Divide + Conquer

- Weekly check-ins (quiz) out Jan due Mon
- Hw 2 due Thurs
- "Hw 3" out Thurs - not to hand in
- Quiz 1 . Feb 3 (Thurs)

Vanilla Master Theorem

Let $T(n) = aT(\frac{n}{b}) + O(n^d \log n)$ $\quad (a \geq 1,\ b > 1)$

$$T(n) = \begin{cases} O(n^d \log n), & \text{if } a < b^d \\ O(n^d \log^2 n), & \text{if } a = b^d \\ O(n^{\log_b a}), & \text{if } a > b^d \end{cases}$$

Comparing n^d at top

$a^{\log_b n}$ bottom

$||$

$n^{\log_b a}$

Longer Master Theorem

$T(n) = aT(\frac{n}{b}) + f(n)$

Then

$$T(n) = \begin{cases} \cancel{O}\,\Theta(f(n)) & \text{if } f(n) = \Omega(n^{\log_b a + \epsilon}) \quad \epsilon > 0 \\ \cancel{O}\,\Theta(f(n) \log n) & \text{if } f(n) = \Theta(n^{\log_b a}) \\ \cancel{O}\,\Theta(n^{\log_b a}) & \text{if } f(n) = O(n^{\log_b a - \epsilon}) \quad \epsilon > 0 \end{cases}$$

where $f = \Theta(g) \Rightarrow f = O(g)$ and $g = O(f)$

$f = \Omega(g) \Rightarrow g = O(f)$

Last time:

Mergesort takes $O(n \log n)$ time worst case, best case

Quicksort takes $O(n \log n)$ best case, avg case (random pivot)

$O(n^2)$ worst case

a_1 _ _ _ a_n

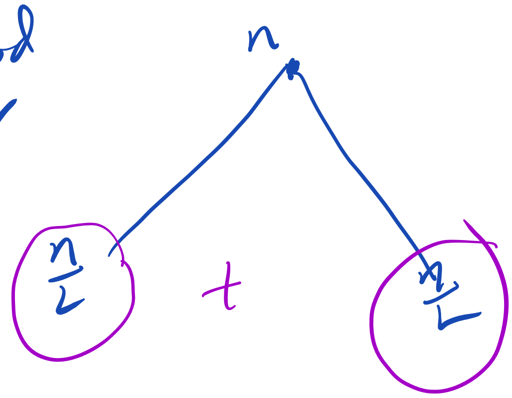
Goal: output in sorted order

choose a pivot p

$< p$ $\quad p \quad$ $> p$

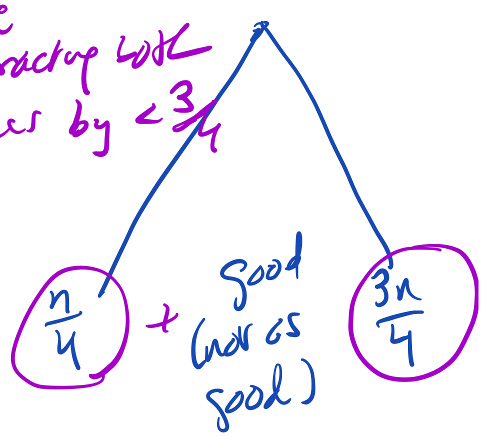
argued

if good pivot



best case

we're contracting both pieces by < 3/4



$$T(n) = a\, T\left(\frac{n}{b}\right) + n^d =$$

$$= \overbrace{T\left(\frac{n}{b}\right) + T\left(\frac{n}{b}\right) + \ldots + T\left(\frac{n}{b}\right)}^{a} + n^d$$

If $a\left(\frac{n}{b}\right) < n$ then we are contracting

$$T(n) = T(c_1 n) + T(c_2 n) + \ldots + T(c_k n) + n^d$$

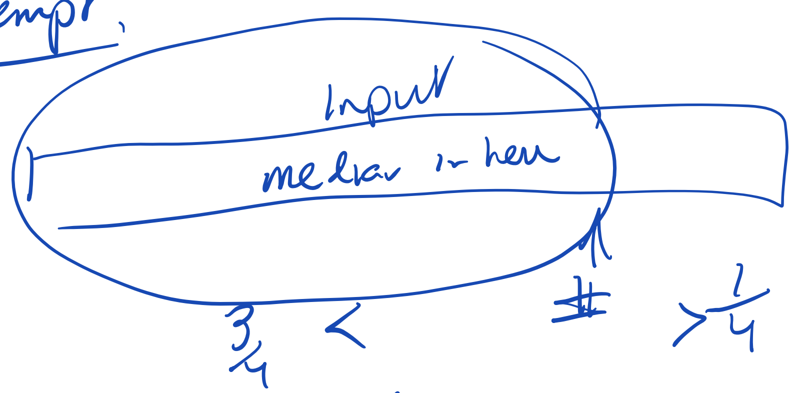
Because you get a pivot every few iterations
it behaves in expectation as $O(n \log n)$

