Quicksort

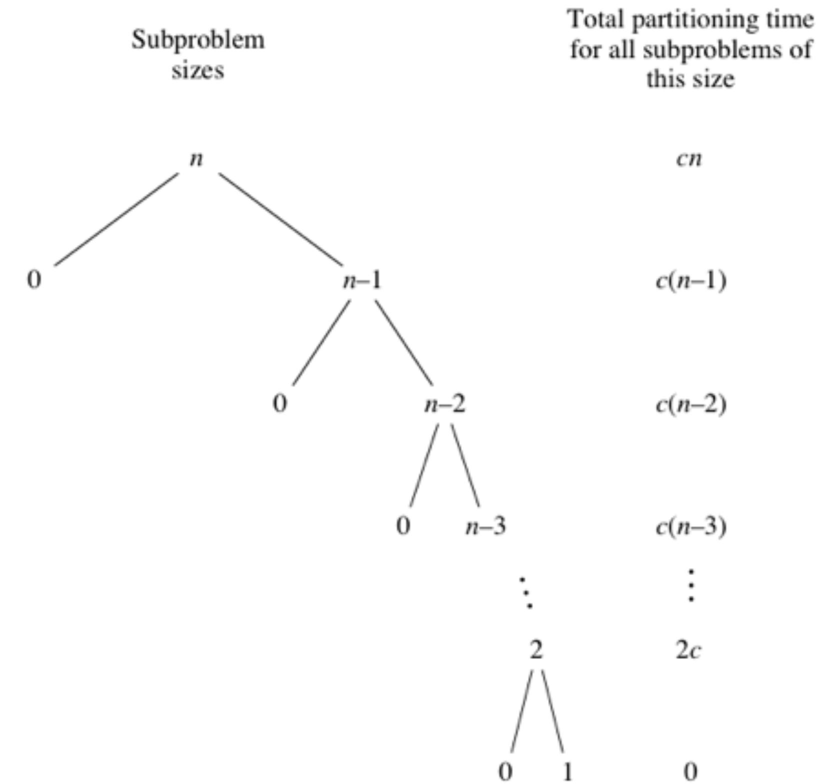
Análisis (informal) de Eficiencia

Análisis del peor caso

- En el peor caso, la función de partición proporciona como pivote el máximo o el mínimo del conjunto
- Una de las particiones no contendrá elementos y la otra contendrá n-1 (todos salvo el pivote)
- Tendremos, por lo tanto, llamadas recursivas a subarrays de tamaño 0 y n-1, respectivamente

- Cada llamada recursiva sobre un array de tamaño n es O(n) (el proceso de partición)

Análisis del peor caso



$$cn + c(n-1) + c(n-2) + \cdots + 2c =$$
$$c(n + (n-1) + (n-2) + \cdots + 2) =$$
$$c((n+1)(n/2) - 1)$$

⬇

O(n²)

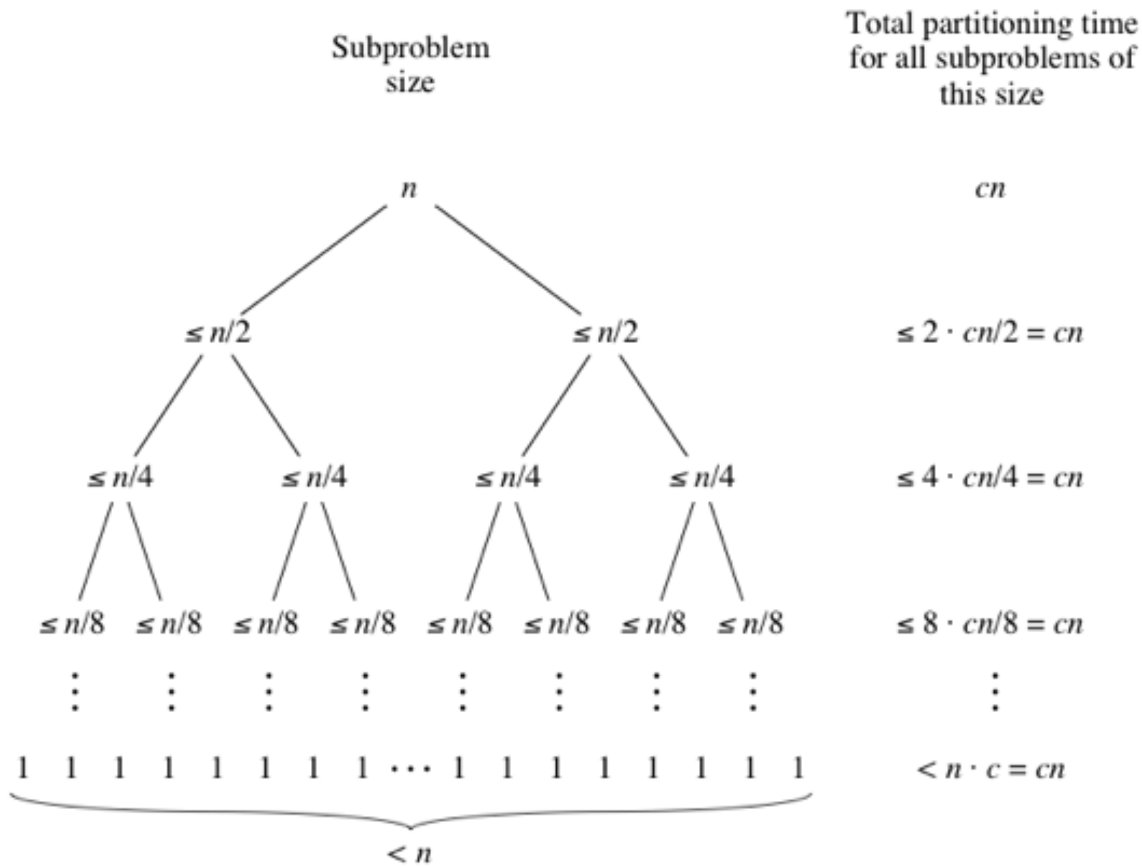
$$\sum_{1}^{n} a_i = \frac{n(a_1 + a_n)}{2} \quad ➡ \quad \sum_{1}^{n} i = \frac{n(1+n)}{2} = (n+1)(n/2)$$

