**CS430 Lecture 8 Activities**

Opening Questions

1. Order Statistics: Select the ith smallest of n elements (the element with rank i)

i = 1: minimum;

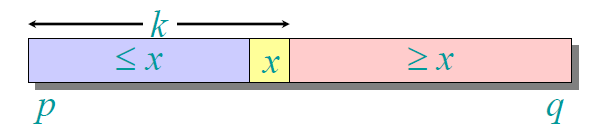
i = n: maximum;

i = floor((n+1)/2) or ceiling((n+1)/2): median

How fast can we solve the problem for various i values?

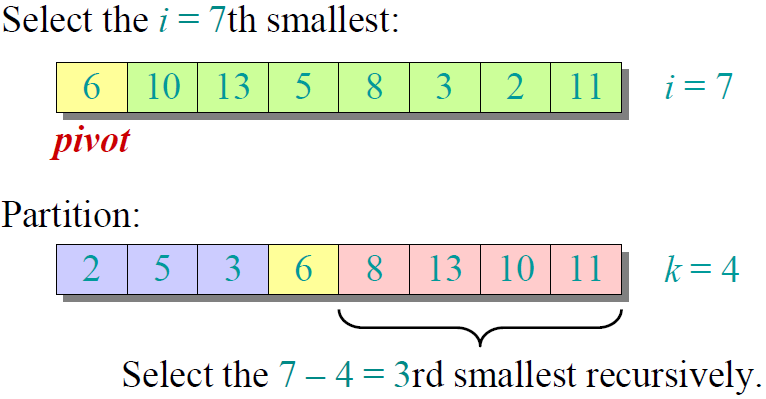
Randomized Algorithm for finding the ith Element

1. Think about partition (with a random choice of the pivot) from quicksort. Can you think of a way to use that and comparing the final location of the pivot to i, and then divide and conquer?

****

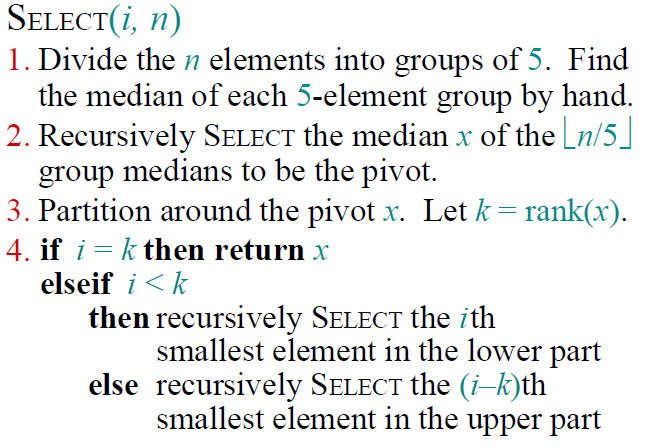
2. Demonstrate on this array to find i=th smallest element

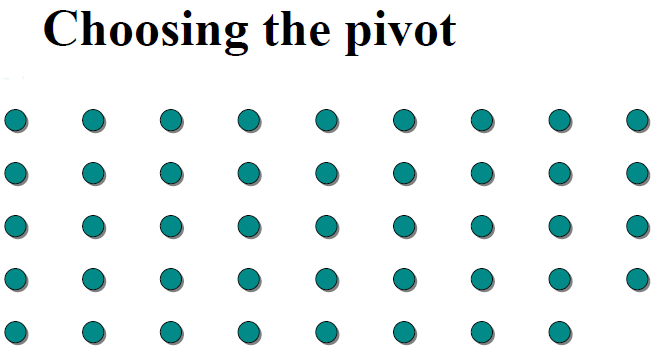
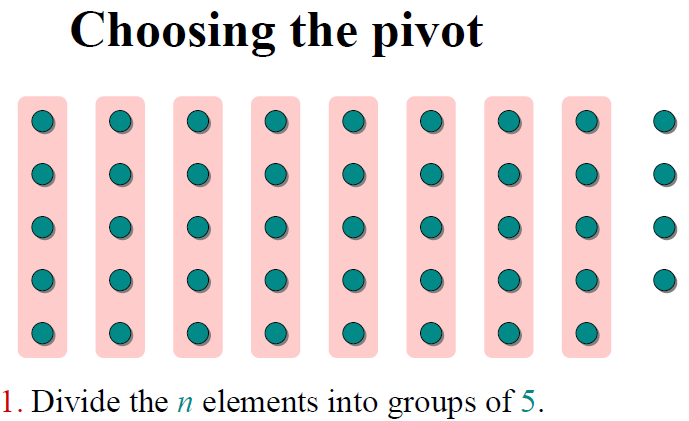
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 6 | 10 | 13 | 5 | 8 | 3 | 2 | 11 |

