

Homework Assignment 3

This assignment is due by Tuesday February 24 (in class). Assignments should be handed in *before the class begins*.

Problem 1: Solve exercises 6.5-7 (page 142), 8.2-4 (page 170) , and 24.3-2, 24.3-8 (page 600) in the textbook.

Problem 2: Prove the correctness of radix-sort by proposing a loop invariant and proving that it holds by induction. Make sure to write the proof and details carefully and clearly.

Problem 3: [extra credit; 20 points] In each of the following questions you are asked to design an $O(n)$ algorithm that takes as input: an array with n distinct numbers and an integer $k \leq n$ (the array is not in sorted order). Note that k is not a constant and it may depend on n (e.g. we may have $k = \sqrt{n}$) so an algorithm with $O(kn)$ run time does not satisfy the requirement. In each part you need to argue both the correctness and run time of the algorithm.

(i) Give an $O(n)$ algorithm that finds the k smallest numbers in the array.

(ii) Give an $O(n)$ algorithm that finds the elements of rank $\lceil \frac{n}{2} \rceil - k + 1, \lceil \frac{n}{2} \rceil - k + 2, \dots, \lceil \frac{n}{2} \rceil, \dots, \lceil \frac{n}{2} \rceil + k$ in the array (so we are looking for a “band” of $2k$ elements around the median).

(iii) Give an $O(n)$ algorithm that finds the k elements that are closest in value to the median. For example, if $k = 5$, $n = 11$ and the array is 1, 2, 3, 4, 5, 6, 8, 14, 15, 16, 17, then the median is 6 and the 5 closest numbers are 2, 3, 4, 5, 8. Note that the array in the example is shown in sorted order but the input array is not necessarily sorted.

Hints: You may assume a linear time algorithm for selection (see Chapter 9) but you do not need to know its details. Parts (i), (ii) should be helpful for part (iii).