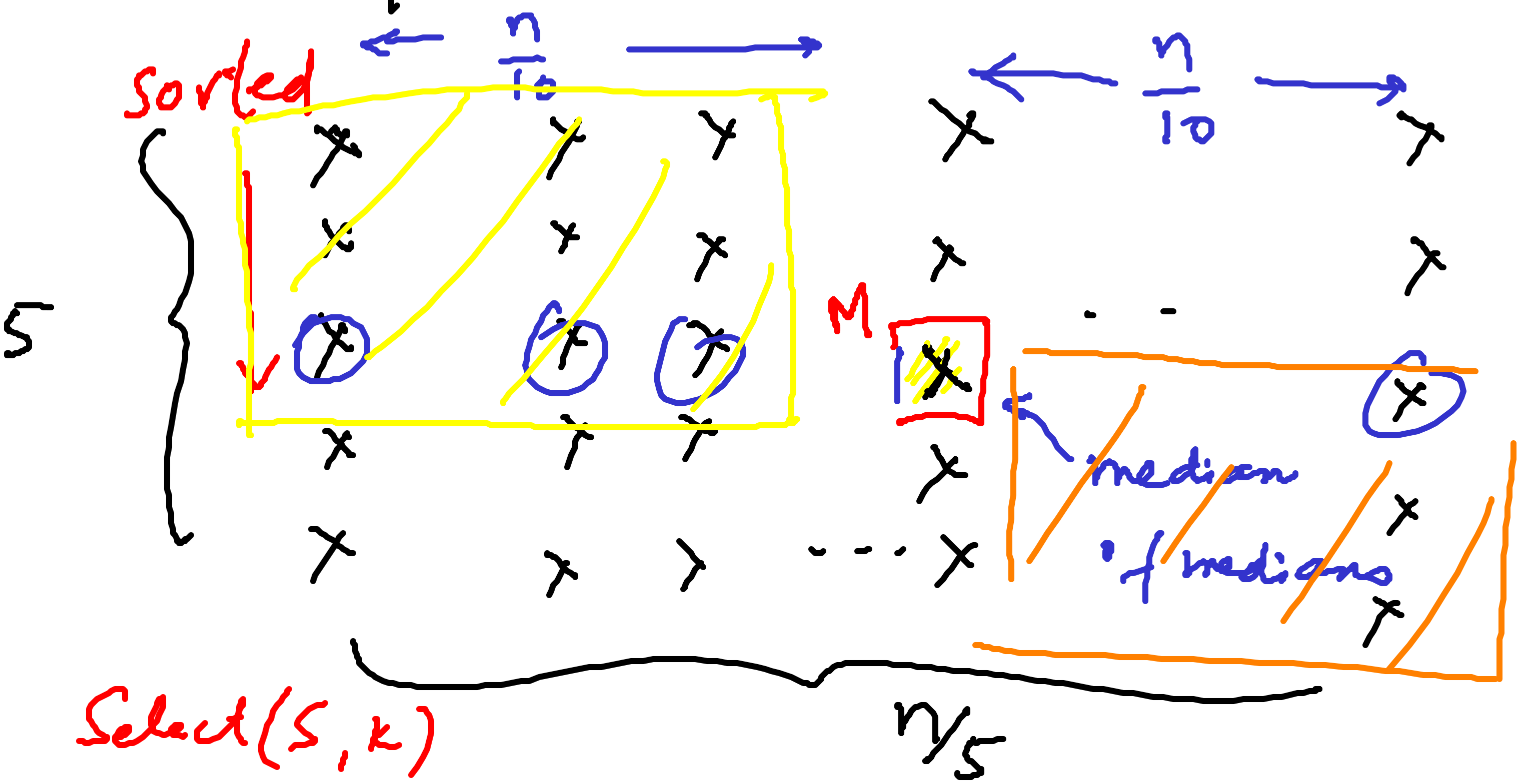
CSL 356 Aug 6

Deterministic Selection

How to find a "good" splitter, so that at least a constant fraction, say $\frac{n}{4}$ elements are eliminated



