u>

let $n > n_0$. Assume for $n_0 \leq k < n$ that $T(k) \leq ck - b$. must show that $T(n) \leq cn - b$.

As before

$$T(n) = 2T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + 1$$

$$\le 2\left(c\left\lfloor \frac{n}{2} \right\rfloor - b\right) + 1 \quad \left\{ \begin{array}{l} \text{by ind. hyp.} \\ \text{with } k = \left\lfloor \frac{n}{2} \right\rfloor \end{array} \right.$$

$$= 2c\left\lfloor \frac{n}{2} \right\rfloor - 2b + 1$$

$$\le \cancel{2} c \cdot \frac{n}{\cancel{2}} - 2b + 1$$

$$= cn - 2b + 1 \underset{\substack{\uparrow \\ \text{want}}}{\le} cn - b$$

i.e. $-2b + 1 \le -b$

i.e. $1 \le b$

If $n_0 = 1$, then base case is

$$T(1) \le c \cdot 1 - b$$

$$1 \le c - b$$

If $b = 1$, then need $1 \le c - 1$,

i.e. $c \ge 2$

Pick $\boxed{n_0 = 1,\ c = 2,\ b = 1}$

<u>Exercise</u> complete problem by proving:

$$\forall n \ge 1 : T(n) \le 2n - 1$$

whence $T(n) = O(n)$

Exercise

show using substitution that

$T(n) = \Omega(n)$, hence

$$T(n) = \Theta(n).$$

Recursion Tree / Iteration

Ex.

$$T(n) = \begin{cases} 1 & 1 \le n < 3 \\ 2T(\lfloor \frac{n}{3} \rfloor) + n & n \ge 3 \end{cases}$$

write: $T(n) = 2T\left(\frac{n}{3}\right) + n$.

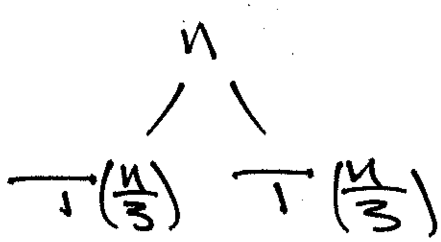
$$= n + T\left(\frac{n}{3}\right) + T\left(\frac{n}{3}\right)$$

