Algorithms: Assignment 7

Due date: October 9 (Wednesday)

Problem 1 (5 points)
Consider the problem of finding the kth smallest element of an array $A[1..n]$, that is, the element that would occupy the kth position after sorting the array. For example, if the array is $<6, 4, 8, 2, 10, 0>$ and $k = 3$, then the kth smallest element is 4, since it is the third element in the sorted array $<0, 2, 4, 6, 8, 10>$. Write an algorithm for finding the kth smallest element of a given array. Its average-case complexity should be *better* than the complexity of sorting. Thus, sorting the array and then returning the kth element is not an appropriate solution.

Problem 2 (5 points)
Suppose that $A[1..n]$ is an array of integer numbers, and some value k occurs at least $\lfloor n/2 \rfloor + 1$ times in this array. Write an efficient algorithm for finding this value and give the running time of your algorithm.