Análisis del mejor caso

- En el mejor caso, la función de partición proporciona como pivote la mediana del conjunto
- Las particiones contendrán n/2 elementos (ó n/2 -1)
- Tendremos, por lo tanto, llamadas recursivas a subarrays de tamaño 0 y n-1, respectivamente

- Cada llamada recursiva sobre un array de tamaño n es O(n) (el proceso de partición)

Análisis del mejor caso



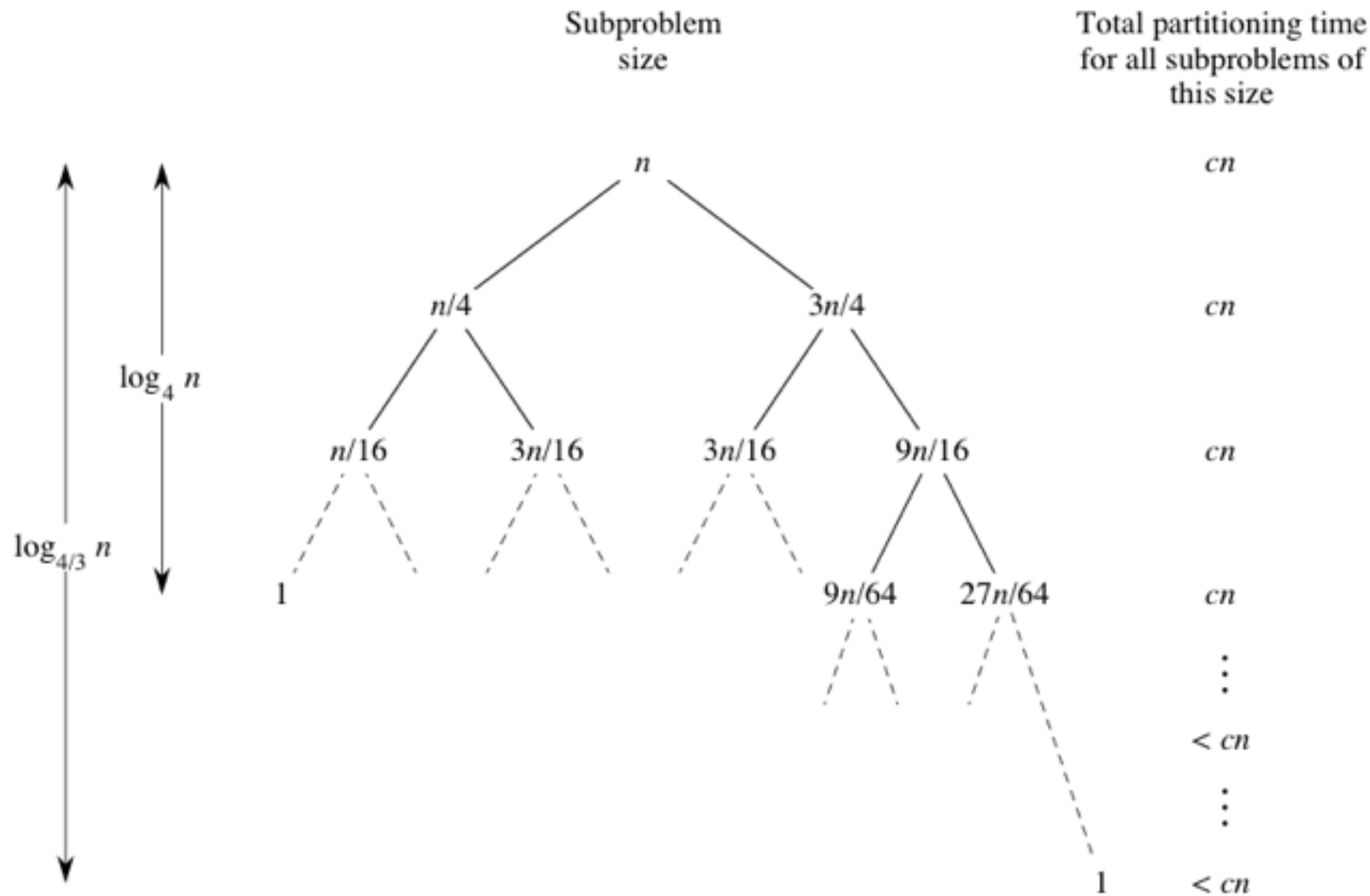
⬇

O(n $\log_2$ n)

