CSCI 3104-Spring 2016: Assignment #3.

Assigned date: Friday, 2/5/2016,
Due date: Tuesday, 2/16/2016, before class
Maximum Points: 50 points (includes 5 points for legibility).

Note: This assignment *must be turned in on paper, before class.* Please do not email: it is very hard for us to keep track of email submissions. Further instructions are on the moodle page.

P1 (10 points) You are given 2 large, **sorted** lists A_1, A_2 each containing n elements each. You are allowed to read any element $A_1[j]$ or $A_2[j]$ for $0 \leq j < n$, but cannot otherwise modify either array.

Write an algorithm to find the r^{th} smallest element of the union of the lists A_1, A_2. What is the running time of your algorithm?

P2 (10 points) Consider the following strategy for choosing a pivot element for quicksort of an array A.

1. Let n be the number of elements of the array A.
2. If $n \leq 15$, perform an insertion sort of A and return.
3. If $n > 15$:
   (a) choose $2\lfloor\sqrt{n}\rfloor$ elements at random from n, let S be the new list with the chosen elements.
   (b) Sort the list S using insertion sort and use the median m of S as a pivot element.
   (c) Partition using m as a pivot.
   (d) Carry out quicksort recursively on the two parts.

(A) If the element m obtained as the median of S is used as a pivot, what can we say about the sizes of the two partitions of the array A?
(B) How much time does it take to sort S and find its median. Give a $\Theta(\cdot)$ bound.
(C) Write a recurrence for the worst case running time of quicksort with our pivoting strategy?
(D, Extra Credit) Solve the recurrence in part (B) to obtain a $\Theta(\cdot)$ bound on the worst case running time.

P3, 10 points You are given n metal balls $B_1, \ldots, B_n$, each having a different weight. You can compare the weights of any two balls by weighing them on a balance and finding which one is heavier.
(A) Consider the algorithm to find the heaviest ball as follows:

1. Divide the n balls into $\frac{n}{2}$ pairs of balls.
2. Compare each ball with its pair, and retain the heavier of the two.
3. Repeat the process, until just one ball remains.

Illustrate the weighings that the algorithm will do for data given $n = 8$ and the weights are

$$B_1 : 3, B_2 : 5, B_3 : 1, B_4 : 2, B_5 : 4, B_6 : 0.5, B_7 : 2.5, B_8 : 4.5 \,.$$

(B) Show that for n balls, the algorithm in **(A)** uses at most n weighings.
(C) Describe an algorithm that uses the results of **(A)** to find the second heaviest ball using at most $\log_2(n)$ additional weighings.
(D) Illustrate the additional weighings that your algorithm in **(C)** will perform for the data from part **(A)**.

P4, 20 points We are given a list A with n elements $(A[0], \ldots, A[n-1])$, all of which are *distinct*, and the associated list of weights w $(w[0], \ldots, w[n-1])$ wherein each weight $0 < w[i] < 1$ and all the weights sum up to 1: $\sum_{j=0}^{n-1} w[j] = 1$.

An element $A[j]$ is the weighted median if the following inequalities hold:

$$\sum_{k \text{ s.t } A[k]<A[j]} w[k] \leq 0.5 \text{ and } \sum_{k \text{ s.t. } A[k]>A[j]} w[k] \leq 0.5$$

In other words, the sum of weights of all elements smaller than the weighted median is at most 0.5 and the sum of weights of all elements larger than the weighted median is at most 0.5.
Example: Take the array A with the weights given:

$A[j]$	1	4	5	2	3	6
$w[j]$	0.1	0.1	0.1	0.1	0.2	0.4

The weighted median is $A[1] = 4$. The elements smaller than 4 are $1, 2, 3$ whose weights sum up to 0.4. The elements larger are $5, 6$ whose weights sum up to 0.5

Modify the selection algorithm to find the weighted median of a given list A and list of weights w. What are the worst case running times of your algorithm in terms of n.