

## Recitation 2-7

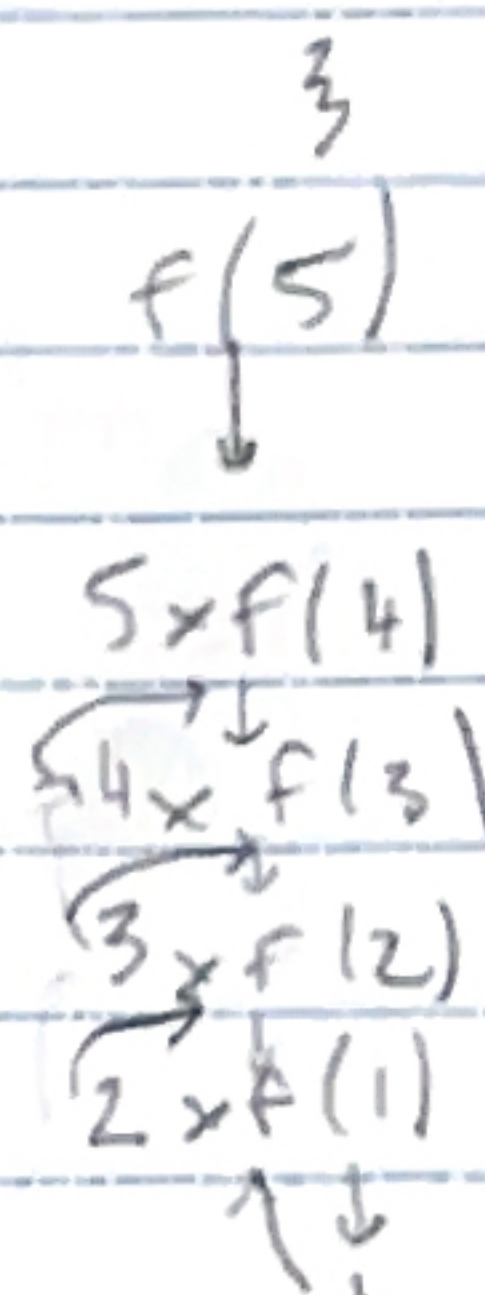
### Recursion tree method

Factorial:

