Median:

Given an unsorted list $A = [a_1, \ldots, a_n]$ of $n$ numbers

Find the median.
(assume $n$ is odd, then for $n=2l+1$, we want to find the $(l+1)^{st}$ smallest number.)

Easy algorithm: Sort $A$ in $O(n \log n)$ time.

Our goal: $O(n)$ time algorithm.

Basic idea is like QuickSort:

1) Choose a good pivot $p$. (How?)
2) Partition $A$ into $A_{<p}, A=p, A_{>p}$.
3) Recurse on the appropriate one of the 3 sets.
   If looking for $k^{th}$ smallest in $A$, then:
   a) if $k \leq |A_{<p}|$, then find $k^{th}$ smallest in $A_{<p}$
   b) if $|A_{<p}| < k \leq |A_{<p}| + |A=p|$, then output $p$
   c) if $k > |A_{<p}| + |A=p|$, then find $(k - |A_{<p}| - |A=p|)^{th}$ smallest in $A_{>p}$.
Example: \( A=\{5, 2, 20, 17, 11, 13, 8, 9, 11\} \)

Say \( p=11 \)

\[ A_{<p}=\{5, 2, 8, 9\}, \quad A_{=p}=\{11, 11\}, \quad A_{>p}=\{20, 17, 13\} \]

Size 4 \hspace{1cm} Size 2 \hspace{1cm} Size 3

if \( k \leq 4 \), then find \( k \)-th smallest in \( A_{<p} \)

if \( 5 \leq k \leq 6 \) then \( 11 \) is the \( k \)-th smallest

if \( k > 6 \), then find the \( (k-6) \)-th smallest in \( A_{>p} \)

How to choose the pivot \( p \)?

Aiming for \( O(n) \) running time:

So we want a recurrence such as:

\[ T(n) = T\left(\frac{9}{10}n\right) + O(n) = O(n) \]

in fact for any constant \( c < 1 \),

\[ T(n) = T(cn) + O(n) = O(n) \]

More generally, for constants \( a, b > 0 \),

if \( a+b < 1 \) then

\[ T(n) = T(an) + T(bn) + O(n) = O(n) \]

We'll use it with \( a=\frac{1}{5} \) \& \( b=\frac{7}{10} \)
We'd like a pivot $p$ so that:

$$|A_p| \leq \frac{3}{4}n \quad \& \quad |A_{\bar{p}}| \leq \frac{3}{4}n$$

Then our running time will be:

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

Hence, say $P$ is a good pivot if:

$$|A_p| \leq \frac{3}{4}n \quad \& \quad |A_{\bar{p}}| \leq \frac{3}{4}n.$$ 

Think of sorted $A$:

So to find the median, we need a "near-median".

How can we find a good pivot in $O(n)$ time?
Randomized approach:

Choose a random element of A.
Check if it's a good pivot, if it is good, use it as p else try again.

# of elements of A that are good = \( \frac{1}{2} \)

\[ \Rightarrow \text{Probability that random element of A is a good pivot} \]
\[ = \frac{\frac{1}{2}}{n} = \frac{1}{2n} \]

How many times do you have to flip a coin to until you get a heads?
In expectation, 2 times.

Here, in expectation, we have to choose 2 random elements of A until we get a good pivot.

Then, expected running time is

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

But this is just the expectation and many times it may be much worse.
So we want worst-case running time.
Deterministic approach:

Want to find a subset $S$ of $A$.
Find the median($S$) recursively, and use $P = \text{median}(S)$.

Say $|S| = \frac{n}{5}$ (the choice of $\frac{1}{5}$ doesn't matter)

Worst-case: Save the $\frac{n}{5}$ smallest elements of $A$.

So median($S$) is the $\frac{N}{10}$th smallest of $A$.

Then, $|A < p| \leq \frac{9}{10}n \& |A > p| \leq \frac{9}{10}n$.

