

# Fundamentos de análisis y diseño de algoritmos

Ecuaciones de recurrencia

```
def factorial(n):
```

```
    if n==0:
```

```
        return 1
```

```
    else:
```

```
        return n*factorial(n-1)
```

$n=0 \rightarrow f(n)=1$   
 $f(0)=1$

$f(n) = n * f(n-1)$

$$f(n) = \begin{cases} 1 & n=0 \\ n * f(n-1) & n > 0 \end{cases}$$

# Recurrencias

Método de iteración

Método maestro\*

Método de sustitución

# Recurrencias

---

## Método de iteración

Expandir la recurrencia y expresarla como una suma de términos que dependen de  $n$  y de las condiciones iniciales

# Recurrencias

$$T(n) = n + 3T(n/4), T(1) = \Theta(1) \text{ y } n \text{ par}$$

Expandir la recurrencia 2 veces

$$0) \quad T(n) = 3 \boxed{T\left(\frac{n}{4}\right)} + n$$

$$1) \quad T(n) = 3 \left( 3 T\left(\frac{n}{4^2}\right) + \frac{n}{4} \right) + n$$

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

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

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

# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3T(n/4^3)$$

# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3T(n/4^3)$$

$$T(n/4^4)$$

$$T(n/4^5)$$

¿Cuándo se detienen las iteraciones?

# Recurrencias

---

$$1) T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$2) n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$3) n + 3*n/4 + 3^2*n/4^2 + 3^3T(\underbrace{n/4^3}_{\hat{1}})$$

¿Cuándo se detienen las iteraciones?

Cuando se llega a  $T(1)$

$$T\left(\frac{n}{4^i}\right) \rightarrow T(1)$$

$$\frac{n}{4^i} = 1$$

$$i = \log_4(n)$$



# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3T(n/4^3)$$

¿Cuándo se detienen las iteraciones?

Cuando se llega a  $T(1)$ , esto es, cuando  $(n/4^i)=1$

# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3T(n/4^3)$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3n/4^3 + \dots + 3^{\log_4 n}T(1)$$

¿Cuándo se detienen las iteraciones?

Cuando se llega a  $T(1)$ , esto es, cuando  $(n/4^i)=1$

# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3T(n/4^3)$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3n/4^3 + \dots + 3^{\log_4 n}T(1)$$

Después de iterar, se debe tratar de expresar como una sumatoria con forma cerrada conocida

# Recurrencias

---

$$T(n) = n + 3T(n/4]$$

$$n + 3 ( n/4] + 3T(n/16])$$

$$n + 3 ( n/4] + 3(n/16] + 3T(n/64]) )$$

$$n + 3*n/4] + 3^2*n/4^2] + 3^3(n/4^3]) + ... + 3^{\log_4 n} \Theta(1)$$

$$\leq n + 3n/4 + 3^2n/4^2 + 3^3n/4^3 + ... + 3^{\log_4 n} \Theta(1)$$

# Recurrencias

---

$$T(n) = n + 3T(n/4)$$

$$n + 3 ( n/4 + 3T(n/16) )$$

$$n + 3 ( n/4 + 3(n/16 + 3T(n/64) ) )$$

$$n + 3*n/4 + 3^2*n/4^2 + 3^3(n/4^3) + \dots + 3^{\log_4 n} \Theta(1)$$

$$\leq n + 3n/4 + 3^2n/4^2 + 3^3n/4^3 + \dots + 3^{\log_4 n} \Theta(1)$$

$$= \left( \sum_{i=0}^{\log_4 n} \left( \frac{3}{4} \right)^i n \right) + 3^{\log_4 n} \Theta(1)$$

$$= n \left( \frac{(3/4)^{(\log_4 n)} - 1}{(3/4) - 1} \right) + n^{\log_4 3} = n * 4 (1 - (3/4)^{(\log_4 n)}) + \Theta(n^{\log_4 3})$$

$$= O(n)$$

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

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

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

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

$$i) \quad T(n) = 3^i T\left(\frac{n}{4^i}\right) + \left(\frac{3}{4}\right)^{i-1}n + \left(\frac{3}{4}\right)^{i-2}n + \dots + \left(\frac{3}{4}\right)^0 n$$

$$\frac{n}{4^i} = 1$$

$$i = \log_4(n)$$

$$T(n) = 3^{\log_4(n)} T(1) + \left(\frac{3}{4}\right)^{\log_4(n)-1}n + \left(\frac{3}{4}\right)^{\log_4(n)-2}n + \dots + \left(\frac{3}{4}\right)^1 n + \left(\frac{3}{4}\right)^0 n$$

$$T(n) = n^{\log_4(3)} T(1) + \sum_{i=0}^{\log_4(n)-1} \left(\frac{3}{4}\right)^i n$$

$$\sum_{i=0}^n ar^i = \frac{ar^{n+1} - a}{r-1}$$

$$a = n \quad r = \frac{3}{4} \quad n = \log_4(n) - 1$$

$$T(n) = n^{\log_4(3)} T(1) + \frac{n \left(\frac{3}{4}\right)^{\log_4(n)} - n}{\frac{3}{4} - 1}$$

$$T(n) = n^{\log_4(3)} T(1) + \frac{n \times n^{\log_4\left(\frac{3}{4}\right)} - n}{-\frac{1}{4}}$$

$$T(1) = \Theta(1)$$

$$T(1) = C$$

# Recurrencias

---

Resuelva por el método de iteración

$$T(n) = \underline{2T(n/2) + 1}, T(1) = \Theta(1)$$

$$1) T(n) = 2 \left( 2T\left(\frac{n}{2^2}\right) + 1 \right) + 1 = 2^2 \underbrace{T\left(\frac{n}{2^2}\right)} + \underline{2 + 1}$$

$$2) T(n) = 2^2 \left( 2T\left(\frac{n}{2^3}\right) + 1 \right) + 2 + 1 = 2^3 T\left(\frac{n}{2^3}\right) + 2^2 + 2^1 + 2^0$$

$$i) T(n) = 2^i T\left(\frac{n}{2^i}\right) + \underline{2^{i-1} + 2^{i-2} + \dots + 2^1 + 2^0}$$

$$T(1)$$

$$\frac{n}{2^i} = 1 \quad n = 2^i \quad i = \log_2(n)$$

$$T(n) = 2^{\log_2(n)} T(1) + 2^{\log_2(n)-1} + 2^{\log_2(n)-2} + \dots + 2^1 + 2^0$$

$$T(n) = n T(1) + \sum_{i=0}^{\log_2(n)-1} 2^i$$

$$T(n) = cn + \frac{2^{\log_2(n)} - 1}{2 - 1}$$

$$T(n) = cn + n - 1 \quad O(n)$$



# Recurrencias

---

Resuelva por el método de iteración

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

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

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

$$T(n) = 2 \left( 2T\left(\frac{n}{2^2}\right) + \frac{n}{2} \right) + \overset{1n}{n} = 2^2 T\left(\frac{n}{2^2}\right) + n + n$$

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

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

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

$$T(1) = T\left(\frac{n}{2^i}\right)$$

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

$$i = \log_2(n)$$

$$T(n) = 2^{\log_2(n)} T(1) + \log_2(n) \times n$$

$$T(n) = n \times T(1) + \log_2(n) \times n \quad O(n \log_2(n))$$

# Recurrencias

---

Resuelva por el método de iteración

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

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

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

# Recurrencias

---

Resuelva por el método de iteración

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

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

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

Demuestre que  $T(n) = T(n/2) + n$ , es  $\Omega(n \log n)$

$$T(n) = 5T(n/4) + n,$$

$$T(n) = 5 \left( 5T\left(\frac{n}{4^2}\right) + \frac{n}{4} \right) + n = 5^2 T\left(\frac{n}{4^2}\right) + \frac{5}{4}n + n$$

$$T(n) = 5^2 \left( 5T\left(\frac{n}{4^3}\right) + \frac{n}{4^2} \right) + \frac{5}{4}n + n$$

$$T(n) = 5^3 T\left(\frac{n}{4^3}\right) + \frac{5^2}{4^2}n + \frac{5}{4}n + n$$

$$T(n) = 5^i T\left(\frac{n}{4^i}\right) + \left(\frac{5}{4}\right)^{i-1}n + \left(\frac{5}{4}\right)^{i-2}n + \dots + \left(\frac{5}{4}\right)^2n + \left(\frac{5}{4}\right)n$$

$$T(1) \quad \frac{n}{4^i} = 1 \quad i = \log_4(n)$$

$$T(n) = 5^{\log_4(n)} T(1) + \sum_{i=0}^{\log_4(n)-1} \left(\frac{s}{4}\right)^i n$$

$$T(n) = n^{\log_4(s)} C \times n + n \left( \frac{\left(\frac{s}{4}\right)^{\log_4(n)} - 1}{\frac{s}{4} - 1} \right)$$

$$T(n) = n^{\log_4(s)} \times C \times n + n \left( n^{\log_4\left(\frac{s}{4}\right)} - 1 \right)$$

