Maximum and Minimum

The **selection problem** is the problem of computing the i-th order statistic (i.e. the i-th smallest number) of a set A.

   Special Cases:

- Min $(i = 1)$

- Max $(i = n)$

- Median $(i = n/2)$

Order Statistics

How many comparisons are necessary and sufficient for computing both the min and max?

Necessary: you cannot find it with fewer comparisons

Sufficient: no more comparisons are needed

- Compute the max: $n - 1$ comparisons

- Compute the min: $n - 1$ comparisons

- Thus, computing both min & max requires $2n - 2$ comparisons

   Is there a way to improve this?

Idea

Maintain the variables min and max. Process the n numbers in pairs.
   Operation:

- Form pairs of elements

- Compare elements in each pair

Initial Values

n **is odd** set both min and max to the first elements

n **is even** compare first two elements, assign smallest to min and largest to max

   Total # of comparisons:

n is od $3(n - 1)/2$

n is even $1 + 3(n - 2)/2 = 3n/2 - 2$

Odd Example

n = 5(odd), array $A = 2, 7, 1, 3, 14$

1. Set min = max = 2

2. Compare elements in pairs

   - $1 < 7 \implies$ compare 1 with min and 7 with max
   - $3 < 5 \implies$ compare 3 with min and 4 with max

   We performed $3(n-1)/2 = 6$ comparisons

Even Example

n = 6(even), array $A = 2, 5, 3, 7, 1, 4$

1. Compare 2 with 5: 2 ¡ 5

2. Set min = 2, max = 5

3. Compare elements in pairs:

   - $3 < 7 \implies$ compare 3 with min and 7 with max
   - $1 < 4 \implies$ compare 1 with min and 4 with max

   We performed $1 + 3(n-2)/2 = 7$ comparisons

   The total # of comparisons is at most

$$3\lfloor n/2 \rfloor$$

What about the median?

- By repeatedly applying the algorihtm for finding min value, it will take $O(n)$ time to find the i-th smallest element.

- Therefore, when finding the median, it will take $O(n^2)$ time. This is more than what sorting takes.

- Can we do better?

Reduction to sorting

1. $O(n \log n)$ algorithm:

2. Step 1: apply merge sort

3. Return i-th element of the array

Definitely better than finding the median using the min algorithm. **Key Idea**: We can use reduction to sorting to help us with the selection problem