Since it takes $T(\frac{n}{5})$ time to find median($S$) recursively.

Our running time is then:

$$T(n) \leq T(\frac{n}{5}) + T(\frac{9}{10}n) + O(n)$$

but $\frac{1}{5} + \frac{9}{10} > 1$ so this doesn't solve to $O(n)$!
Need a more clever choice of $S$. For each $x \in S$, want that at least a few elements of $A$ are $\geq x$ & a few are $\leq x$.

So break $A$ into groups of size $5$ (as a power of 5)

Groups $G_1, G_2, \ldots, G_{\frac{n}{5}}$ each has 5 elements.

Since a group $G_i$ has only 5 elements, we can sort it in $O(1)$ time, and then its median is the middle element.

Let $m_i$ be the median of $G_i$.

Let $S = \{ m_1, m_2, \ldots, m_{\frac{n}{5}} \}$

We'll use these $\frac{n}{5}$ medians as our set $S$.

Let $p = \text{Median}(S)$. 
New algorithm:

Select \((A, k)\):

input: unsorted \(A = [a_1, \ldots, a_n]\) (where \(n\) is a power of 5) & integer \(k\) where \(1 \leq k \leq n\)

output: \(k\)th smallest of \(A\).

1) Break \(A\) into \(\frac{n}{5}\) groups of 5 elements each. Call these groups \(G_1, G_2, \ldots, G_{\frac{n}{5}}\)

2) For \(i = 1 \rightarrow \frac{n}{5}\), sort \(G_i\)

3) Let \(m_i = \text{median}(G_i)\)
   
   Let \(S = \{m_1, m_2, \ldots, m_{\frac{n}{5}}\}\)

4) \(p = \text{Select}(S, \frac{n}{10})\) (so \(p\) is the median of \(S\))


6) If \(k \leq |A < p|\), then return \((\text{Select}(A < p, k))\)
   
   If \(|A_p| < k \leq |A < p| + 1 |A = p|\)
   
   Then return \((p)\)

   If \(k > |A < p| + 1 |A = p|\)
   
   Then return \((\text{Select}(A > p, k - |A < p| - |A = p|))\)
Claim: \( p \) is a good pivot.

In particular,
\[
\geq \frac{3n}{10} \text{ elements of } A \text{ are } \leq p \Rightarrow |A_{\geq p}| \leq \frac{7n}{10}
\]
\[
\geq \frac{3n}{10} \text{ of } A \text{ are } \geq p \Rightarrow |A_{<p}| \leq \frac{7n}{10}
\]

From the claim, the running time is
\[
T(n) \leq T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right) + O(n)
\]

\[
\uparrow \quad \uparrow \quad \uparrow
\]

to find \(\text{median}(S)\) \quad to recurse \quad to break \( A \)

\[
\text{step (4)} \quad \text{step (6)} \quad \text{into groups}
\]

\[
\text{because of the} \quad \text{find medians} \quad \text{find medians}
\]

\[
\text{claim} \quad \text{of every group} \quad \text{of every group}
\]

\[
\text{Partition } A \quad \text{into } A_{<p}, A_{>p}
\]

\[
\text{steps 1, 2, 5.}
\]

Since \[
\frac{1}{5} + \frac{7}{10} = \frac{9}{10} < 1
\]

This solves to: \( T(n) = O(n) \).
Proof of the claim:

Sort the groups by their medians, so that:

\[ m_1 \leq m_2 \leq \ldots \leq m_{\frac{n}{5}} \]

Then \[ p = \frac{m_n}{10} \]

Here's the picture:

Which elements of S are guaranteed to be \( \leq p \)?

\( \geq \frac{3n}{10} \) for each of these, 3 elements in its group are \( \leq p \).

Hence, \( \geq \left( \frac{3n}{10} \right) \) are \( \leq p \).

Similarly, \( \geq \frac{3n}{10} \) are \( \geq p \).