$$T(1) = C \times n$$

$$T(1) = C$$

# Recurrencias

---

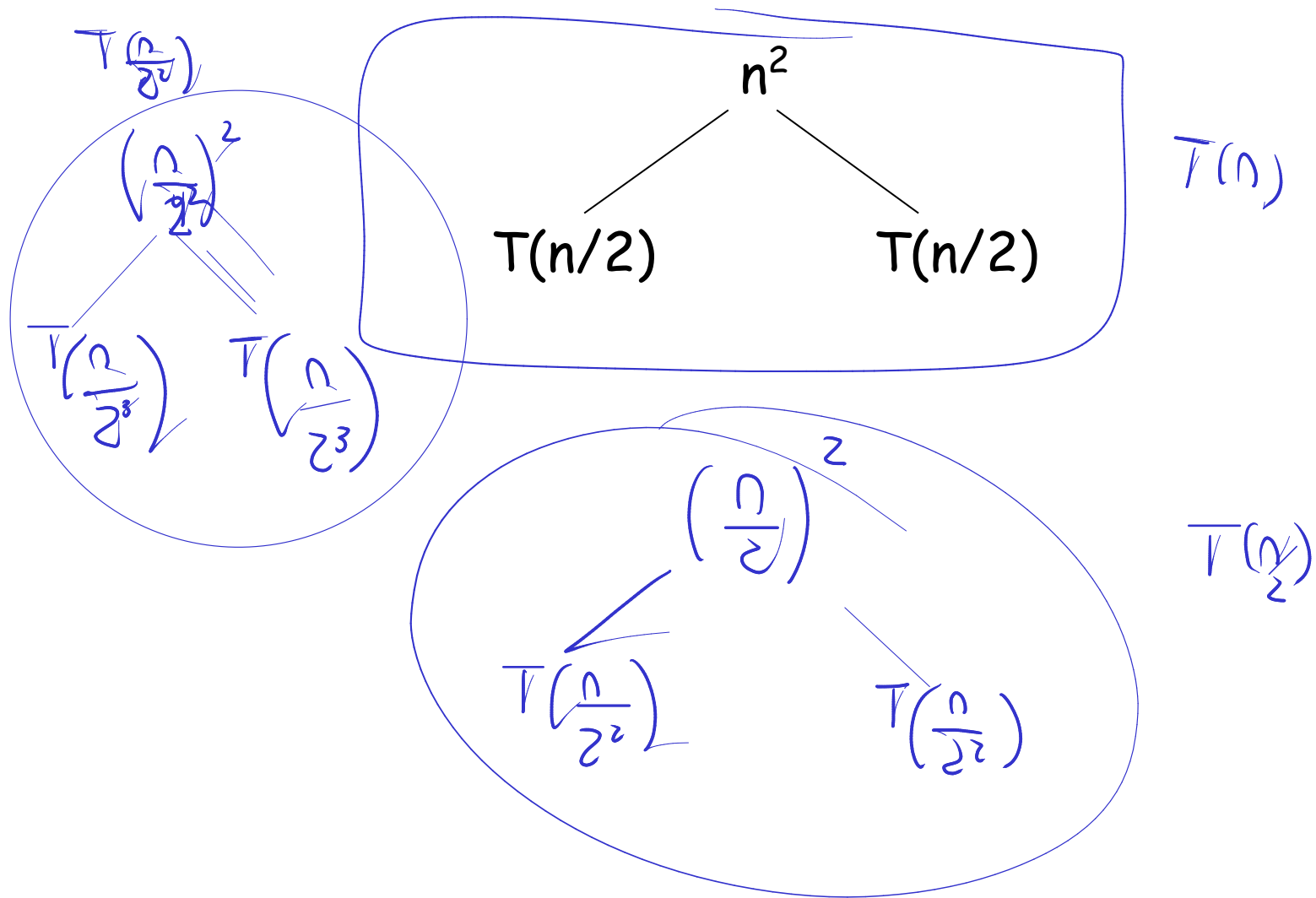
Iteración con árboles de recursión

$$T(1) = 1$$

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

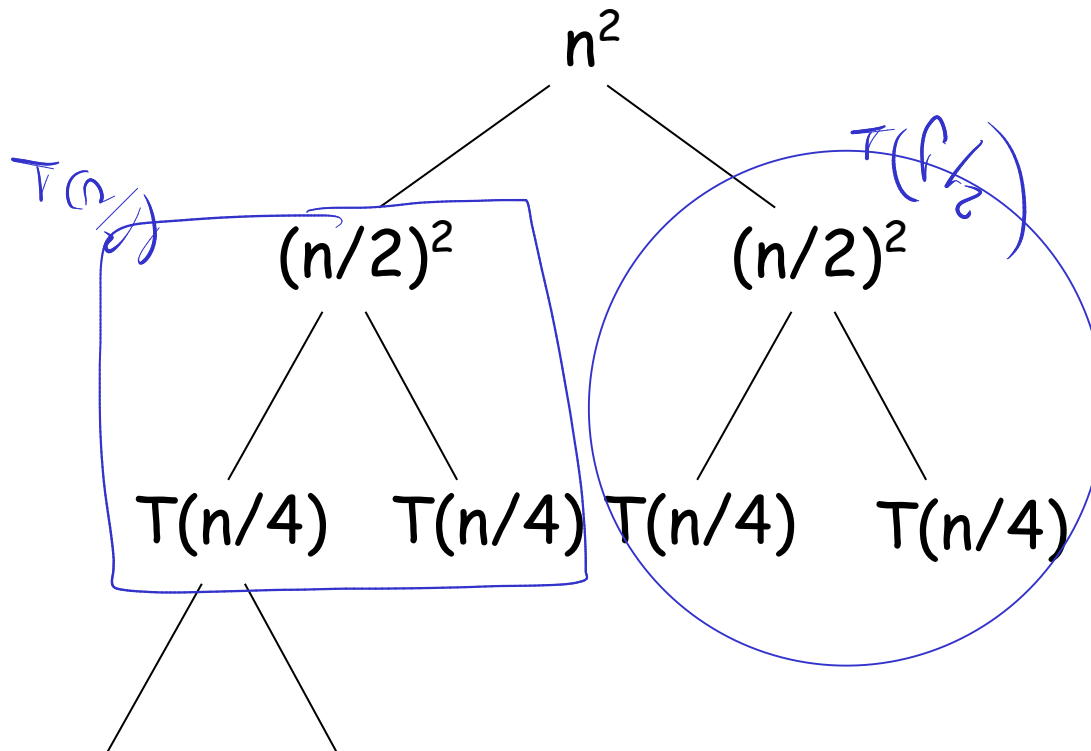


# Recurrencias



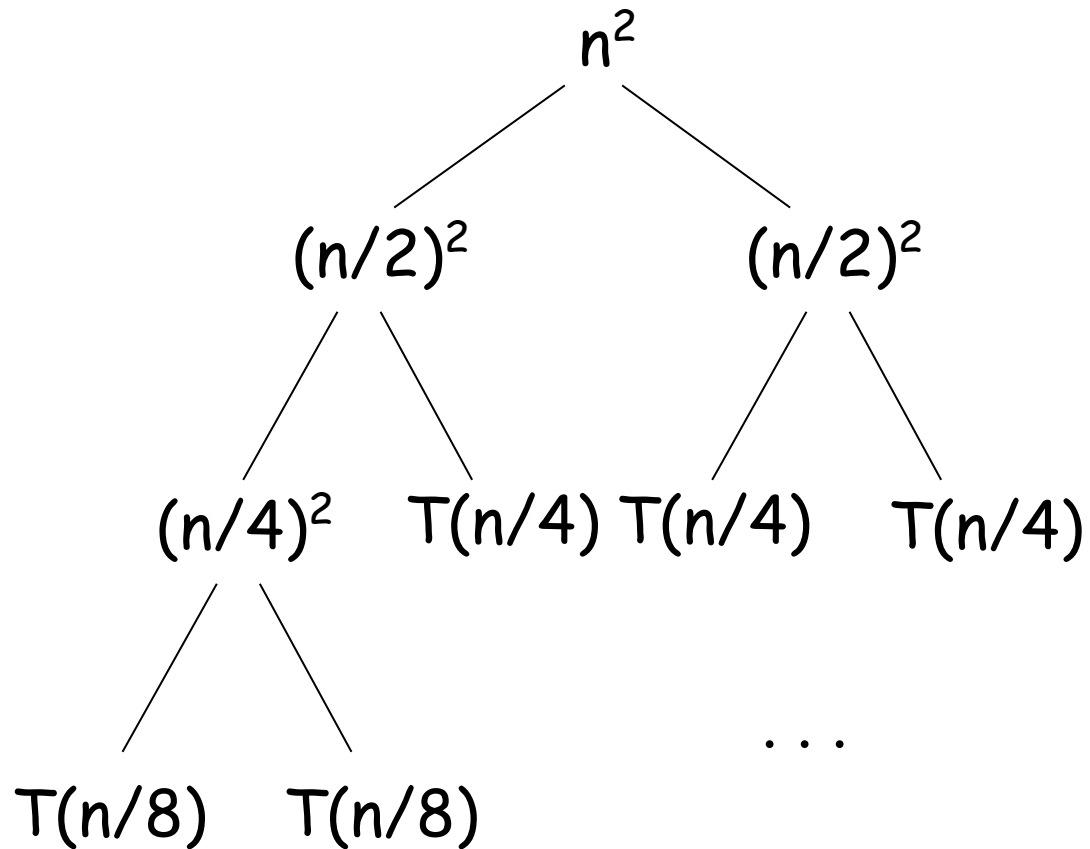
# Recurrencias

---

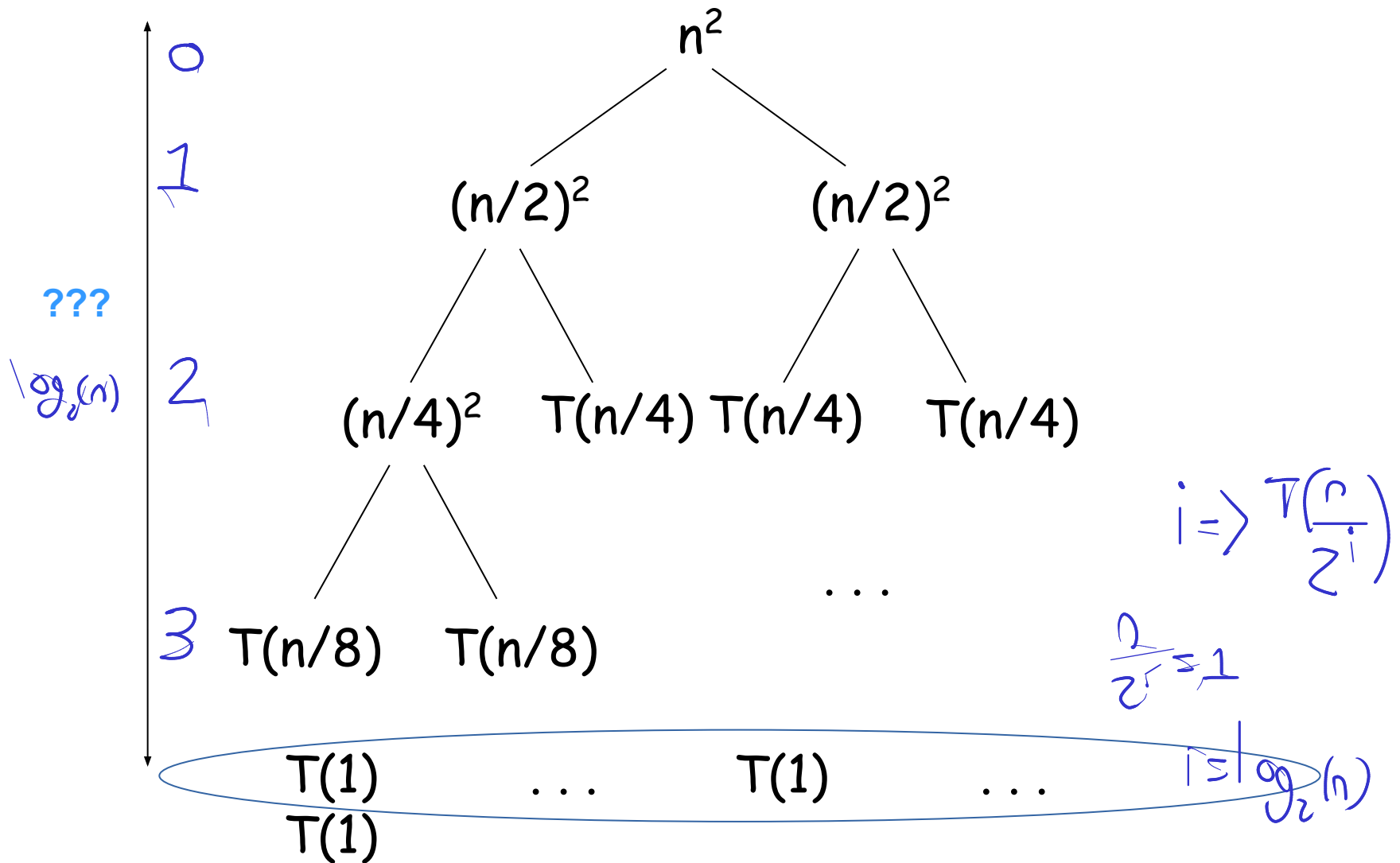


# Recurrencias

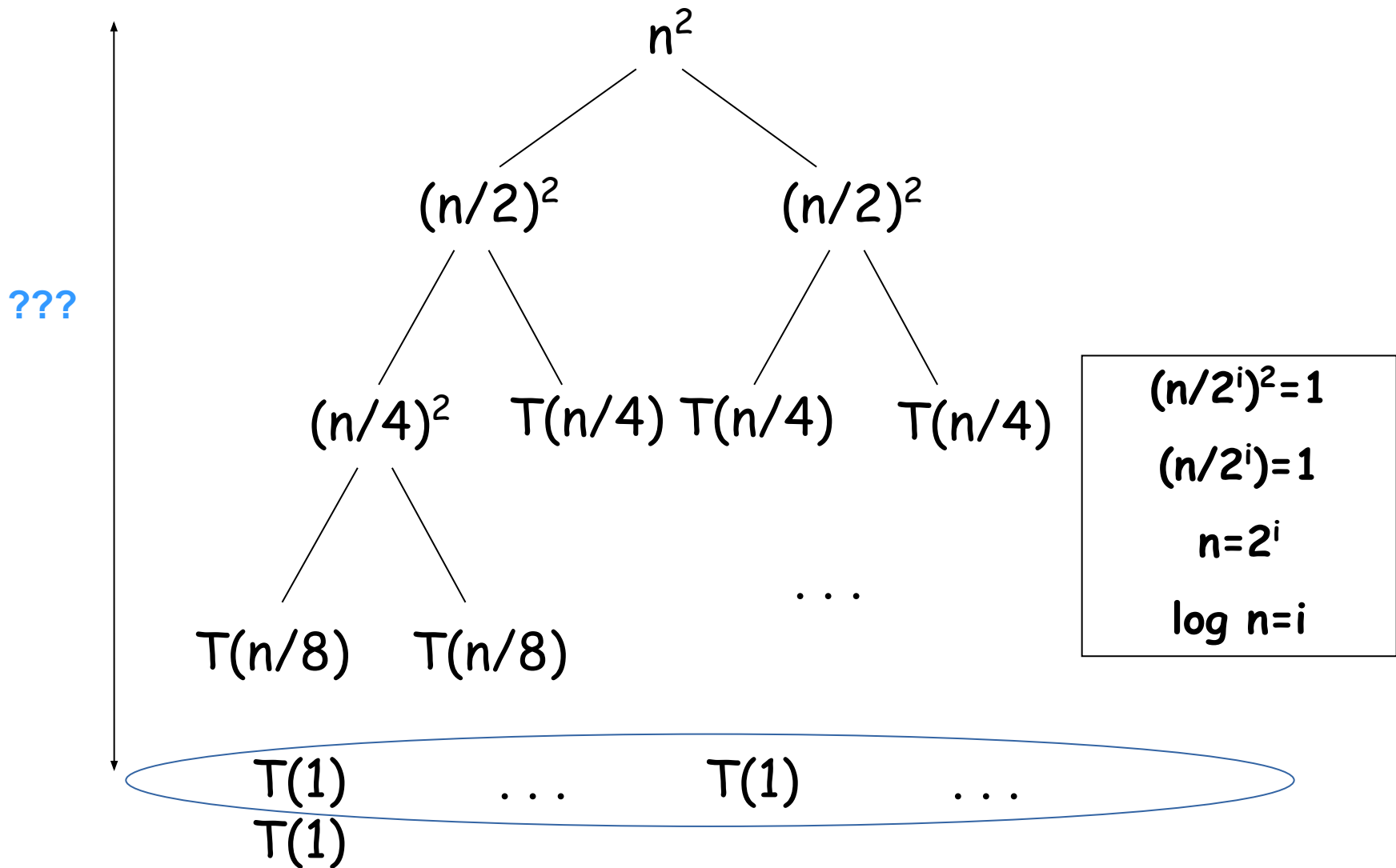
---



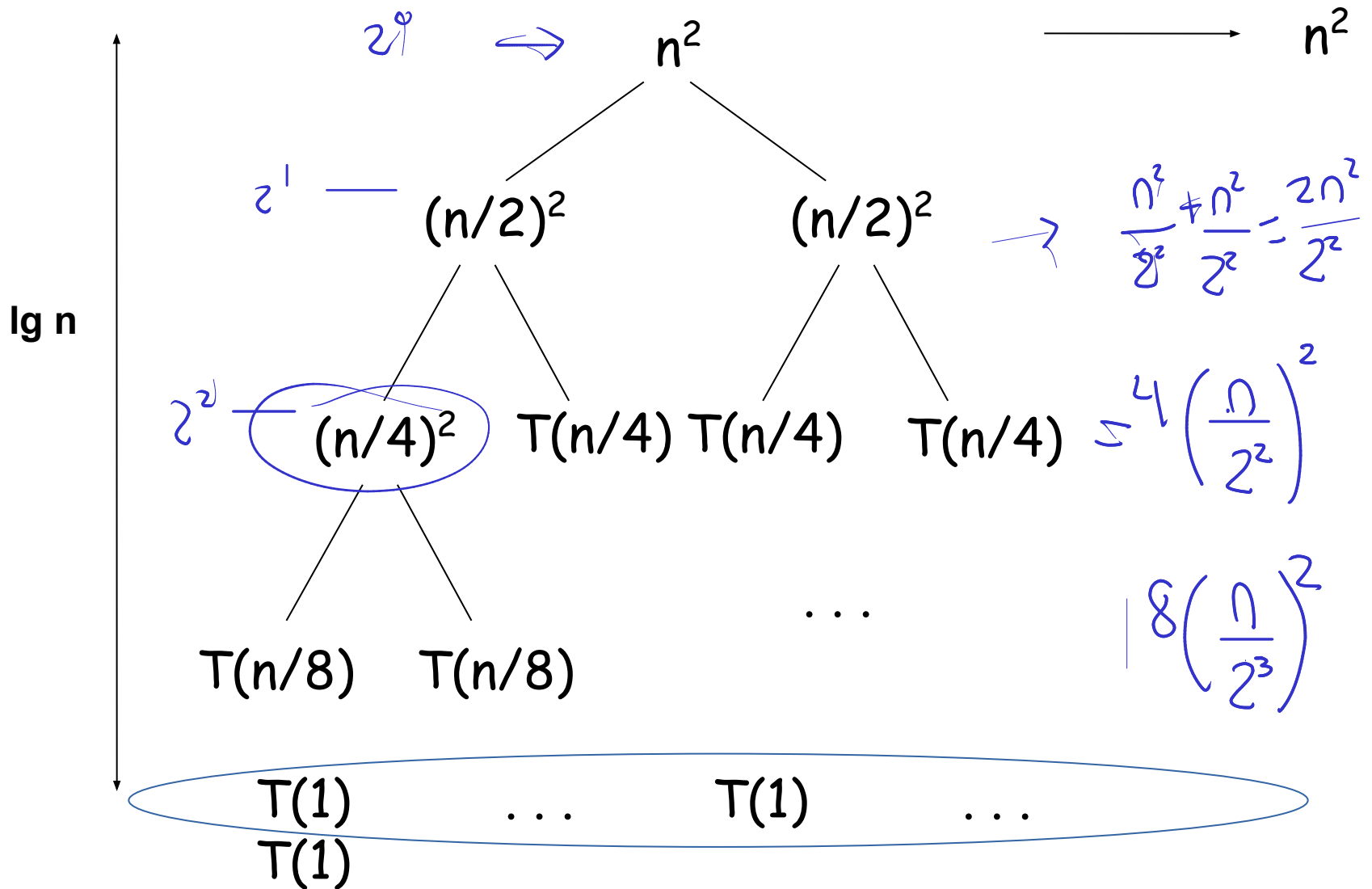