Análisis del caso intermedio

- Resulta obvio que el caso intermedio no puede ser mejor que el mejor caso, por lo que tenemos una cota inferior de $O(n \log_2 n)$
- Supongamos que no obtenemos particiones equilibradas, pero que sí obtenemos al menos particiones 1/4 - 3/4

Análisis del caso medio



$n \log_{4/3} n$

Análisis del caso intermedio

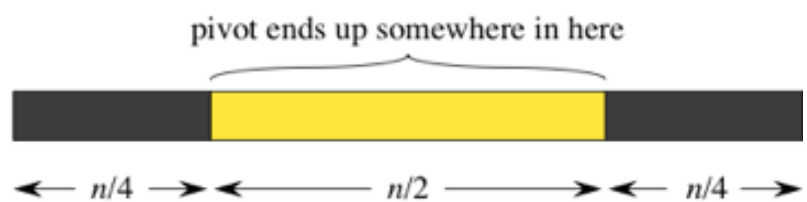
- Sabemos que

$$log_a n = \frac{log_b n}{log_b a} \quad \Longrightarrow \quad log_{4/3} n = \frac{log_2 n}{log_2 (4/3)}$$

- Por lo tanto, podemos afirmar que para una partición 1/4 - 3/4 Quicksort es O(n $\log_2$ n)

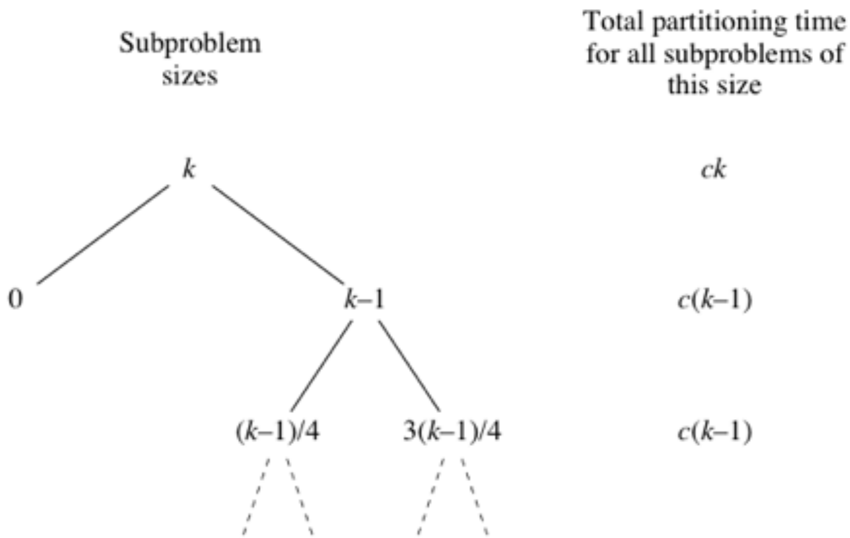
(¡Aunque con un factor constante grande!)

- ¿Qué probabilidad hay de hacer una partición 1/4 - 3/4 o mejor?

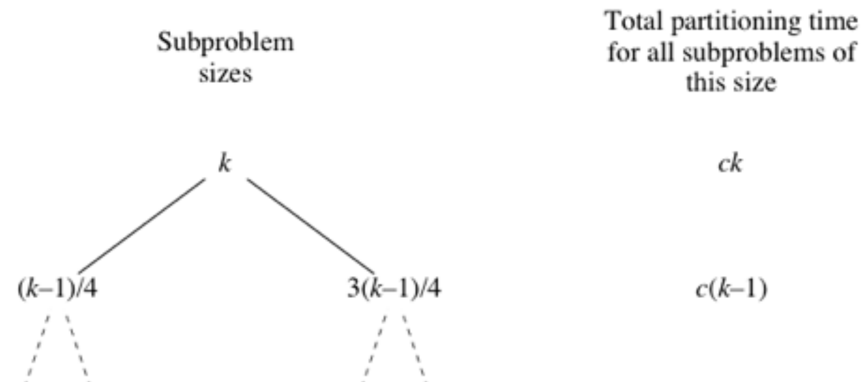


Análisis del caso intermedio

- Supongamos que se alternan el peor caso y el caso intermedio (1/4-3/4)



en lugar de:



- De nuevo, la diferencia sólo es un factor constante, por lo que podemos afirmar que es $O(n \log_2 n)$