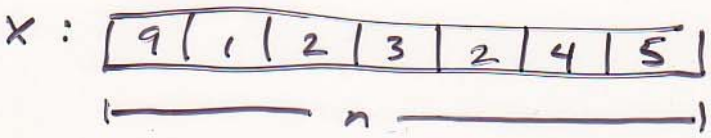
# CS 223 Lec 4

## Statistics on Arrays




e.g. find mean ? ✓

min ?

max ?

median ?

mode ?

Finding Minimum

$\Rightarrow$ $n-1$ comparisons

Finding Maximum

$\Rightarrow$ $n-1$ comparisons

# Find Both the Min + Max

| $x_1$ | $x_2$ | $x_3$ | ... | | $x_{10}$ |
|---|---|---|---|---|---|
| 6 | 1 | 5 | 3 | 9 .. | 2 |

$x_1 \lessgtr x_5$ ?

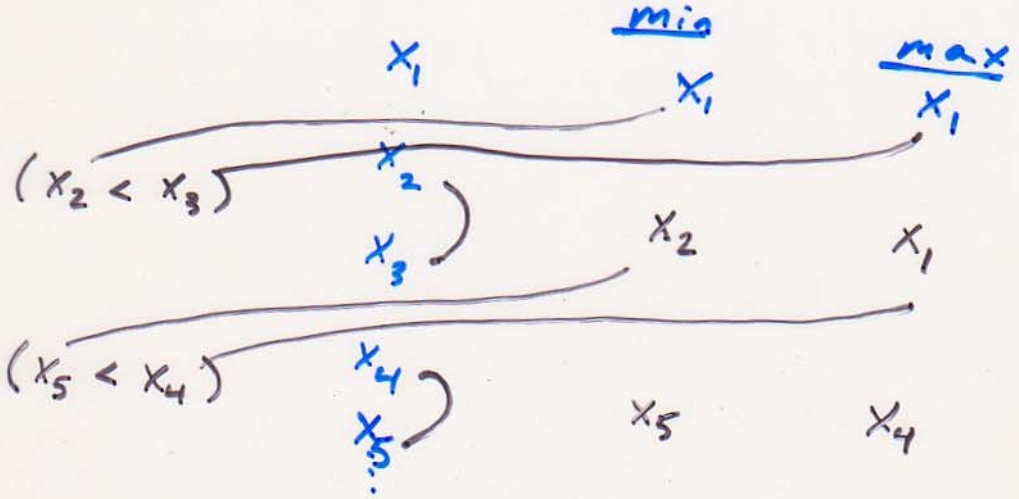
⋮

$x_2$ is largest

$x_5$ is smallest

17, 15, 13, 14, 13

# Algorithm for finding Min / Max ...




$x_n$

($n$ odd vs. even)

$x_1$, $x_2$ ) min $x_2$ max $x_1$

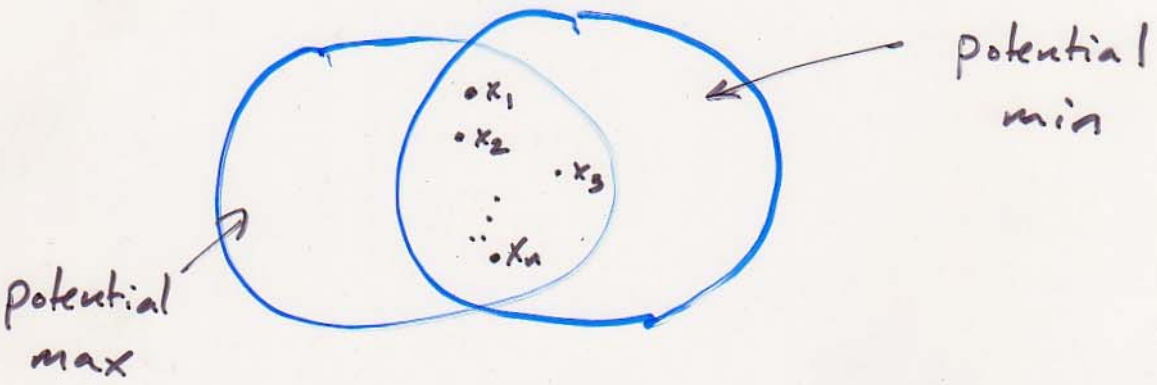
$x_3$, $x_4$ )