# Recurrencias



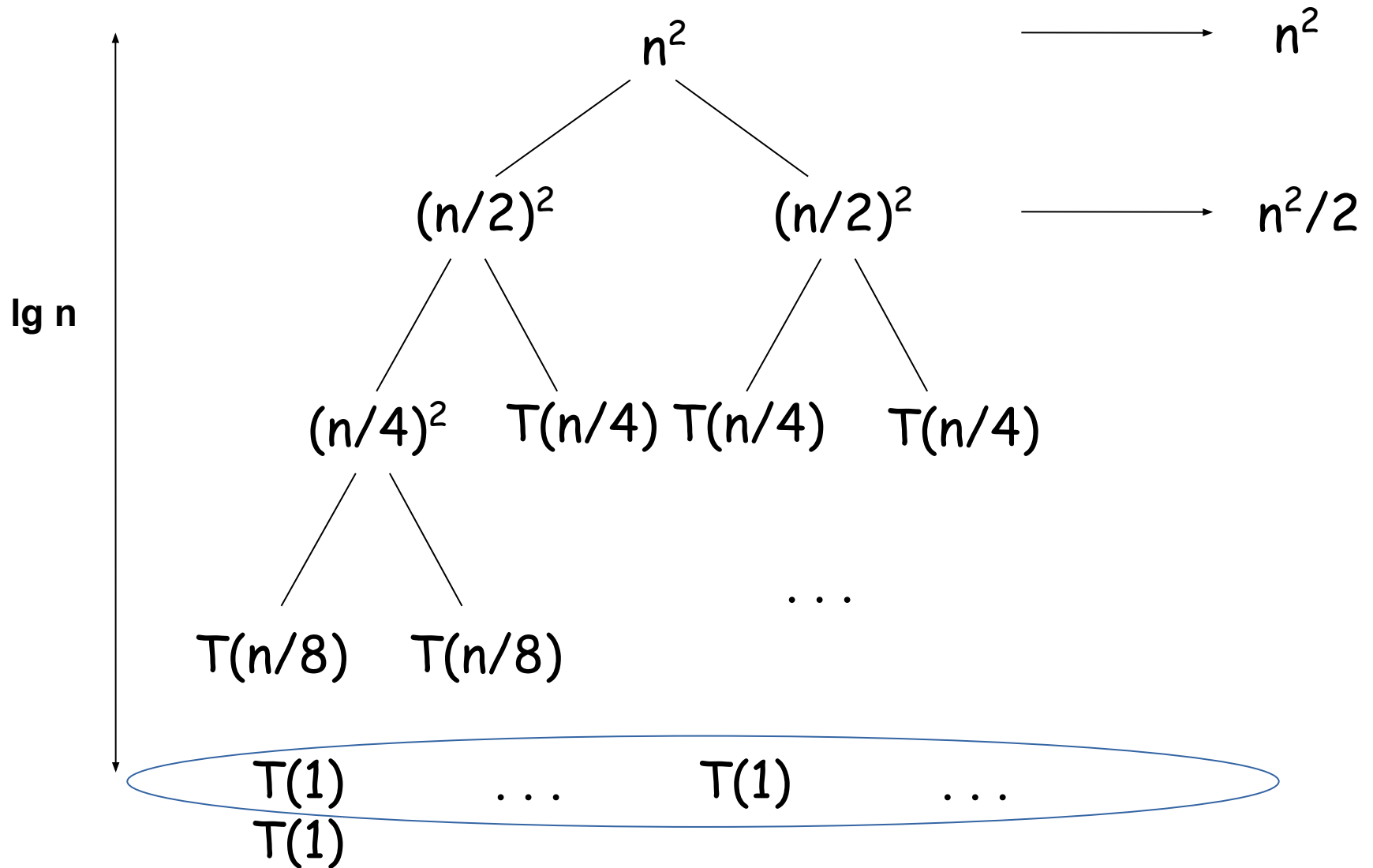
# Recurrencias



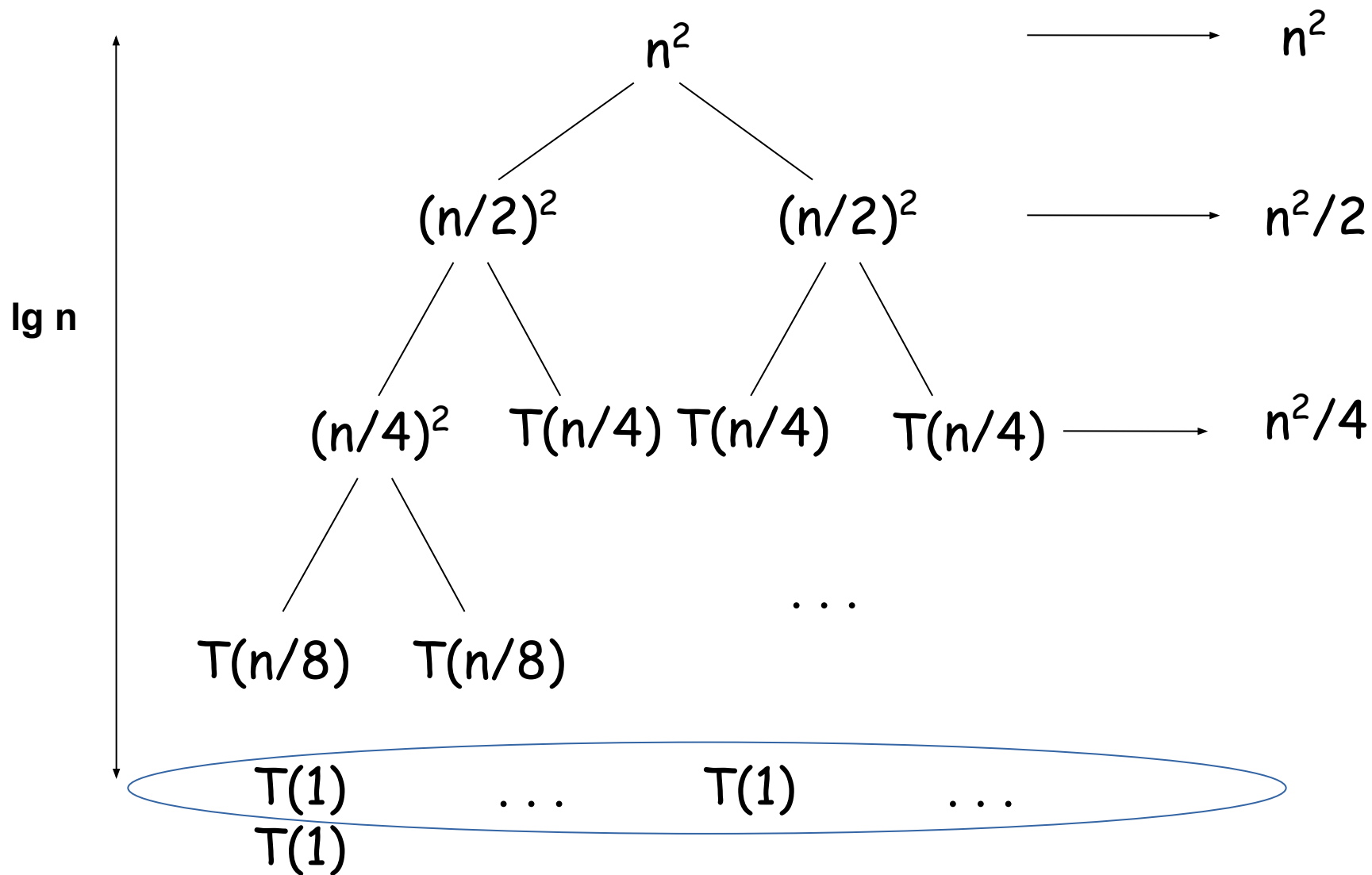
# Recurrencias



# Recurrencias

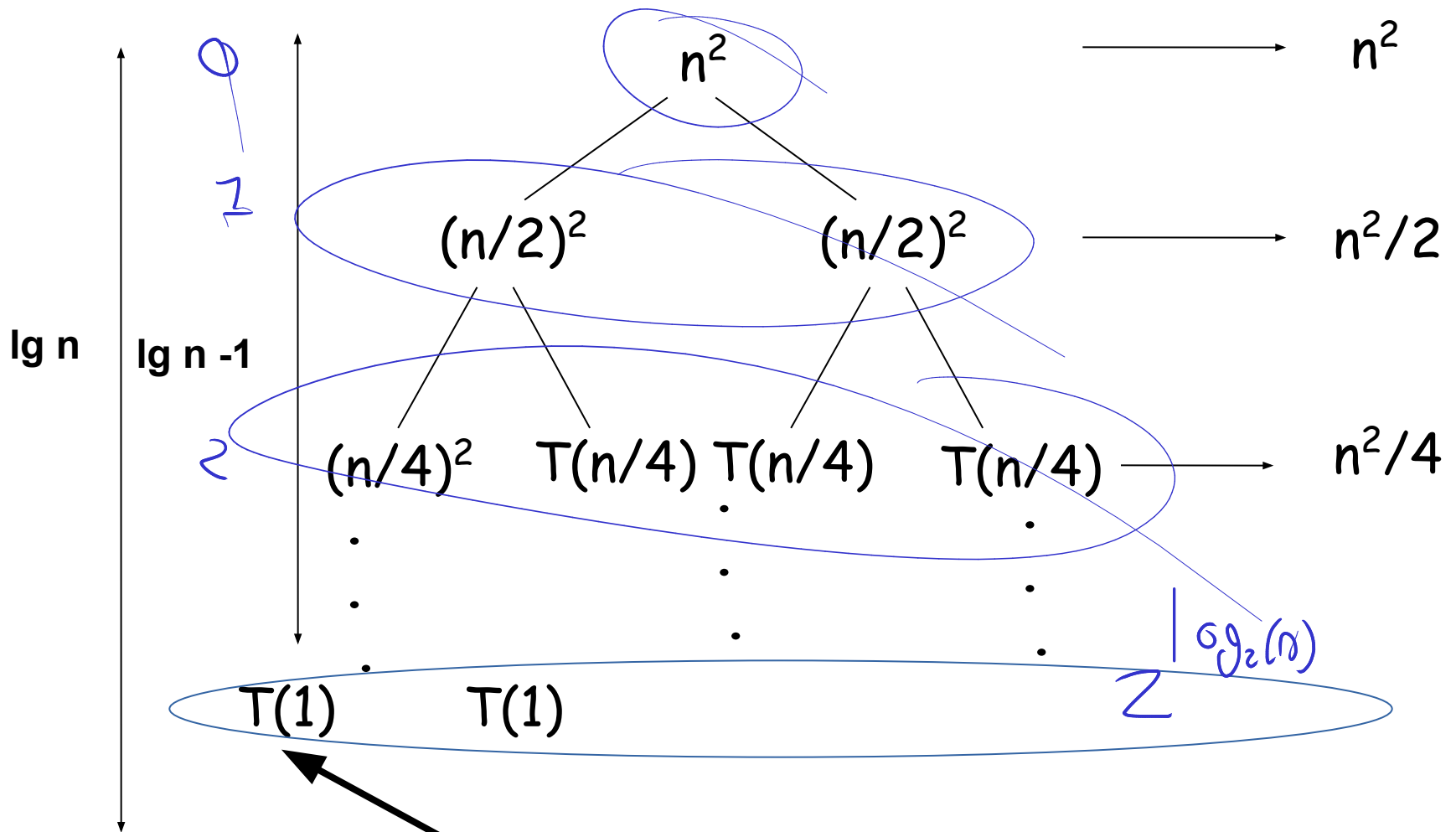


# Recurrencias



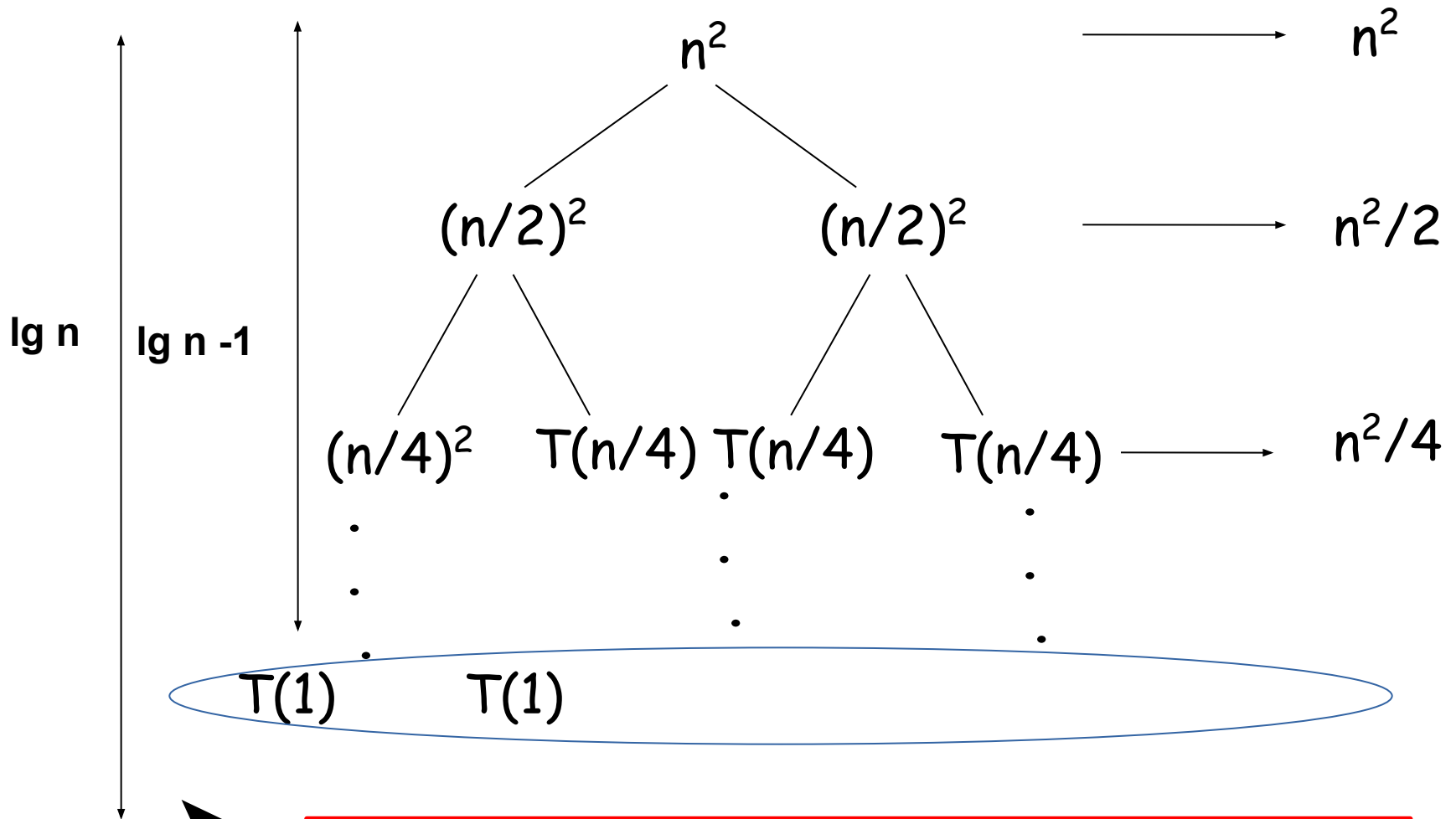


# Recurrencias



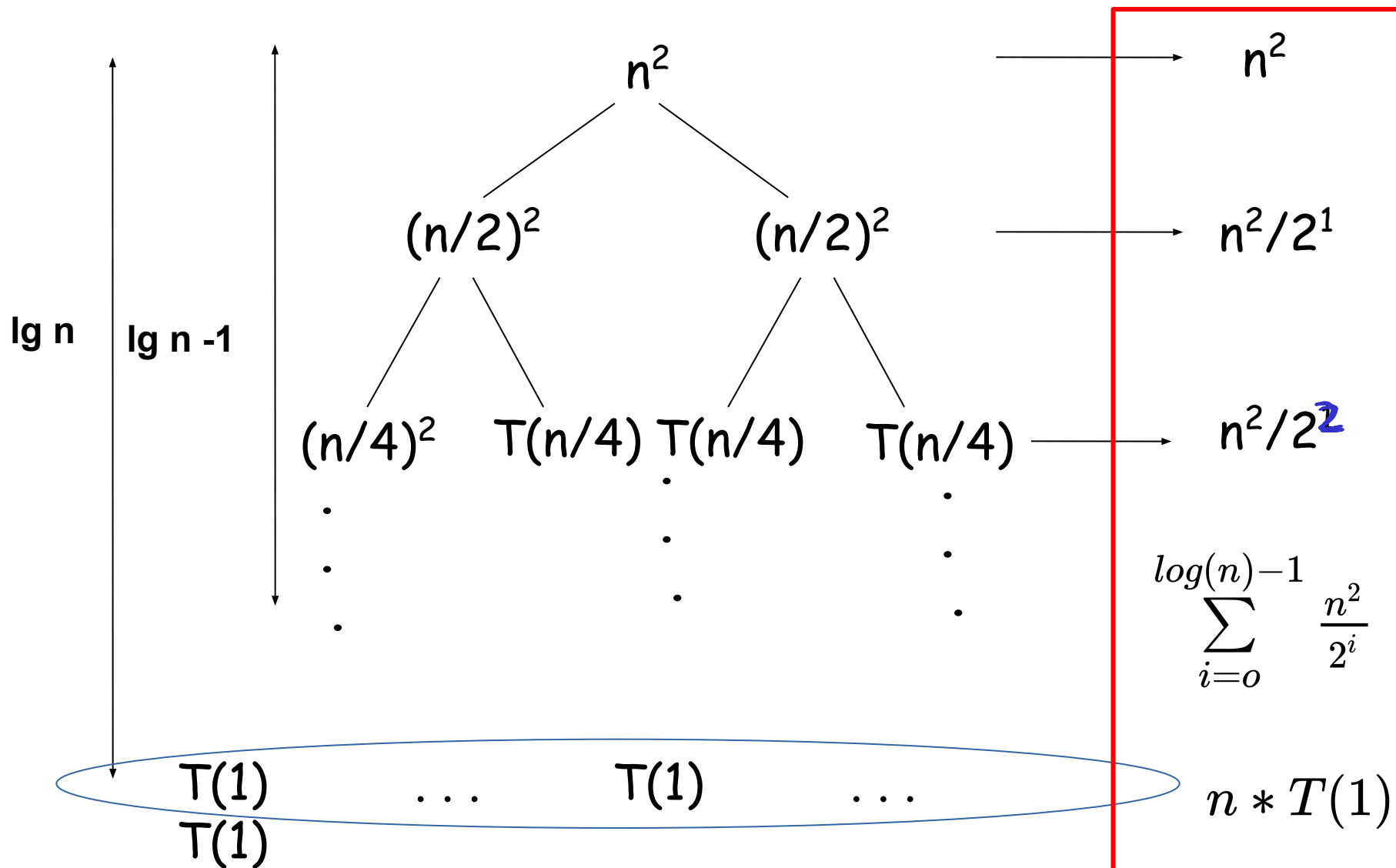
¿Cuántas hojas  $T(1)$   
tenemos?

# Recurrencias



Si recuerda en un árbol m-ario se tienen máximo  $m^h$ . En este caso al ser árbol binario  $m=2$ , tenemos  $2^{\lg(n)}$  hojas. Por lo tanto se