Idea: What if we could find the median
in $O(n)$ time?

$$T(n) = 2T\left(\frac{n}{2}\right) + \underbrace{O(n)}_{\text{median finding and partitioning}}$$

$$= O(n \log n)$$

1st attempt:



Instead solve selection

selection$(a_1, \ldots, a_n, 1 \le k \le n)$
outputs the k^{th} smallest element.

selection $(a_1, \dots a_n, k)$

pick p in $\{a_1, \dots, a_n\}$

partition into $S_{<p}$ p $S_{>p}$

if k?

find k^{th}

$<p$

k?

if

k

$>p$

$k - |S_{<p}|$

If we have a good pivot
then we eliminate const fraction each step

bad Good bad

sorted sort

select

Good bad

If I can get good pivots often enough

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

$a=1 \quad b=\frac{4}{3} \quad d=1$

$a < b^d \Rightarrow O(n)$

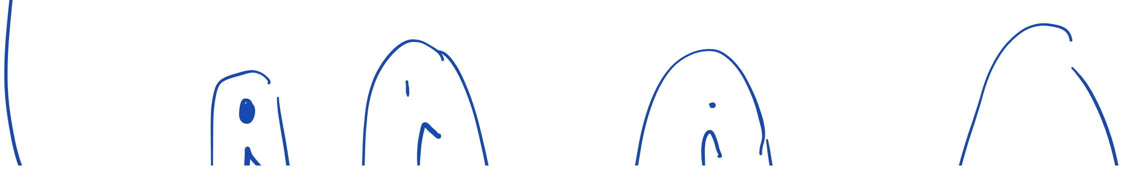
rand. alg
(in expected time)

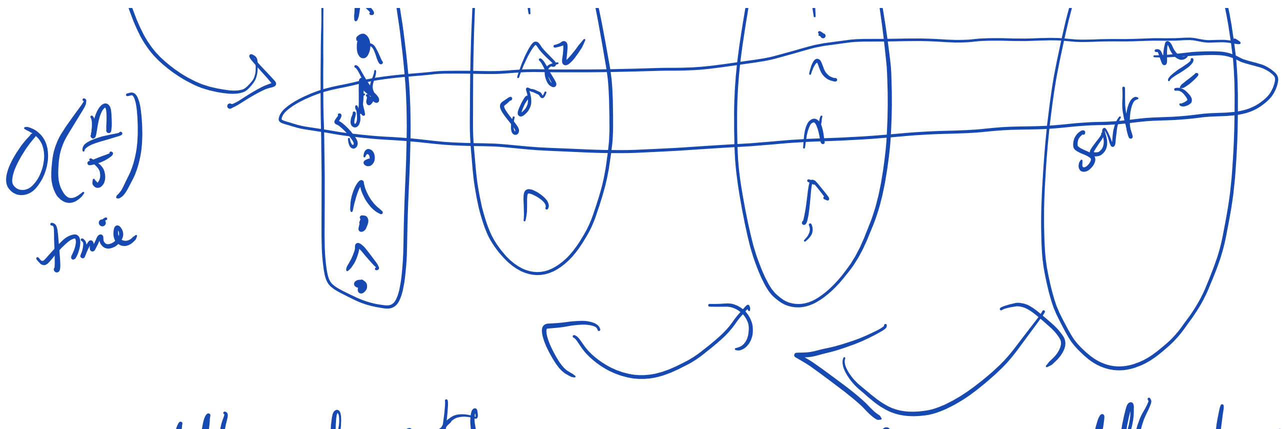
2nd (better) attempt

$T(n)$ = time find median of n items

$a_1, \ldots \ldots a_n$

$$\begin{array}{cccccc}
a_1 & a_2 & a_3 & \cdots & \cdots & a_{n/5} \\
a_{\frac{n}{5}+1} & a_{\frac{n}{5}+2} & - & & - & a_{\frac{2n}{5}} \\
\vdots & \vdots & \vdots & \vdots & & \vdots \\
\vdots & \vdots & \vdots & \vdots & & \vdots \\
\vdots & \vdots & \vdots & \vdots & & \vdots
\end{array}
\left.\vphantom{\begin{array}{c}a\\a\\a\\a\\a\end{array}}\right\}
\begin{array}{l}\text{5 rows}\\ \text{of size}\\ \frac{n}{5} \text{ each}\end{array}$$

