CSci 231 Homework 4

Selection and Divide-and-conquer

CLRS Chapter 9

1. (CLRS 9.3-5) Suppose that you have a "black-box" worst-case linear-time median subroutine. Give a simple, linear-time algorithm that solves the selection problem SELECT(i) for an arbitrary i.

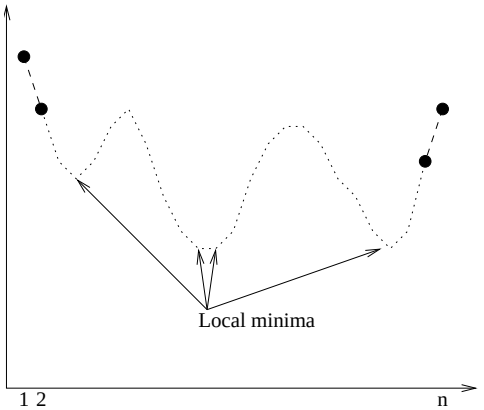
2. (CLRS 9.3-7) Describe an $O(n)$ algorithm that, given a set S of n distinct numbers and a positive integer $k \leq n$, determines the k numbers in S that are closest to the median of S.

3. Let A be a list of n (not necessarily distinct) integers. Describe an $O(n)$-algorithm to test whether any item occurs more than $\lceil n/2 \rceil$ times in A. Your algorithm should use $O(1)$ additional space.

4. (CLRS 9.3-6) Give an $O(n \lg k)$ algorithm to find the $k-1$ elements in a set that partition the set into (approx.) k equal-sized sets $A_1, A_2, \ldots A_k$ such that all elements in A_i are smaller than all elements in A_{i+1}. Assume k is a power of 2.

5. CLRS 2-4 (The inversion problem).

6. Consider an array A of length n for which we know that $A[1] \geq A[2]$ and $A[n-1] \leq A[n]$. We say that $A[x]$ is a *local minimum* if $A[x-1] \geq A[x]$ and $A[x] \leq A[x+1]$. Note that A must have at least one local minimum.



We can obviously find a local minimum in $O(n)$ time by scanning through A. Describe an $O(\log n)$ algorithm for finding a local minimum.