# Recurrencias



# Recurrencias

---

$$T(n) = n * T(1) + \sum_{i=0}^{\log(n)-1} \frac{n^2}{2^i}$$

$$T(n) = n * c + n^2 \frac{0.5^{\log(n)} - 1}{0.5 - 1}$$

$$T(n) = n * c + n^2 \frac{n^{\log(0.5)} - 1}{-0.5}$$

$$T(n) = n * c + n^2 \frac{n^{-1} - 1}{-0.5}$$

$$T(n) = n * c - \frac{n}{0.5} + \frac{n^2}{0.5} = O(n^2)$$

# Recurrencias

---

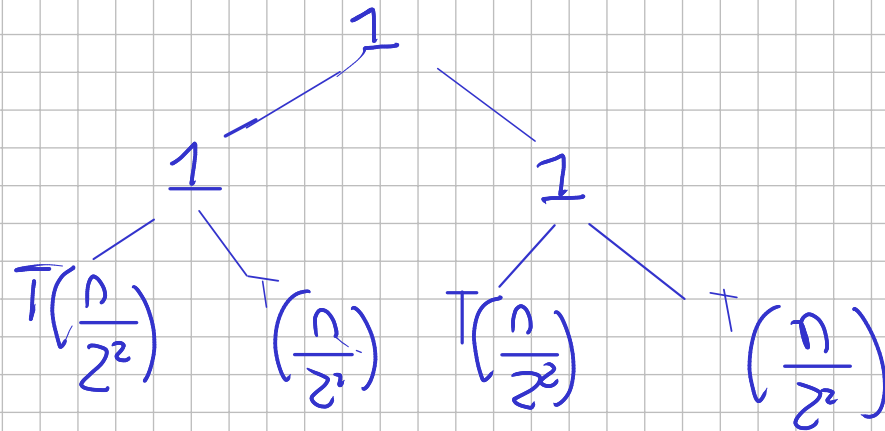
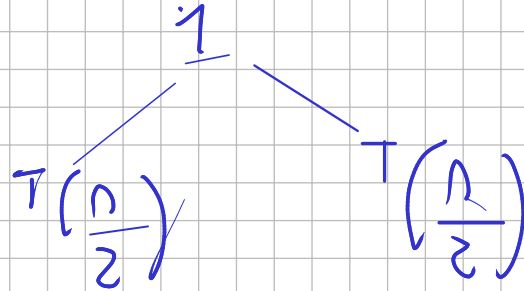
Resuelva construyendo el árbol