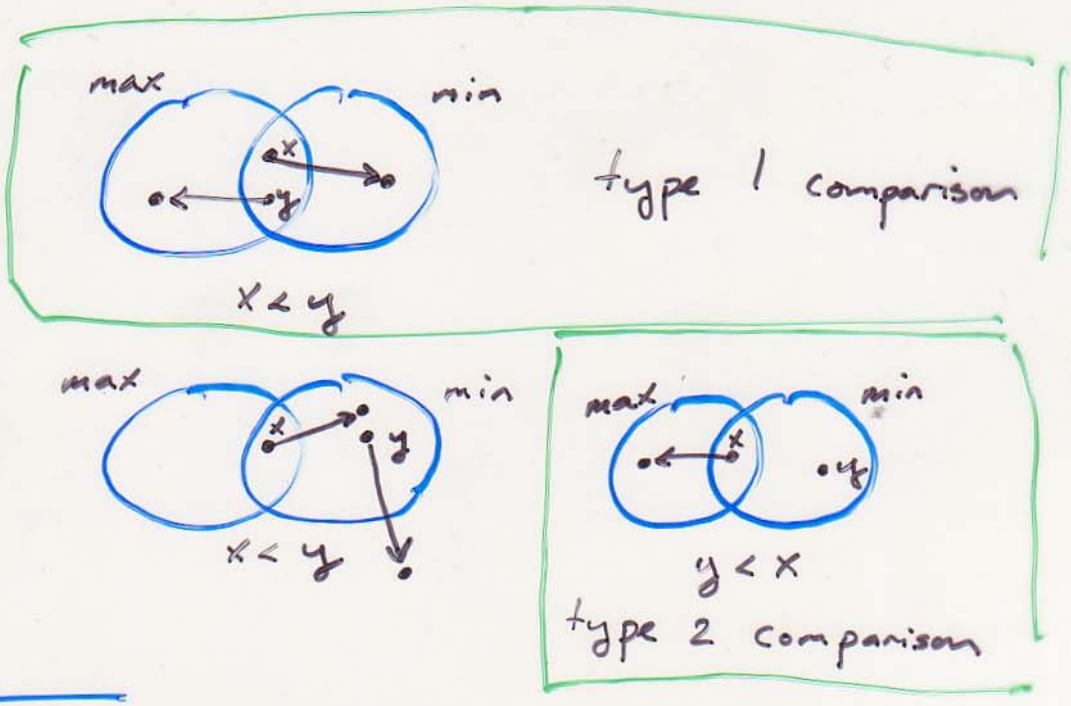
Claim $\leq \left\lfloor \frac{3n}{2} \right\rfloor$ comparisons.

# Finding a Lower Bound on the # of comparisons needed identify max/min in an unsorted array

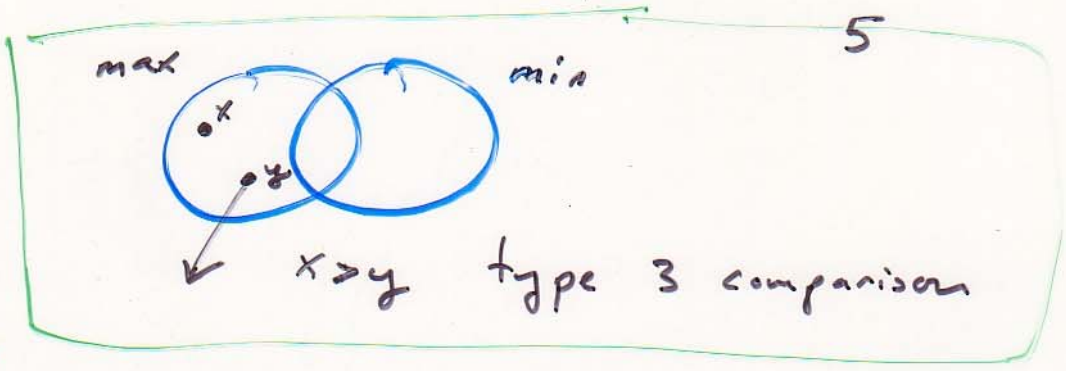
---

at start:

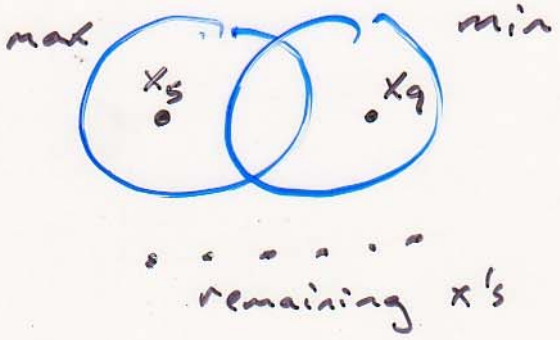


can force only comparisons




at end:



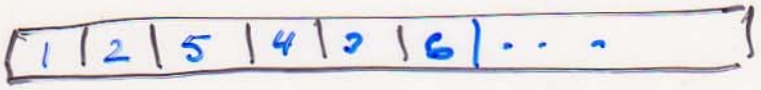

# of comparisons

$$\geq \lceil n/2 \rceil + n - 2$$

$$= \left\lceil \frac{3n}{2} \right\rceil - 2$$

# The Selection Problem

given: an unsorted array

A 


## order statistics

Defn The ith order statistic for A is the ith smallest value in A (x)

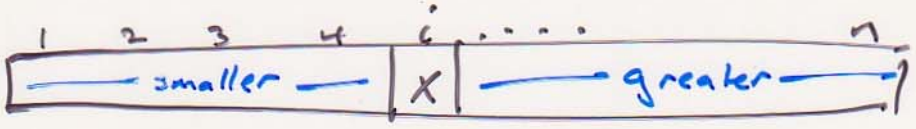
in other words there are

$i-1$ values $\leq x$

$n-i$ values $\geq x$

$n-1$

Sort A ...