Select(S, k)

1. Sort every column containing at most 5 elements

   Time : $O(1) \cdot \frac{n}{5} = O(n)$

2. Find the "median" of the median of each row, i.e. median of $\frac{n}{5}$ elements
3. Find the rank of M $O(n)$

Claim : The median of medians has rank in the range $[\alpha n, (1-\alpha)n]$ for some constant $\alpha < 1$

$\frac{n}{10} \times 3$ elements are smaller than M

$\frac{3n}{10}$ " " larger than M

$\alpha = \frac{3}{10}$

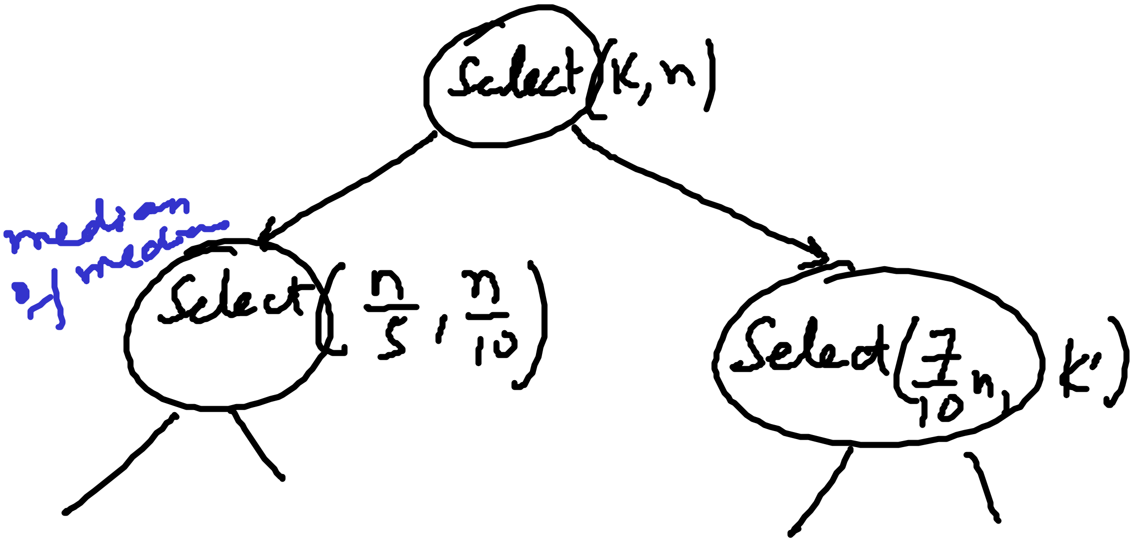
Step 4 : If $\text{rank}(M,S) < K$ then we can discard all element in the shaded box to the right bottom

Time : $O(n)$

Step 5 : Call recursively on the remaining elements with the adjusted value of K.

$T(n)$: the worst case time over all $K \le n$

$$T(n) = T\left(\frac{n}{5}\right) + O(n) + T\left(\frac{7}{10}n\right) \qquad T(2) = O(1)$$



Prove (using induction) that

$T(n) \leq \alpha \cdot n$ for some constant α.

<u>Soft-heaps</u> by Chazelle

Prune and Search

Finite alphabet Σ

Strings $s_1, s_2, \ldots s_n$ defined over Σ of lengths $l_1, l_2 \ldots$

$|s_i| = l_i$

01, 1100, 101010, ...

. ... -

$L = \sum l_i$

possible that some string has size 1 whereas some string may have size $\frac{L}{4}$