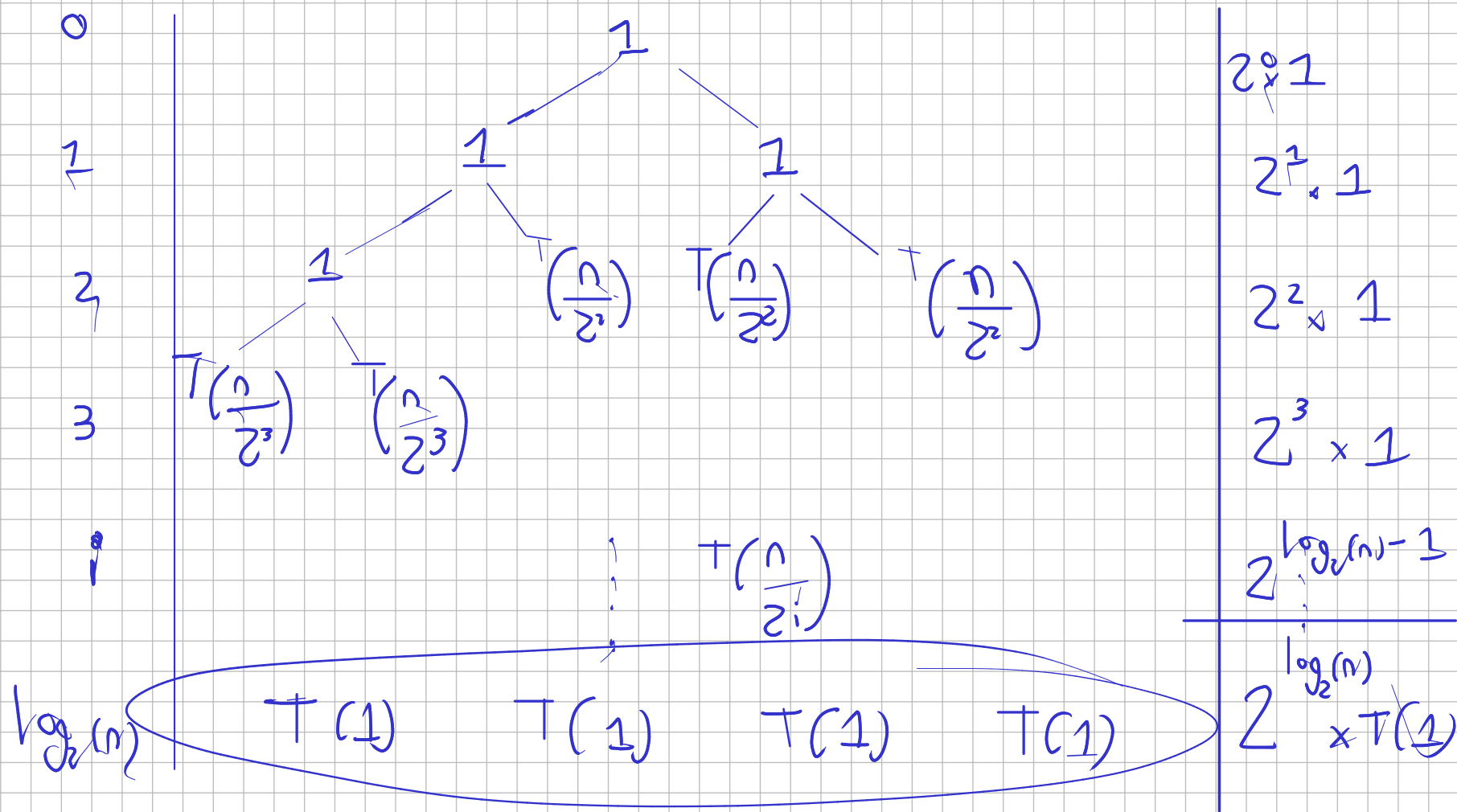
$$T(n) = 2T(n/2) + 1, T(1) = \Theta(1)$$

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



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





$$\frac{n}{2^i} \leq 1$$

$$i \geq \log_2(n)$$

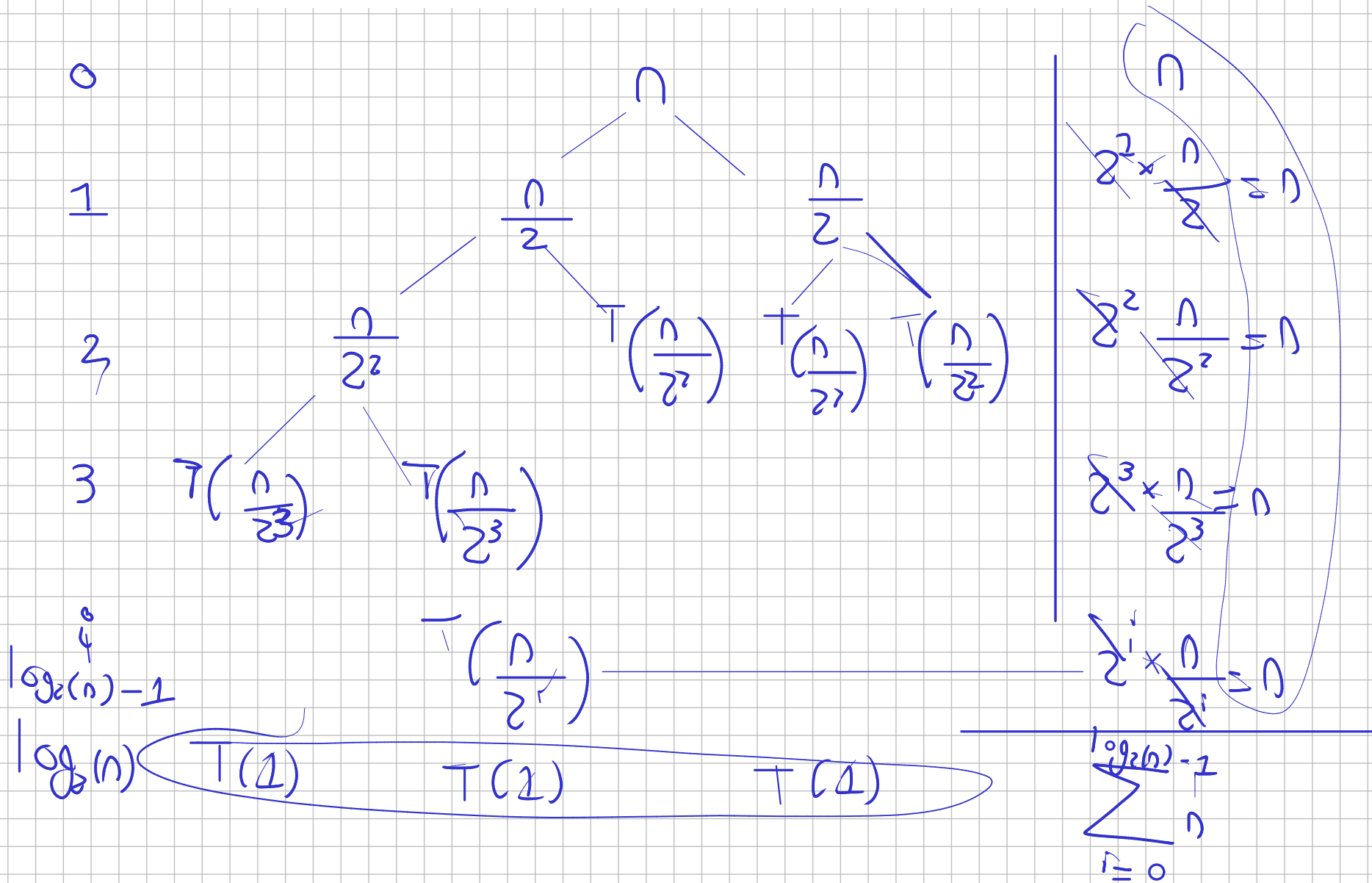
$$2^{\log_2(n)} T(1) + \sum_{i=0}^{\log_2(n)-1} 2^i$$

$$T(n) = 2^{\log_2(n)} T(1) + \sum_{i=0}^{\log_2(n)-1} 2^i$$

$$T(n) = nC + \frac{2^{\log_2(n)} - 1}{2 - 1} = \boxed{nC + n - 1}$$



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



$$T(n) = n T(1) + \sum_{i=0}^{\log_2(n)-1} n$$

$$T(n) \leq Cn + \log_2(n) \times n$$

$$\sum_{i=1}^n C \leq C \times n$$

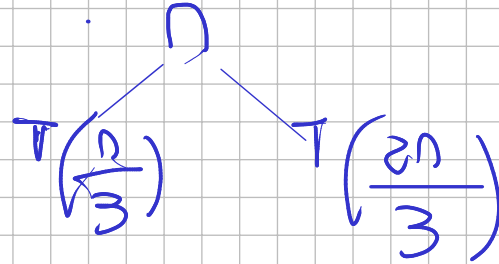
# Recurrencias

---

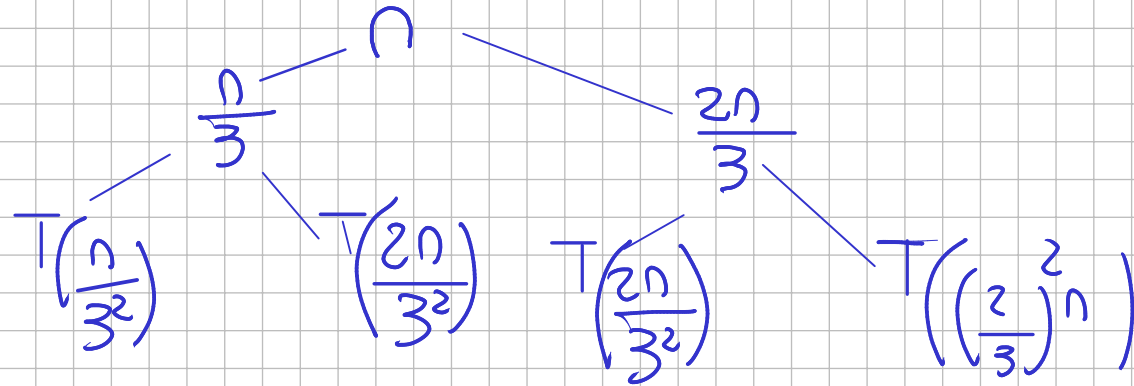
Resuelva la recurrencia  $T(n) = T(n/3) + T(2n/3) + n$