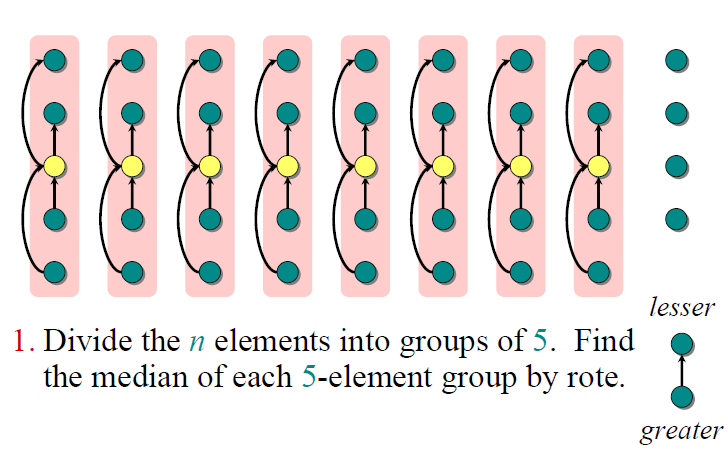
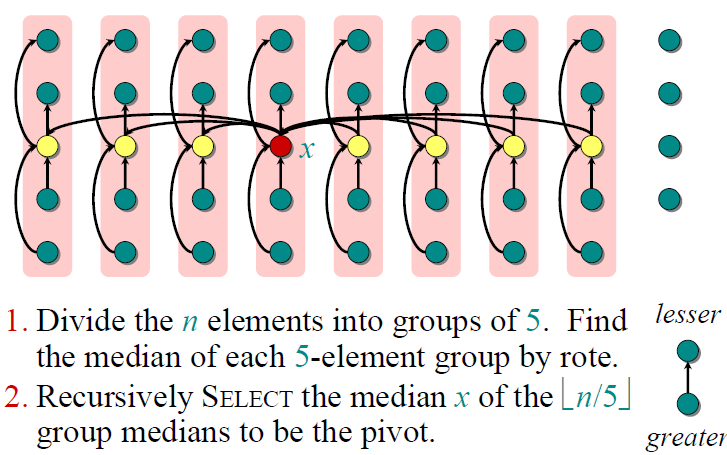


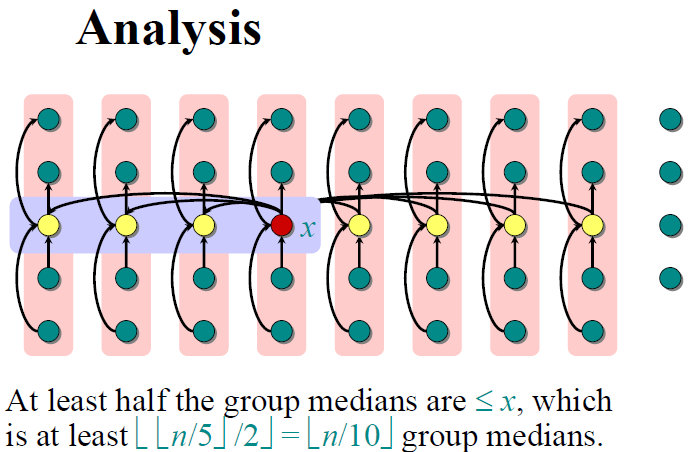
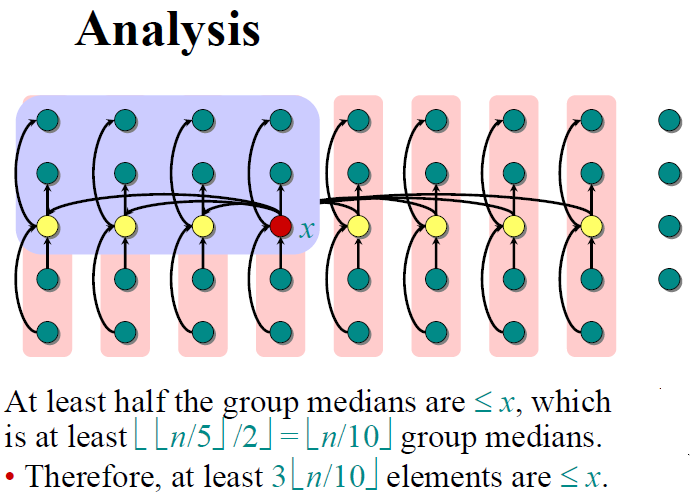
3. What is the worst case running time if your find the ith smallest element?