# SELECTION

Given an unsorted array $A$, find the $i$th order statistic.

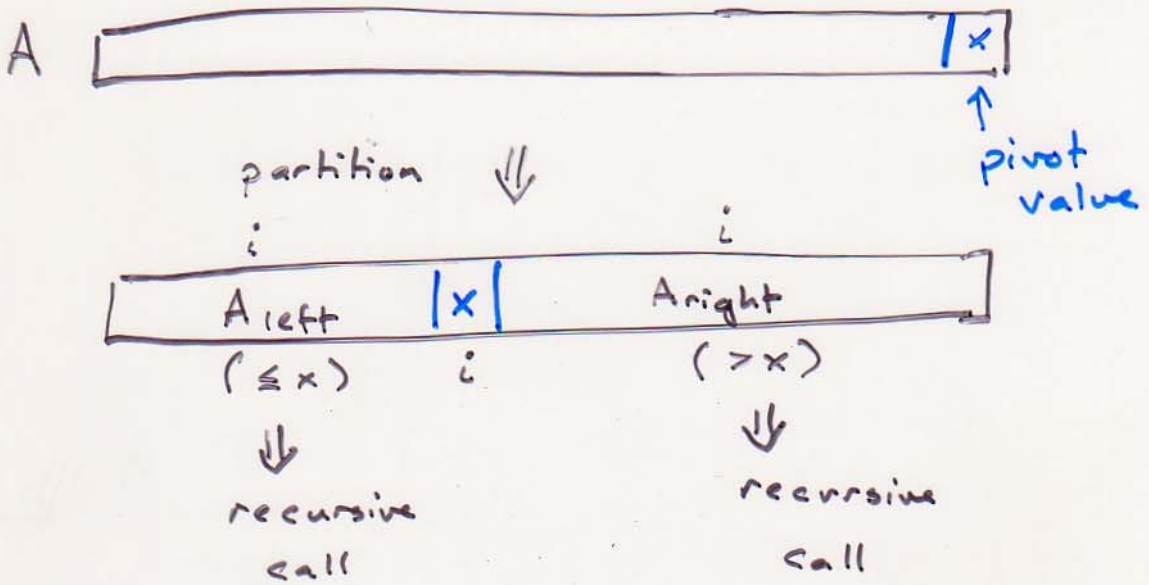
a possible solution

- sort the array

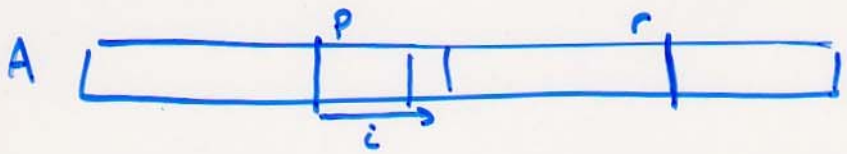
$\Rightarrow O(n \lg n)$ time

it turns out, there is a linear time ($O(n)$) algorithm to solve this.

First approach: use quicksort algorithm ideas...




SELECT ( A , p , r , i )




p = start index

r = end index

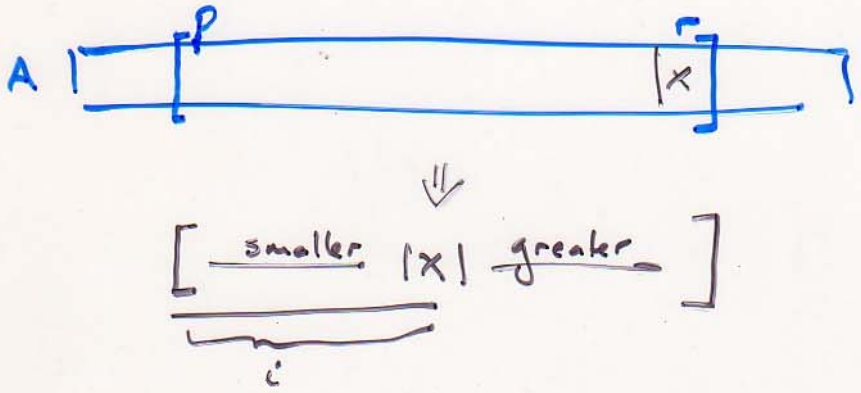
i = order statistic to find inside A [p... r]

PARTITION ( A, p, r )

A = array

p = start index

r = end index




returns i