Indique una cota superior y una inferior

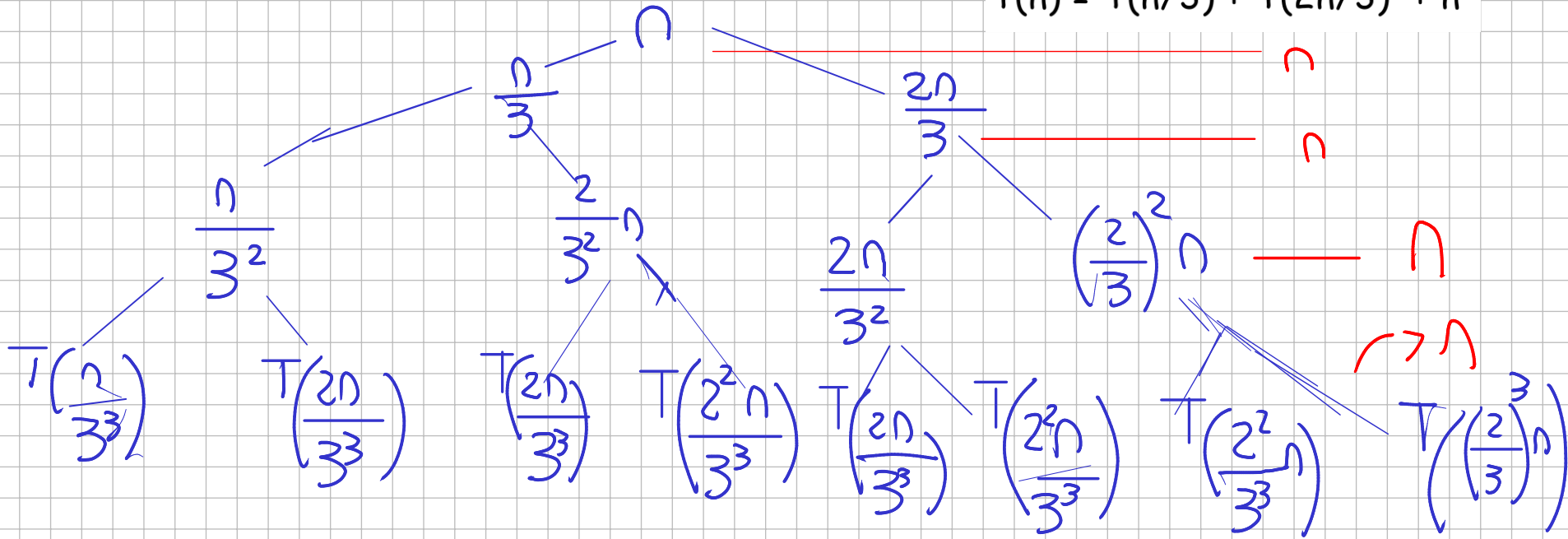
$$T(n) = T(n/3) + T(2n/3) + n$$



$$n \approx \frac{2}{3}n$$



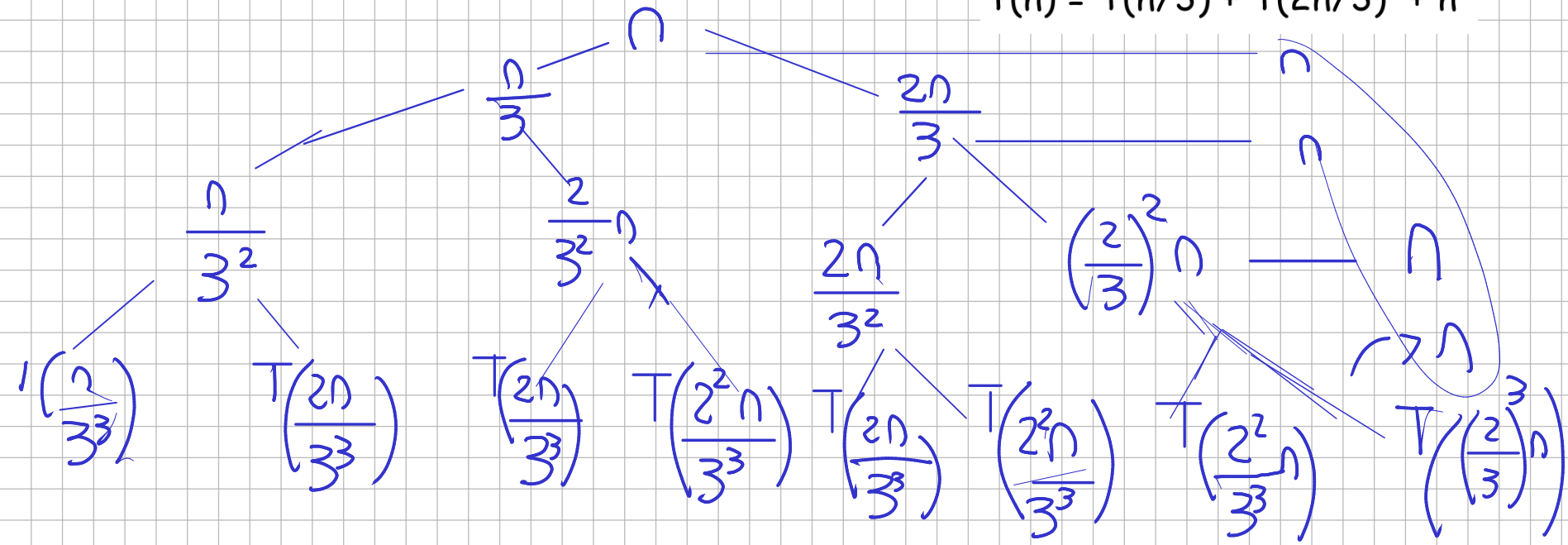
$$T(n) = T(n/3) + T(2n/3) + n$$



$$\frac{1}{3^2} + \frac{2}{3^2} + \frac{2}{3^2} + \frac{4}{3^2} = \frac{9}{3^2}$$

$$\frac{1}{3^3} + \frac{2}{3^3} + \frac{2}{3^3} + \frac{4}{3^3} + \frac{2}{3^3} + \frac{4}{3^3} + \frac{4}{3^3} + \frac{8}{3^3} = \frac{27}{3^3} = 1$$

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

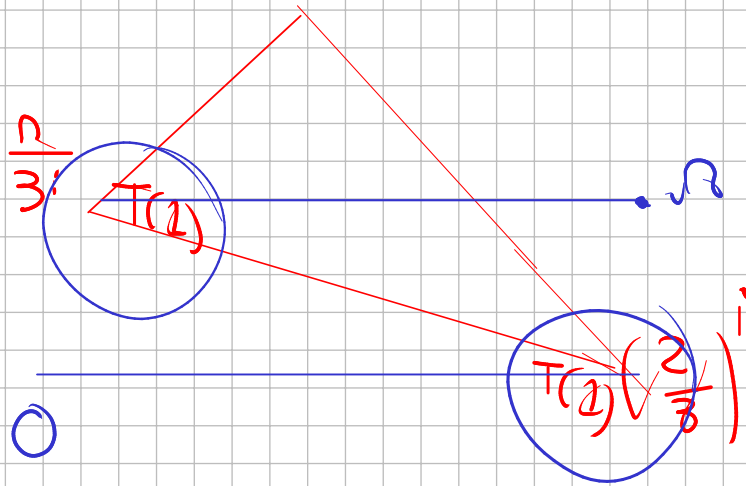


$$\frac{n}{3^i}$$

$$i_1 \leq \log_3(n)$$

$$(\frac{2}{3})^i \leq n$$

$$i \log_{\frac{2}{3}}(\frac{2}{3}) \approx \log_{\frac{2}{3}}(n)$$



$$(\frac{2}{3})^i n$$

$$i_2 \approx \log_{\frac{2}{3}}(n)$$

$$2^{\log_3(n)} T(1) + \sum_{i=0}^{\log_3(n)-1} n$$

$$n^{\log_3(2)} + \log_3(n) \times n$$

Cota inferior

$$\log_3(n) = \frac{\log_2(n)}{\log_3(2)}$$

$$2^{\log_{\frac{2}{3}}(n)} + \sum_{i=0}^{\log_{\frac{2}{3}}(n)-1} n$$

$$C \times \log(n) \times n$$

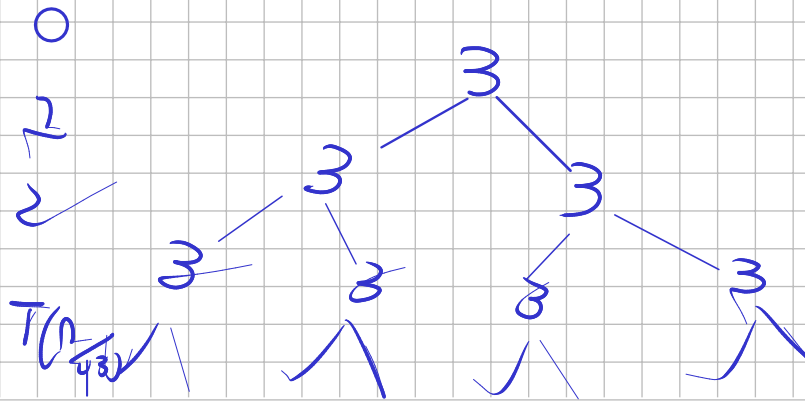
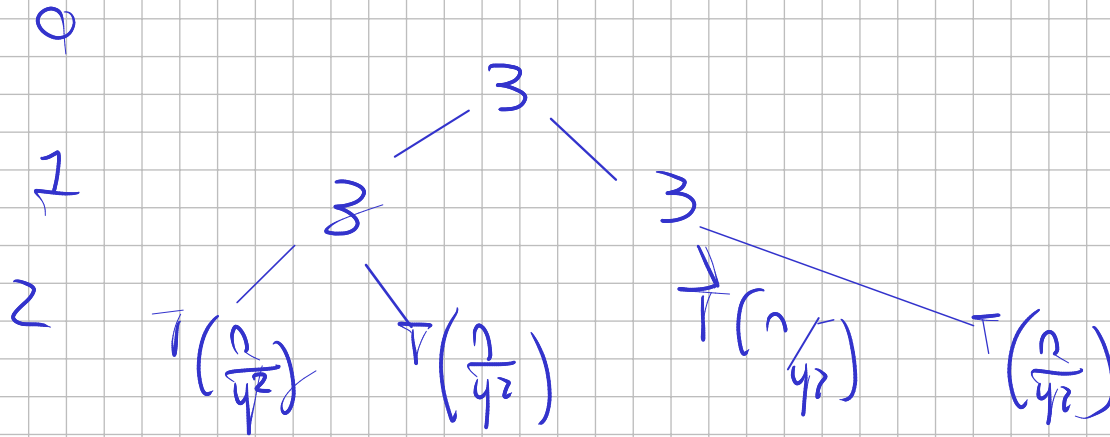
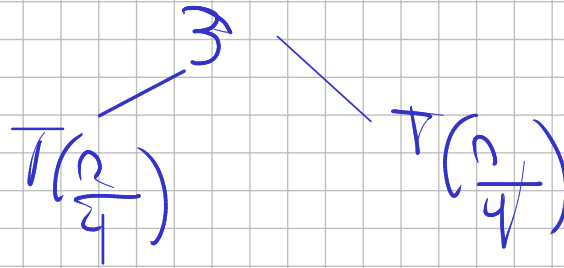
$$n^{\log_{\frac{2}{3}}(2)} + \log_{\frac{2}{3}}(n) \times n$$

Cota Superior

$$\log_{\frac{2}{3}}(n) = \frac{\log_2(n)}{\log_2(\frac{2}{3})}$$

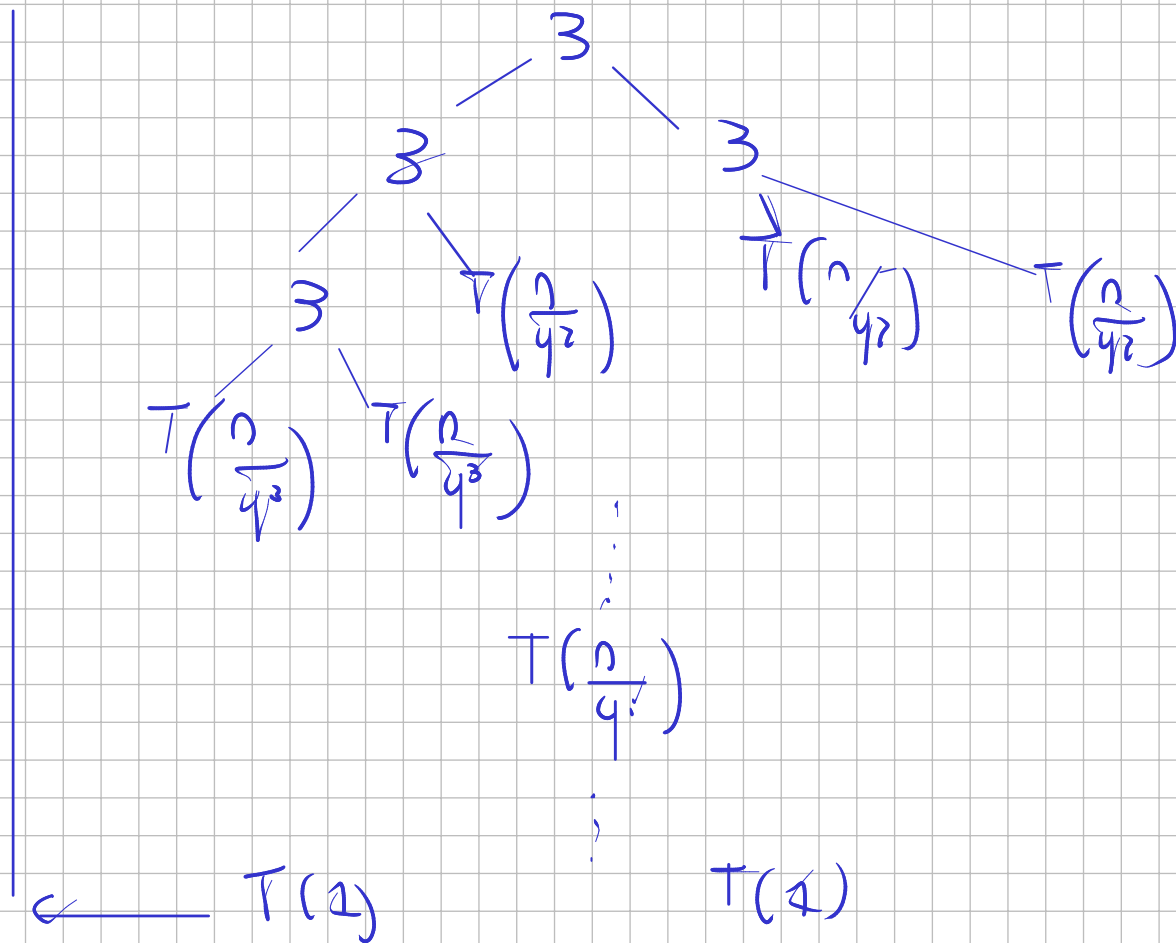
$$\Theta(n \log(n))$$

$$T(n) = 2T(n/4) + 3$$

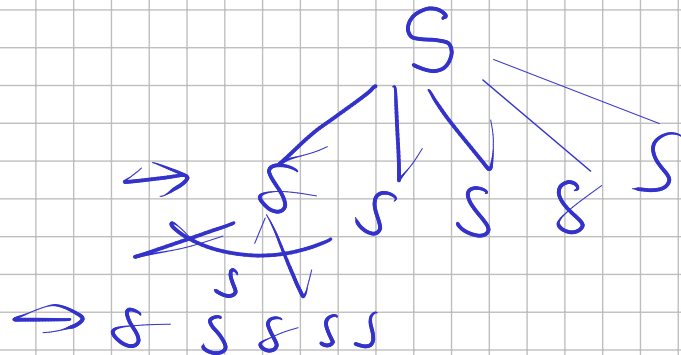
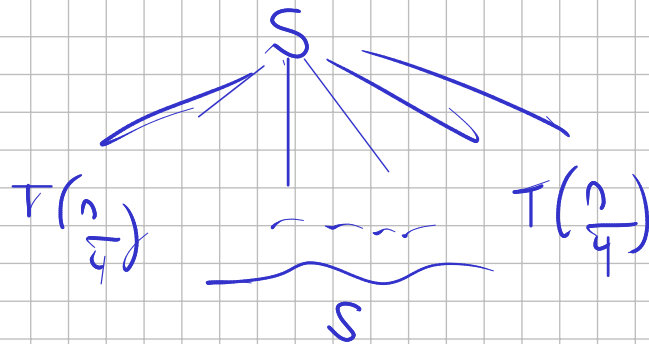


