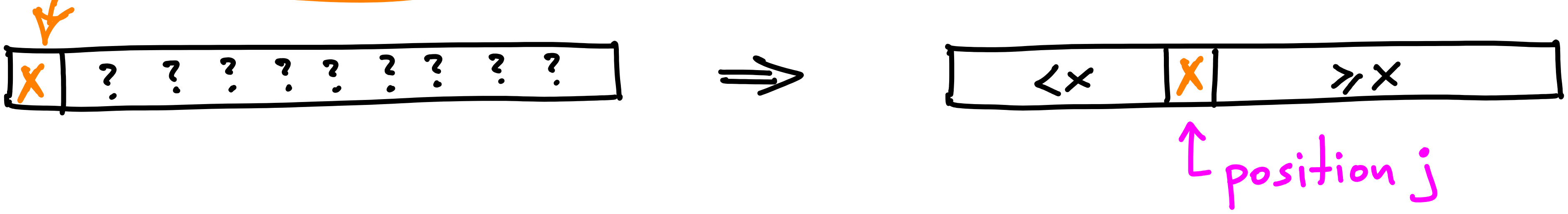
QUICKSORT

- If necessary shuffle data to make random order.

- DIVIDE: choose a pivot & partition. → $\Theta(n)$



- CONQUER: Quicksort each side.

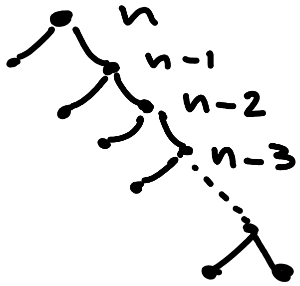
$$T(n) = \Theta(n) + T(j-1) + T(n-j)$$

$$T(n) = \Theta(n) + T(j-1) + T(n-j)$$

What is the worst-case time complexity, and why?

$\hookrightarrow$ already sorted input, reverse-sorted, nearly sorted...

$$T(n) = T(0) + T(n-1) + \Theta(n) \quad = \Theta(n^2)$$



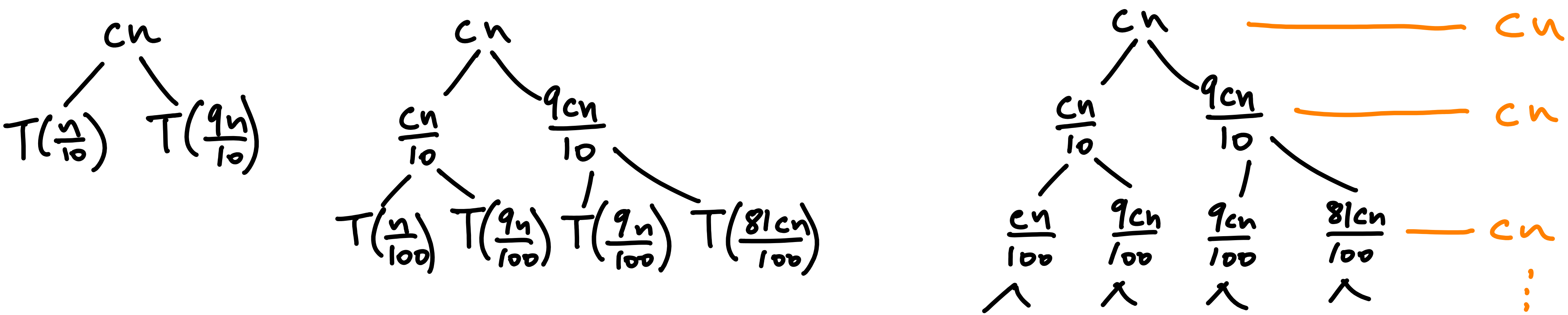
What would be ideal ?

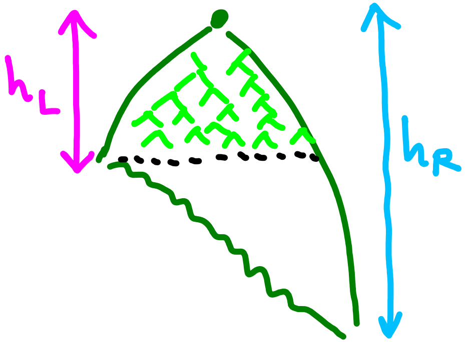
$\hookrightarrow$ ~ balanced partition, every time

$$T(n) = 2 \cdot T\left(\frac{n}{2}\right) + \Theta(n) \quad = \Theta(n \log n)$$

What if we always have a "sort-of-balanced" partition?

e.g., $T(n) = T\left(\frac{n}{10}\right) + T\left(\frac{9n}{10}\right) + c \cdot n$





$h_L \sim \log_{10} n \rightarrow T(n) \geq cn \log_{10} n$

$h_R \sim \log_{10/9} n \rightarrow T(n) \leq cn \log_{10/9} n$

Any constant-fraction-split will give $\Theta(n \log n)$

Expected time: call a partition $\to$ balanced if pivot ranks in $[\frac{n}{4} \dots \frac{3n}{4}]$

$\to$ unbalanced otherwise

$E[T(n)]$

but let's just use $T(n)$

(redefining what balanced means)

Worst case if balanced: $T(n) \leq T(\frac{3n}{4}) + T(\frac{n}{4}) + cn$

Worst case if unbalanced: $T(n) \leq T(n-1)+cn < T(n)+cn$

Each partition has a 50% chance of being balanced

$$T(n) \leq \frac{1}{2}(T(n)+cn) + \frac{1}{2}\left(T(\frac{3n}{4}) + T(\frac{n}{4}) + cn\right)$$

$$\frac{1}{2}T(n) \leq cn + \frac{1}{2}\left(T(\frac{3n}{4}) + T(\frac{n}{4})\right)$$

$$T(n) \leq T(\frac{3n}{4}) + T(\frac{n}{4}) + 2cn \quad = O(n\log n)$$