$f(n) \{$

if  $f(n=1)$  return 1; ← Base condition

return  $n \cdot f(n-1)$ ; ← recursion step



$$T(n) = T(n-1) + 1$$

$$T(n-1) \rightarrow 1$$

$\vdots$

$$1 \rightarrow 1$$

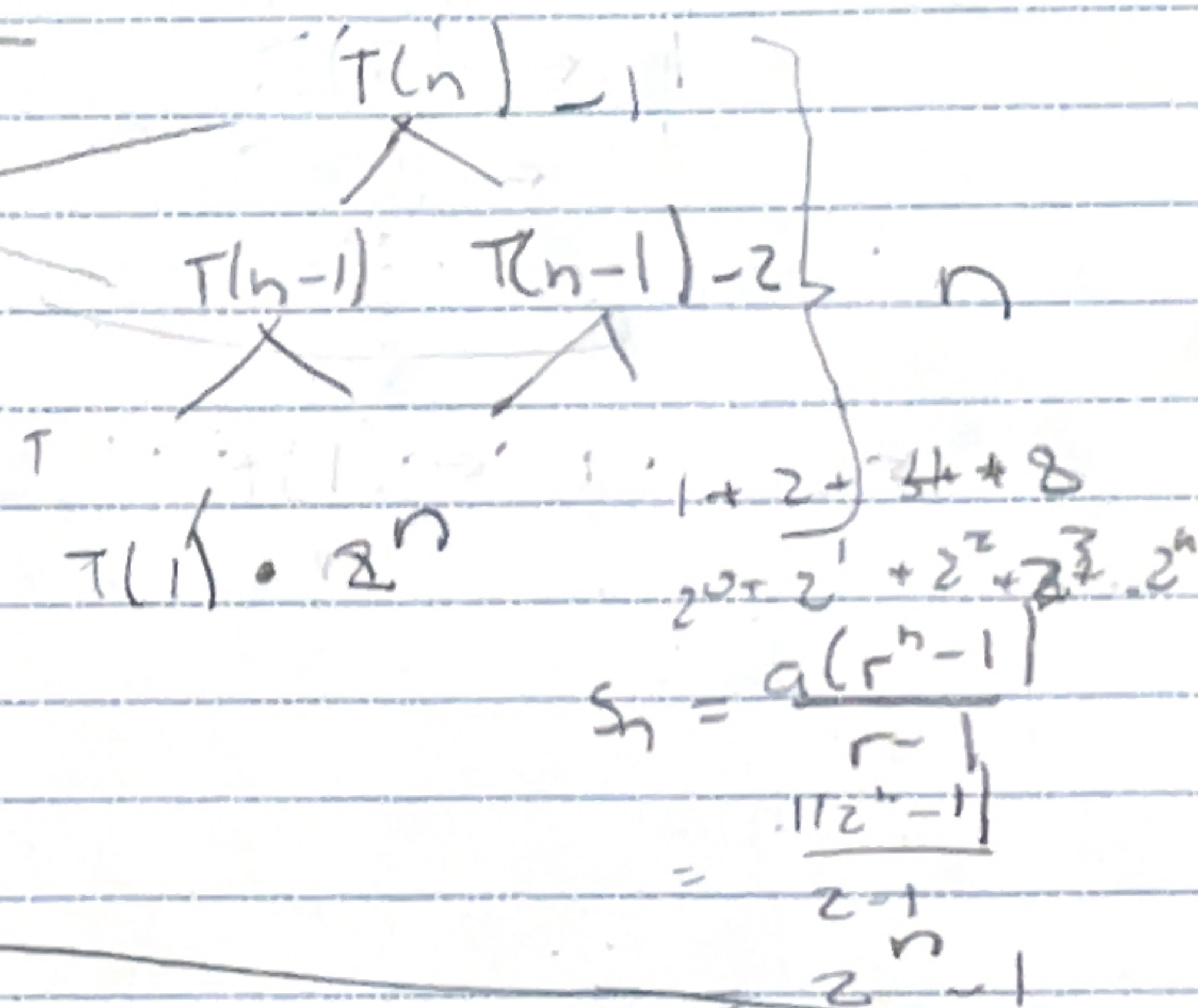
$$T(n) = O(n)$$

Example)