$2^0 \times 3$
$2^2 \times 3'$
$2^2 \times 3$





$$1) T(n) = 5T(n/4) + 5$$

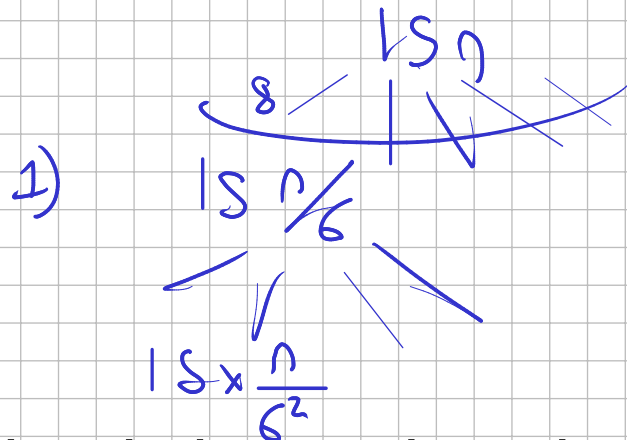


$$S^0 \times S$$

$$S^1 \times S = S^2$$

$$S^2 \times S = S^3$$

$$2) T(n) = 8T(n/6) + 15n$$



$$S \times 1S \frac{n}{6}$$

$$3) T(n) = T(n-1) + n$$

$$T(n-1)$$

$$T(n-2)$$

# Recurrencias

---

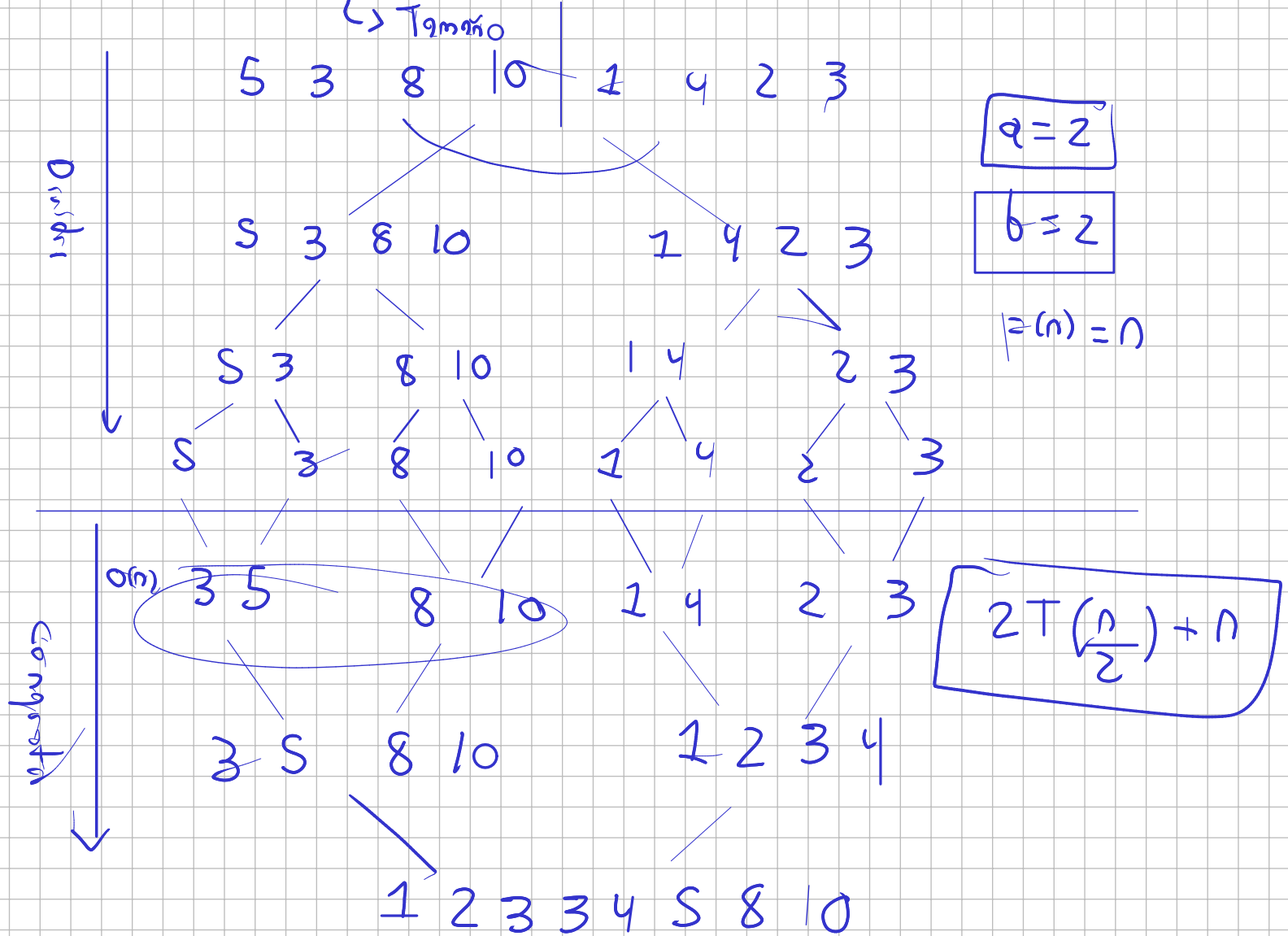
## Método maestro

Permite resolver recurrencias de la forma:

$$T(n) = aT(n/b) + f(n), \text{ donde } a \geq 1, b > 1$$

$T(n) = \boxed{a} T\left(\frac{n}{b}\right) + \text{Const}$

\* prob  
 Constante  
 Divide y vencerás  
 Tamaño



# Recurrencias

---

Dado  $T(n) = aT(n/b) + f(n)$ , donde  $a \geq 1$ ,  $b > 1$ , se puede acotar asintóticamente como sigue:

1.  $T(n) = \Theta(n^{\log_b a})$

Si  $f(n) = O(n^{\log_b a - \varepsilon})$  para algún  $\varepsilon > 0$

2.  $T(n) = \Theta(n^{\log_b a} \lg n)$

Si  $f(n) = \Theta(n^{\log_b a})$  para algún  $\varepsilon > 0$

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

Si  $f(n) = \Omega(n^{\log_b a + \varepsilon})$  para algún  $\varepsilon > 0$  si  $a * f(n/b) \leq c * f(n)$

para algún  $c < 1$

# Recurrencias

---

Dado  $T(n) = 9T(n/3) + n$

$$n^{\log_3 9} = n^2 \quad \text{Vs} \quad f(n) = n$$

$$\text{Es } f(n) = O(n^{\log_b a - \varepsilon}) \quad ?$$

$$\text{Es } n = O(n^{2 - \varepsilon}) \quad ?$$

# Recurrencias

---

Dado  $T(n) = 9T(n/3) + n$

$$n^{\log_3 9} = n^2 \quad \text{vs} \quad f(n) = n$$

Es  $f(n) = O(n^{\log_b a - \varepsilon})$  ?

Es  $n = O(n^{2-\varepsilon})$  ?

Si  $\varepsilon = 1$  se cumple que  $n = O(n)$  , por lo tanto, se cumple que:

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

# Dado $T(n) = 9T(n/3) + n$

$a=9$     $b=3$     $F(n)=n$

$T(1) = 10$

1)  $n$  es  $O(n^{\log_3 9 - \epsilon})$   
 $n$  es  $O(n^{2 - \epsilon})$     $\epsilon = 0.5$   
 $n$  es  $O(n^{1.5})$  ✓

$T(n) = \Theta(n^{\log_3 9})$

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

$$T(n) = 9^i T\left(\frac{n}{3^i}\right) + \left[\left(\frac{9}{3}\right)^{i-1} n + \left(\frac{9}{3}\right)^{i-2} n + \dots + \left(\frac{9}{3}\right)^0 n\right]$$

$$T(n) = 9^{\log_3 n} T(1) + \sum_{i=0}^{\log_3 n - 1} 3^i n$$

1.  $T(n) = \Theta(n^{\log_b a})$

Si  $f(n) = O(n^{\log_b a - \epsilon})$  para algún  $\epsilon > 0$

2.  $T(n) = \Theta(n^{\log_b a} \lg n)$

Si  $f(n) = \Theta(n^{\log_b a})$  para algún  $\epsilon > 0$

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

Si  $f(n) = \Omega(n^{\log_b a + \epsilon})$  para algún  $\epsilon > 0$  si  $a \neq f(n/b)$   
 $\leq c \cdot f(n)$     $c < 1$

$$T(n) = 9T(n/3) + n$$

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

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

$$T(n) = n^2 T(1) + \frac{3^{\log_3(n)} - 1}{3 - 1}$$

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



# Recurrencias

---

$$T(n) = T(2n/3) + 1$$

$$n^{\log_{3/2} 1} = n^0 = 1 \quad \text{Vs} \quad f(n) = 1$$

$$\text{Es } f(n) = O(n^{\log_b a - \varepsilon}) \quad ?$$

$$\text{Es } 1 = O(n^{0 - \varepsilon}) \quad ?$$

No existe  $\varepsilon > 0$

# Recurrencias

---

$$T(n) = T(2n/3) + 1$$

$$n^{\log_{3/2} 1} = n^0 = 1 \quad \text{vs} \quad f(n) = 1$$

$$\text{Es } f(n) = \Theta(n^{\log_b a}) \quad ?$$

$$\text{Es } 1 = \Theta(1) \quad ?$$

Si, por lo tanto, se cumple que:

$$T(n) = \Theta(1 * \lg n) = \Theta(\lg n)$$

$$T(n) = T\left(\frac{2n}{3}\right) + 1$$

$$a=1$$

$$f(n)=1$$

$$b=\frac{3}{2}$$

$$2) 1 \in \Theta(n^{\log_{\frac{3}{2}} 1 - \epsilon})$$

$$\times 1 \in \Theta(n^{0 - \epsilon})$$

$$1 \in \Theta\left(\frac{1}{n^\epsilon}\right)$$

$$2) \in \Theta(1) \in \Theta(n^{\log_{\frac{3}{2}} 1})$$

$$1 \in \Theta(1)$$

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

$$T(n) = \Theta(n^0 \log(n))$$

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

$$1. T(n) = \Theta(n^{\log_b a})$$

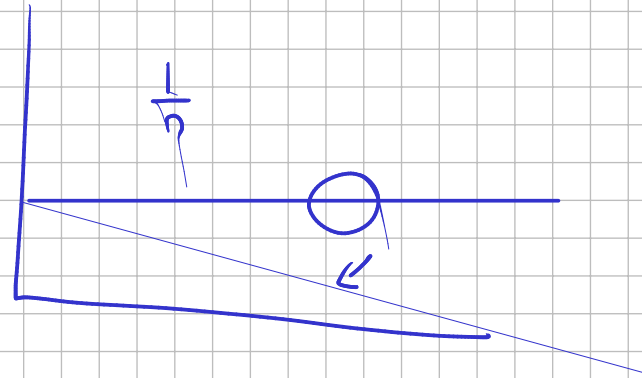
$$\text{Si } f(n) = O(n^{\log_b a - \epsilon}) \text{ para algún } \epsilon > 0$$

$$2. T(n) = \Theta(n^{\log_b a} \lg n)$$

$$\text{Si } f(n) = \Theta(n^{\log_b a}) \text{ para algún } \epsilon > 0$$

$$3. T(n) = \Theta(f(n))$$

$$\text{Si } f(n) = \Omega(n^{\log_b a + \epsilon}) \text{ para algún } \epsilon > 0 \text{ si } a * f(n/b) \leq c * f(n) \quad c < 1$$