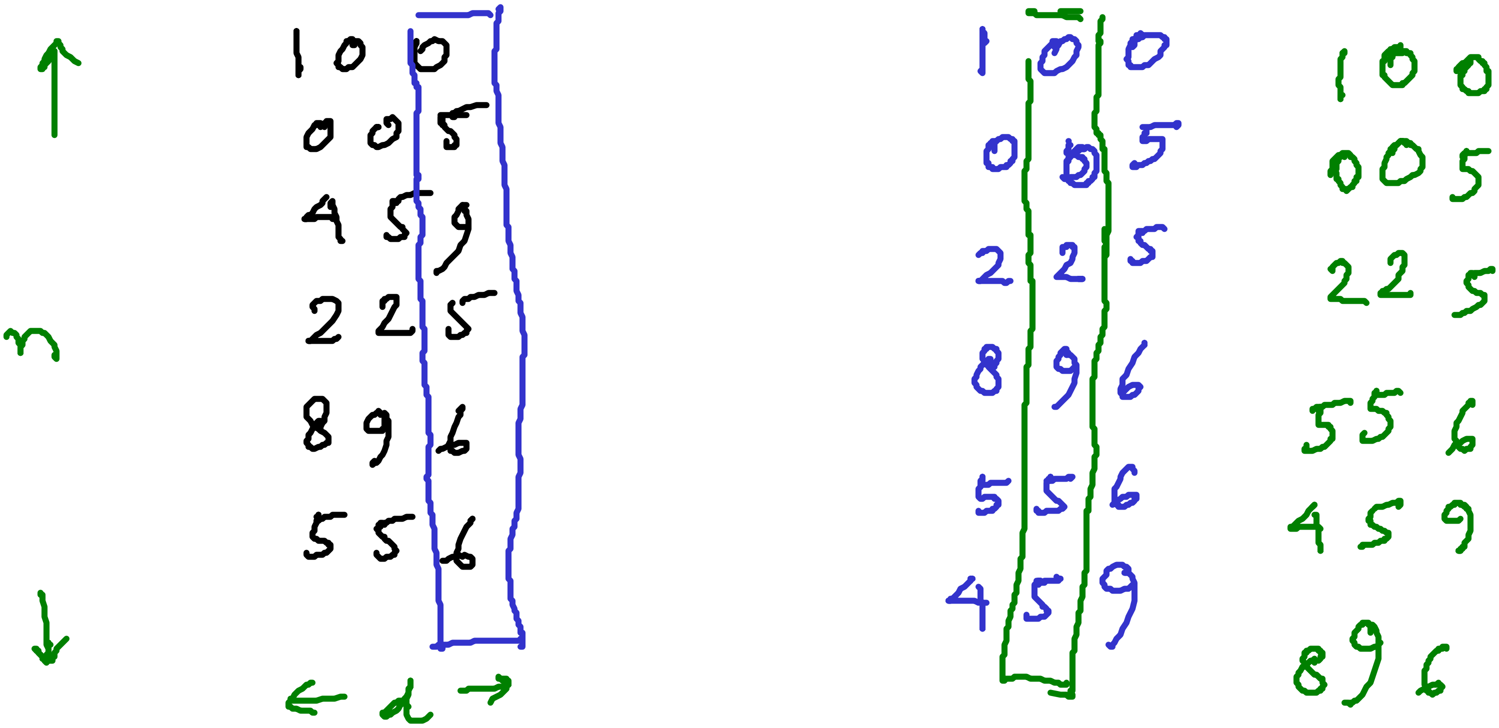
We would like to 'order' the strings

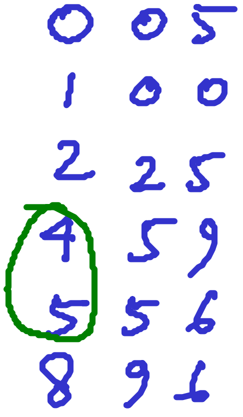
Use "lexicographic ordering"

What algorithm should we use?

Radix sort?

100, 255, 099, 005, ... 855





Using count sort, time for each phase

$\Theta\left(|\Sigma| + n\right)$

range

Stable sort : doesn't alter the ordering of numbers with identical

Radix sort has to be based on some stable sorting

Count Sort : Stable ?

Every comparison sorting can be made stable

Radix Sort

$$O\left(d\left(\overset{n}{|\Sigma|}+n\right)\right)$$

As long as d is constant

$\Sigma = 1..n$, i.e. $\log n$ bits

We can sort n numbers in the range $[1..n^d]$ in $O(n)$ time.