K-th Order Statistics

- Input: Set of integers S
- Output: *k-th* smallest integer in S

What if k=1?
What if k=n?
What if k=n/2?

FIND-MIN

- How many comparisons does it take to find the minimum in a set of integers?
- Answer: n-1

Upper Bound for FIND-MIN

Upper Bound Theorem: Finding the minimum in a set of n integers requires no more than n-1 comparisons.

Proof: Give algorithm

Lower Bound for FIND-MIN

- Lower Bound Theorem: Finding the minimum in a set of integers requires at least n-1 comparisons.

Note: We need to show that every algorithm makes at least n-1 comparisons for some input of size n.

Note: There are only n possible outcomes so the decision tree argument we used for sorting doesn't work.

A Different Decision Tree Argument

- Consider the decision tree model of algorithm **A** for finding the minimum and maximum of a set of size n.
- The information **A** has gathered at any node can be represented by a partial order.
- An answer to a query can reduce the number of weakly connected components in the partial order by at most 1.
- The partial order associated with a leaf must be weakly connected.
- Thus algorithm **A** must make at least n-1 queries before reaching a leaf.

Find min & max

- Upper Bound Theorem: Finding the minimum and maximum in a set of n integers requires no more than ceil(3n/2)-2 comparisons.
- Proof: Give an algorithm

Upper Bound for Finding the min and max

- Algorithm for even n:
  - Make n/2 pairwise comparisons
  - Find the maximum of the winners with n/2-1 comparisons
  - Find the minimum of the losers with n/2-1 comparisons
- Algorithm for odd n:
  - Run even algorithm on first n-1 integers
  - Compare the min and max to the last integer

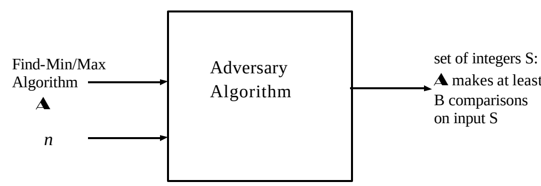
FIND MIN/MAX(S)

$S_{max} = \phi$, $S_{min} = \phi$
For i=1 to floor(n/2)
    If $x_{2i-1} < x_{2i}$ then
        $S_{max} = S_{max} + \{x_{2i}\}$ and $S_{min} = S_{min} + \{x_{2i-1}\}$
    Else
        $S_{max} = S_{max} + \{x_{2i-1}\}$ and $S_{min} = S_{min} + \{x_{2i}\}$
MAX = FIND_MAX(S_{max}) and MIN = FIND_MIN(S_{min})
If n is odd then
    MAX =FIND_MAX(MAX,x_n)
    MIN =FIND_MIN(MIN,x_n)
Return MIN and MAX

Finding the min and max

- Lower Bound Theorem: Finding the minimum and maximum in a set of n integers requires at least ceil(3n/2) - 2 comparisons.
- Proof: Adversary argument

Adversary Argument
(for finding min/max)



Example of an adversary

You pick a number y between 1 and 100
I have to guess by posing queries of the form
    “Is it x?”
To which you answer yes or y<x or y>x.

- How many queries do I need to make?
- Prove it!