# Recurrencias

---

$$T(n) = 3 T(n/4) + n \lg n$$

$$n^{\log_4 3} = n^{0.793} \quad \text{vs} \quad f(n) = n \lg n$$

$$\text{Es } f(n) = O(n^{\log_b a - \epsilon}) \quad ?$$

$$\text{Es } f(n) = \Theta(n^{\log_b a}) \quad ?$$

$$\text{Es } f(n) = \Omega(n^{\log_b a + \epsilon}) \quad ?$$

Si, y además,  $af(n/b) \leq cf(n)$

$$3(n/4) \lg(n/4) \leq cn \lg n$$

$$3(n/4) \lg n - 3(n/4) \cdot 2 \leq cn \lg n$$

$$(3/4)n \lg n \leq cn \lg n \rightarrow c = 3/4 \text{ y se concluye } T(n) = \Theta(n \lg n)$$

$$T(n) = 3T\left(\frac{n}{4}\right) + \log(n)$$

$$a=3 \quad b=4 \quad f(n)=\log(n)$$

$$1) \quad n \log(n) \neq O(n^{\log_4 3 - \epsilon})$$

$$\times \quad n \log(n) \text{ es } O(n^{0.793 - \epsilon}) \quad \epsilon = 0.093$$

$$2) \quad n \log(n) \neq \Theta(n^{0.793}) \quad \times$$

$$3) \quad n \log(n) \text{ es } \Omega(n^{0.793 + \epsilon}) \quad \epsilon = 0.207$$

$$n \log(n) \text{ es } \Omega(n) \quad \checkmark$$

$$3\left(\frac{n}{4} \log\left(\frac{n}{4}\right)\right) \leq c n \log(n)$$

$$\frac{3}{4} \log\left(\frac{n}{4}\right) \leq c \log(n)$$

$$\frac{3}{4} \log(n) \leq c \log(n)$$

$$\frac{3}{4} n \log\left(\frac{n}{4}\right) \leq c n \log(n)$$

$$\frac{3}{4} \log(n) - \frac{3}{4} \log(4) \leq c \log(n)$$

$$\frac{3}{4} \log(n) - \frac{3}{2} \leq c \log(n)$$

$$n \gg$$

$$c \geq \frac{3}{4}$$

$$\left[\frac{3}{4}, 1\right)$$

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

$$1. \quad T(n) = \Theta(n^{\log_b a})$$

$$\text{Si } f(n) = O(n^{\log_b a - \epsilon}) \quad \text{para algún } \epsilon > 0$$

$$2. \quad T(n) = \Theta(n^{\log_b a} \lg n)$$

$$\text{Si } f(n) = \Theta(n^{\log_b a}) \quad \text{para algún } \epsilon > 0$$

$$3. \quad T(n) = \Theta(f(n))$$

$$\text{Si } f(n) = \Omega(n^{\log_b a + \epsilon}) \quad \text{para algún } \epsilon > 0 \quad \text{si } a \cdot f(n/b) \leq c \cdot f(n) \quad c < 1$$

# Recurrencias

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

$$\log_b a = \log_2 2 = 1$$

Muestre que no se puede resolver p

$$1) n \log(n) \text{ es } O(n^{1-\epsilon}) \times$$

$$2) n \log(n) \text{ es } \Theta(n^2) \times$$

$$3) n \log(n) \text{ es } \Omega(n^{1+\epsilon})$$

$$\Omega(n^{1+1})$$

$$2) \frac{n}{2} \log\left(\frac{n}{2}\right) \leq c \cdot n \log(n) \Rightarrow \frac{n \log(n)}{2} \leq c \cdot n \log(n)$$

$$c \geq 1 \quad c < 1 \times$$

$$1. T(n) = \Theta(n^{\log_b a - \epsilon})$$

$$\text{Si } f(n) = O(n^{\log_b a - \epsilon}) \text{ para algún } \epsilon > 0$$

$$2. T(n) = \Theta(n^{\log_b a} \lg n)$$

$$\text{Si } f(n) = \Theta(n^{\log_b a}) \text{ para algún } \epsilon > 0$$

$$3. T(n) = \Theta(f(n))$$

$$\text{Si } f(n) = \Omega(n^{\log_b a + \epsilon}) \text{ para algún } \epsilon > 0 \text{ si } a \neq f(n/b)$$

$$\leq c \cdot f(n)$$

$$c < 1$$

# Recurrencias

Resuelva usando método del maestro

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

*Handwritten notes: a=4, b=2*

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

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

1.  $T(n) = \Theta(n^{\log_b a})$

Si  $f(n) = O(n^{\log_b a - \epsilon})$  para algún  $\epsilon > 0$

2.  $T(n) = \Theta(n^{\log_b a} \lg n)$

Si  $f(n) = \Theta(n^{\log_b a})$  para algún  $\epsilon > 0$

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

Si  $f(n) = \Omega(n^{\log_b a + \epsilon})$  para algún  $\epsilon > 0$  si  $a \cdot f(n/b) \leq c \cdot f(n)$  *Handwritten note: c < 1*

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

$$a=4 \quad b=2 \quad f(n)=n \quad T(n) = \Theta(n^2)$$

$$n \in O(n^{2-\epsilon})$$

$$n \notin O(n) \quad \epsilon=1$$

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

$$1) \quad n^2 \in O(n^{2-\epsilon}) \quad \times$$

$$2) \quad n^2 \in \Theta(n^2) \quad \checkmark$$

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

$$1. \quad T(n) = \Theta(n^{\log_b a})$$

$$\text{Si } f(n) = O(n^{\log_b a - \epsilon}) \quad \text{para algún } \epsilon > 0$$

$$2. \quad T(n) = \Theta(n^{\log_b a} \lg n)$$

$$\text{Si } f(n) = \Theta(n^{\log_b a}) \quad \text{para algún } \epsilon > 0$$

$$3. \quad T(n) = \Theta(f(n))$$

$$\text{Si } f(n) = \Omega(n^{\log_b a + \epsilon}) \quad \text{para algún } \epsilon > 0 \quad \text{si } a^* f(n/b) \leq c^* f(n)$$

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

$$1) \quad n^3 \in O(n^{2-\epsilon}) \quad \times$$

$$2) \quad n^3 \in \Theta(n^2) \quad \times$$

$$3) \quad n^3 \in \Omega(n^{2+\epsilon}) \quad \checkmark$$

$$4 \times \left(\frac{n}{2}\right)^3 \leq c \times n^3$$

$$\frac{1}{2} n^3 \leq c n^3$$

$$c \geq \frac{1}{2} \quad \left[\frac{1}{2}, 1\right) \quad \checkmark$$

$$\Theta(n^3)$$



# Recurrencias

---

Método de sustitución

Suponer la forma de la solución y probar por inducción matemática

# Recurrencias

---

$$T(n) = 2T(\lfloor n/2 \rfloor) + n, \quad T(1) = 1$$

Suponer que la solución es de la forma  $T(n) = O(n \lg n)$

Probar que  $T(n) \leq cn \lg n$ .

Se supone que se cumple para  $n/2$  y se prueba para  $n$

Hipotesis inductiva:  $T(n/2) \leq cn/2 \lg(n/2)$

# Recurrencias

---

$$T(n) = 2T(\lfloor n/2 \rfloor) + n, \quad T(1) = 1$$

Probar que  $T(n) \leq cn \lg n$ .

Hipótesis inductiva:  $T(n/2) \leq cn/2 \lg (n/2)$

Paso inductivo:

$$\begin{aligned} T(n) &\leq 2(cn/2 \lg (n/2)) + n \\ &\leq cn \lg (n/2) + n \\ &= cn \lg (n) - cn + n, \text{ para } c \geq 1, \text{ haga } c=1 \\ &\leq cn \lg n \end{aligned}$$

# Recurrencias

---

$$T(n) = 2T(\lfloor n/2 \rfloor) + n, \quad T(1) = 1$$

Probar que  $T(n) \leq cn \lg n$ .

Paso base: si  $c=1$ , probar que  $T(1)=1$  se cumple

$$T(1) \leq 1 * 1 \lg 1 ?$$

$$1 \leq 0 ?$$

No, se debe escoger otro valor para  $c$

# Recurrencias

---

$$T(n) = 2T(\lfloor n/2 \rfloor) + n, \quad T(1) = 1$$

Probar que  $T(n) \leq cn \lg n$ .

Paso base: si  $c=2$ , probar que  $T(1)=1$  se cumple

$$T(1) \leq 2 * 1 \lg 1 ?$$

$$1 \leq 0 ?$$

No, se puede variar  $k$ .

Para esto, se calcula  $T(2)$  y se toma como valor inicial

# Recurrencias

---

Probar que  $T(n) \leq cn \lg n$ .

$$T(2) = 2T(0) + 2 = 4$$

Paso base: si  $c=1$ , probar que  $T(2)=4$  se cumple

$$T(2) \leq 1 * 2 \lg 2 ?$$

$$4 \leq 2 ?$$

No, se puede variar  $c$ .

# Recurrencias

---

Probar que  $T(n) \leq cn \lg n$ .

$$T(2) = 2T(0) + 2 = 4$$

Paso base: si  $c=3$ , probar que  $T(2)=4$  se cumple

$$T(2) \leq 3 \cdot 2 \lg 2 ?$$

$$4 \leq 6 ?$$

Si, se termina la demostración

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

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

Per induzione  $T(n) \leq C \times n$

$$T(1) = 3$$

$$T(n) \leq C \cdot \frac{n}{2} + 1$$

$$C = 1$$

$$3 \leq \frac{1}{2} + 1 \quad \times$$

$$C = 10$$

$$3 \leq \frac{10n}{2} + 1$$

$$3 \leq 4 \quad \checkmark$$

$n$	$T(n)$	$5 \frac{n}{2} + 1$
1	3	6
2	7	11
4	15	21
8	31	41
16	63	81
32	127	161



$$T(n) = 4T\left(\frac{n}{2}\right) + n^2 \quad T(1) = 0$$

$$a=4 \quad b=2 \quad \log_2 4 = 2$$

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

$$T(n) \in \Theta(n^2 \log(n))$$

$$T(n) \leq C \cdot n^2 \log(n)$$

$$1. T(n) = \Theta(n^{\log_b a})$$

$$\text{Si } f(n) = O(n^{\log_b a - \epsilon}) \quad \text{para alg\'un } \epsilon > 0$$

$$2. T(n) = \Theta(n^{\log_b a} \lg n)$$

$$\text{Si } f(n) = \Theta(n^{\log_b a}) \quad \text{para alg\'un } \epsilon > 0$$

$$3. T(n) = \Theta(f(n))$$

$$\text{Si } f(n) = \Omega(n^{\log_b a + \epsilon}) \quad \text{para alg\'un } \epsilon > 0 \quad \text{si } a \cdot f(n/b) \leq c \cdot f(n)$$

$$T(n) \geq c n^2 \log(n)$$

n  
1  
2  
4  
8  
16  
32

# Recurrencias

---

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

Suponer que la solución es de la forma  $T(n)=O(2^n)$

Probar que  $T(n) \leq c2^n$ .

Se supone que se cumple para  $n-1$  y se  $n-2$  prueba para  $n$

Hipotesis inductiva:  $T(n-1) \leq c2^{(n-1)}$  y  $T(n-2) \leq c2^{(n-2)}$

# Recurrencias

---

$$T(n) = T(n-1) + T(n-2) + 1, \quad T(1) = O(1), \quad T(2) = O(1)$$

Ahora se debe probar que:  $T(n) \leq c2^n$

$$T(1) \leq c2^1 \rightarrow 1 \leq 2 * c$$

$$T(2) \leq c2^2 \rightarrow 1 \leq 4 * c$$

$$T(3) \leq c2^3 \rightarrow 2 \leq 8 * c$$

$$T(4) \leq c2^4 \rightarrow 3 \leq 16 * c$$

$$T(5) \leq c2^5 \rightarrow 5 \leq 32 * c$$

$$T(6) \leq c2^6 \rightarrow 8 \leq 64 * c$$

$$T(7) \leq c2^7 \rightarrow 13 \leq 128 * c$$

$$T(8) \leq c2^8 \rightarrow 21 \leq 256 * c$$

Con  $c = 1$ , se cumple.

# Referencias

---

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. Introduction to Algorithms, Third Edition (3rd ed.). The MIT Press. Chapter 4

# Gracias

---

Próximo tema:

Divide y vencerás