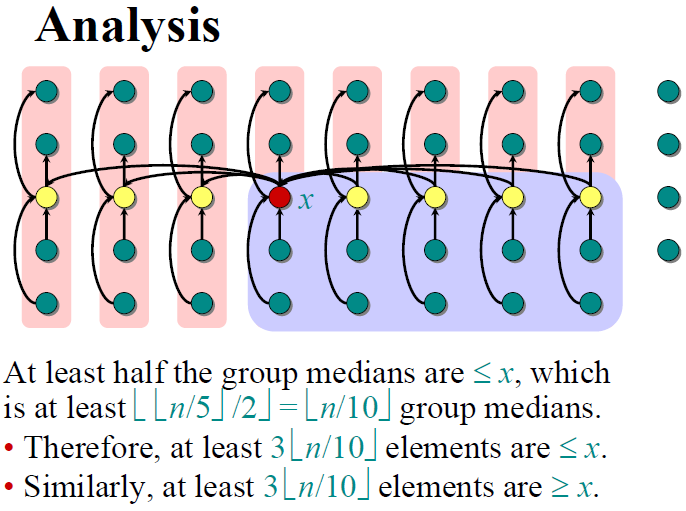
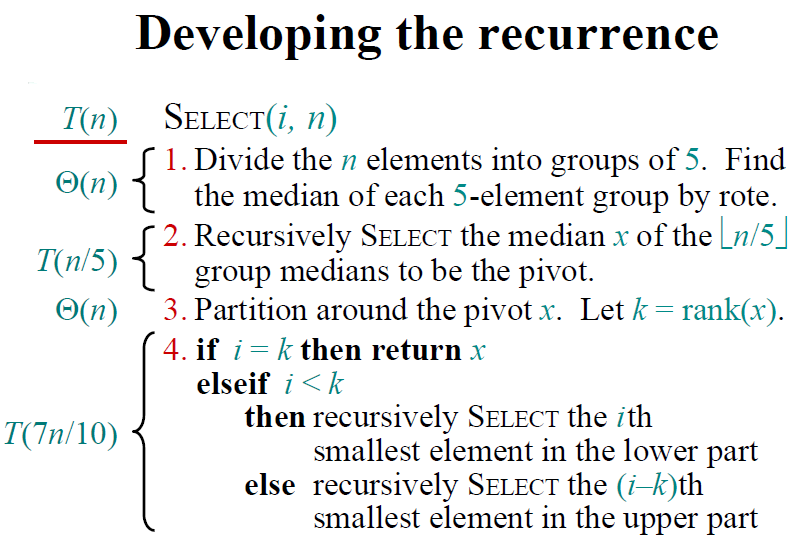
Is there an algorithm to find the ith smallest element that runs in linear time in the worst case?

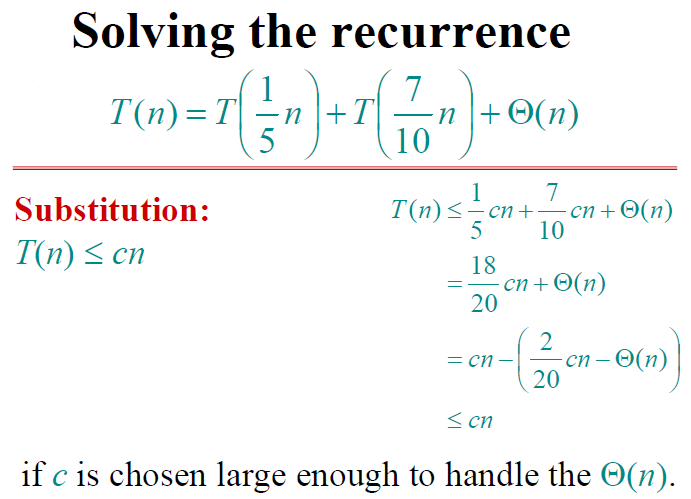




In practice, this algorithm runs slowly, because the constant in front of n is large.

Would we use this approach to find the median to partition around in Quicksort, and achieve in worst-case Theta (n log n) time?