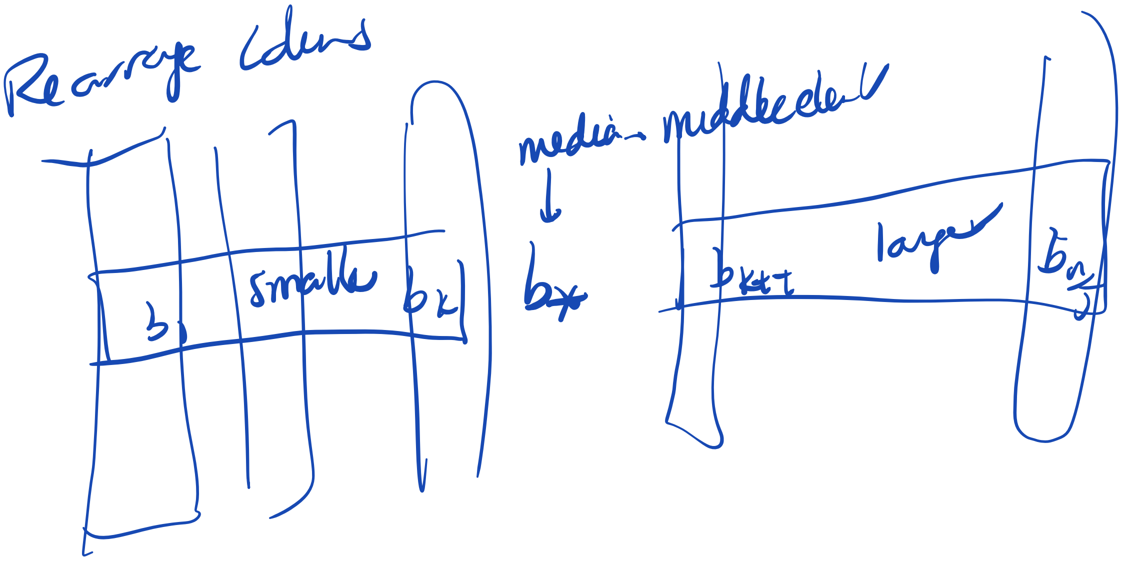


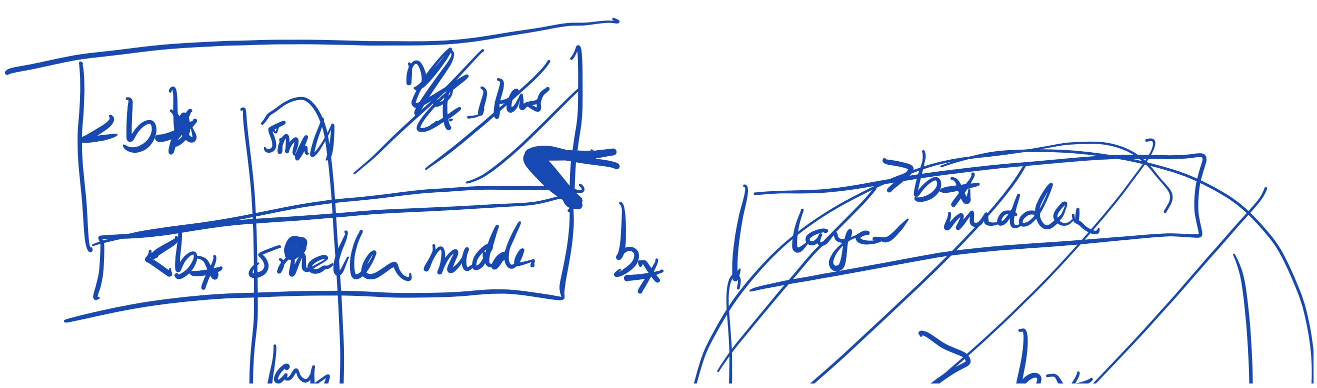


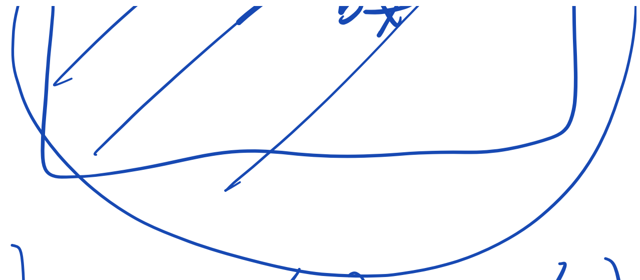
middle elements

find the median of the middle elements

Rearrange elems





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

recursion

finding median of middle element

key: Because

$$\frac{3}{4} + \frac{1}{5} < 1$$

$\Rightarrow O(n)$ alg.