$$T(n) = 2T(n-1) + 1$$

$$T(n) = O(2^n - 1)$$

$$= O(2^n)$$



### Recursion tree method

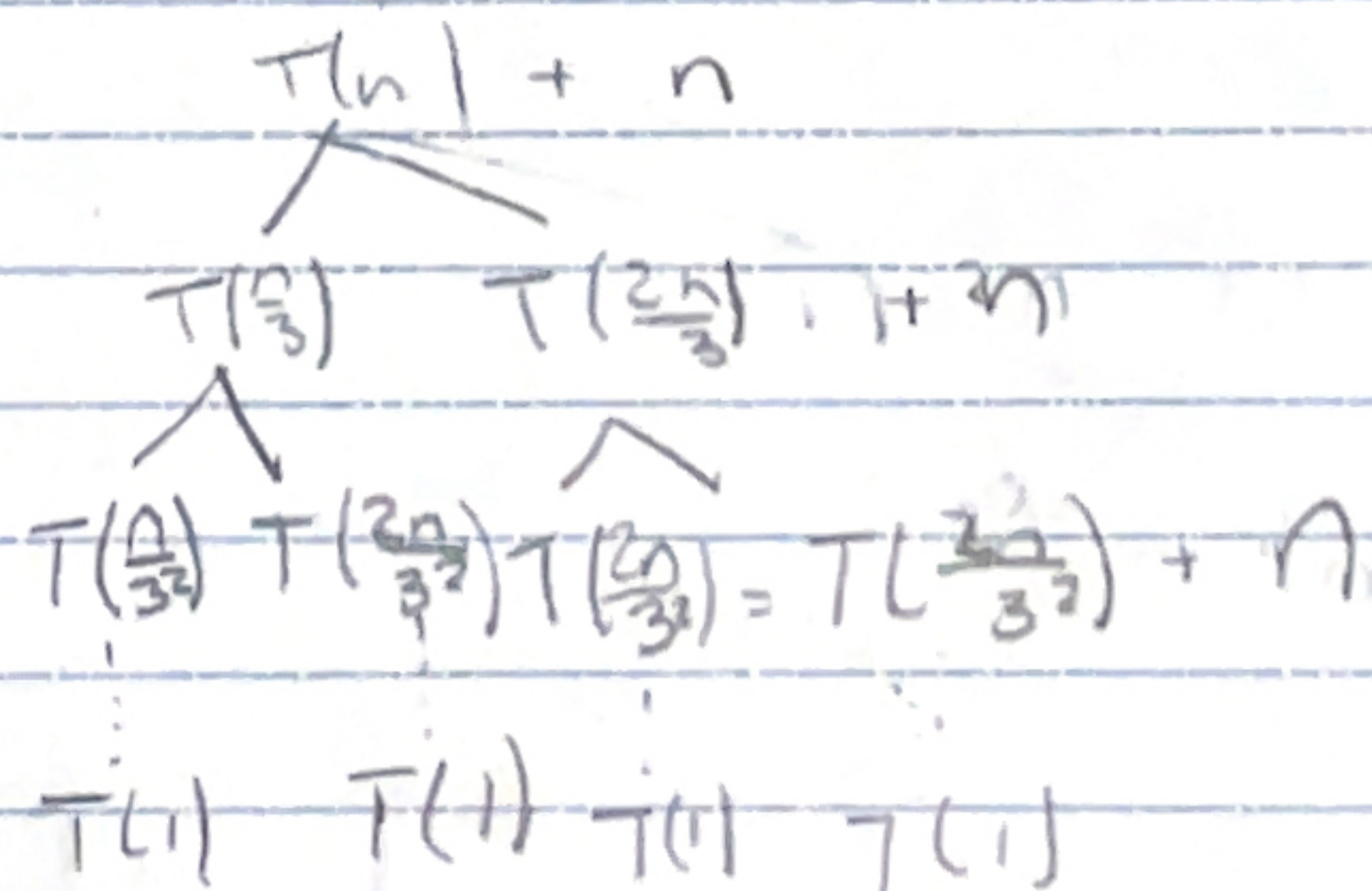
$$T(n) = T(n/3) + T(2n/3) + n$$

A symmetric tree

$$n, \frac{2n}{3}, \frac{4n}{9}, \frac{8n}{27}, \frac{2^k n}{3^k}$$

$$n = \left(\frac{3}{2}\right)^k = \log_{\frac{3}{2}} n = \log_{\frac{3}{2}} \left(\frac{3}{2}\right)^k = k$$

$$n \times \log_{\frac{3}{2}} n = O(n \log n)$$





Master Theorem:  $T(n) = aT(\frac{n}{b}) + F(n)$

Case 1:

IF there exists a constant  $\epsilon > 0$

Such that  $F(n) = O(n^{\log_b a - \epsilon})$

Then,  $T(n) = \Theta(n^{\log_b a})$

e.g.  $T(n) = 9T(\frac{n}{3}) + n$

$$a = 9$$

$$b = 3$$

$$F(n) = n$$

$$n^{\log_b a} = n^{\log_3 9} = n^2$$

$$O(n^{2-\epsilon}) \quad 0 < \epsilon \leq 1$$

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

Case 2:

IF there exists constant  $k \geq 0$

Such that  $F(n) = \Theta(n^{\log_b a} \log^k n)$ , then

$$T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$$

$$T(n) = T(\frac{n}{2}) + 1$$

$$a = 1$$

$$b = 2$$

$$g(n) = 1$$