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

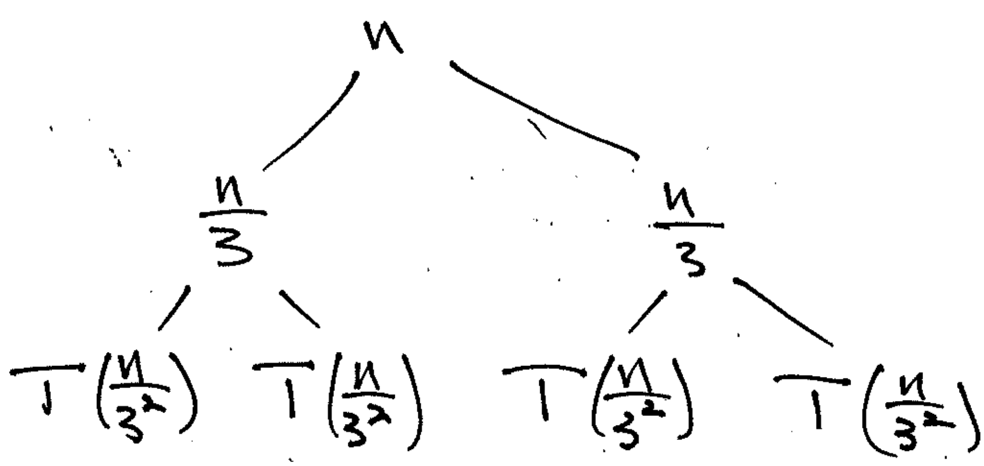
0^{th} tree

$T(n)$

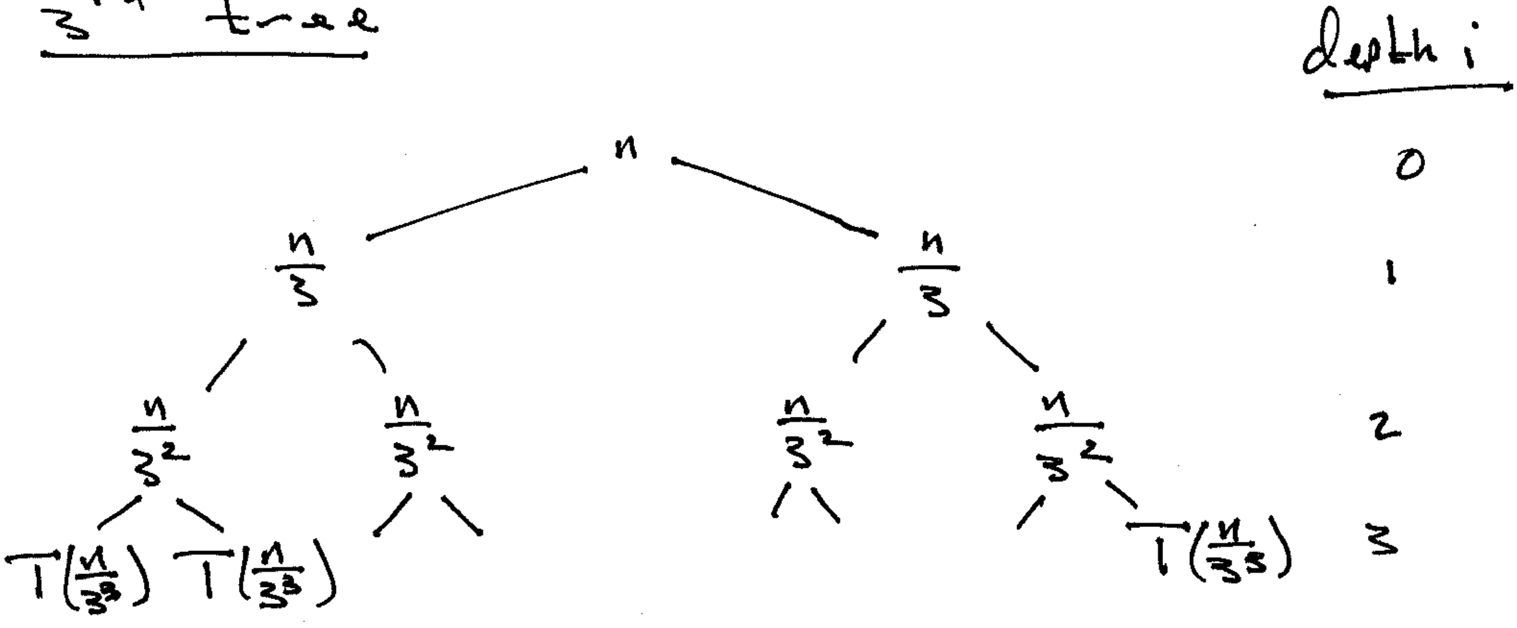
1^{st} tree



2^{nd} tree

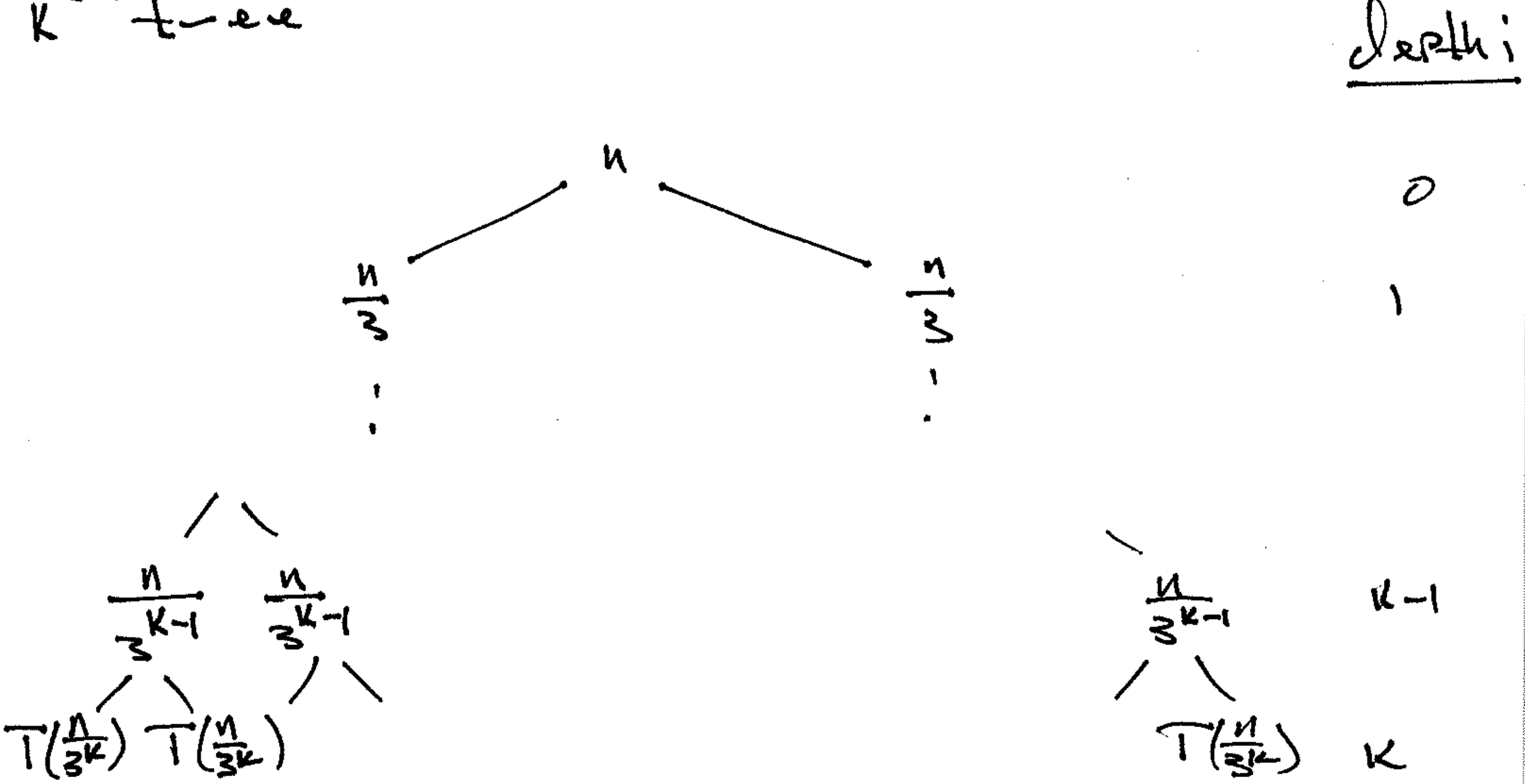


3rd tree



$\vdots$

k^{th} tree



nodes at depth i : 2^i

Value of nodes at depth i : $\frac{n}{3^i}$
$(0 \le i \le k-1)$

Value of nodes at depth k : $T\left(\frac{n}{3^k}\right)$

so

$$\boxed{T(n) = \sum_{i=0}^{k-1} 2^i \cdot \left(\frac{n}{3^i}\right) + 2^k \cdot T\left(\frac{n}{3^k}\right)}$$