Purdue University | Gopal/Hambrusch
CS 381 | Spring 2008

Assignment 3

Due: Wednesday, February 6, 2008 (before class)

1) You are given an (unsorted) array A consisting of n positive integers and an integer k such that $n > k \geq 2$. The elements in the array have a special structure: they are distinct except for one integer that occurs *more than* n/k times. We call this element the repeating element. You are asked to design algorithms finding the repeated element.

When $k = 2$, the problem is known as finding the *majority* element as it is an element that occurs more than $n/2$ times (the repeated element is also the median). For $k = 3$, the problem is sometimes called finding the trijority element. Note that identifying any two identical elements in A identifies the repeated element as all other elements are unique.

(a) (10 points) Give an algorithm that runs in $O(nk)$ time and uses $O(n)$ space.
Hint: First, identify a set of at most k elements that are possible candidates for the repeated element. What can you say about their order statistics? Use the linear time selection algorithm as a subroutine to check these potential candidates.

(b) (15 points) Give an algorithm that runs in $O(n \lg k)$ time and uses $O(n)$ space. It is not necessary to provide a solution for (a) if you have a correct $O(n \lg k)$ solution. (A correct solution for (b) and no answer for (a) will give 25 points.)
Hint: Use the same idea as in (a). To improve the time, avoid scanning the same elements of the array multiple times. To do this, augment the linear time selection algorithm.

2) (15 points) Consider again the problem of finding the repeated element defined in 1). Below is the outline of a randomized algorithm:

1. Pick a random array index j ($1 \leq j \leq n$).
2. Compare $A[j]$ with each element of the array and count the number of its occurrences.
3. If $A[j]$ occurs more than n/k times then output $A[j]$, else go to 1.

(a) What is the probability of picking the repeated element in Step 1?
(b) How many times do you expect to execute Step 1? Analyze the expected running time of the randomized algorithm.

3) (10 points) Assume that N persons checked their coats in a restaurant. The coats get mixed and each person gets back a random coat. We want to compute the expected number of people getting back their own coat.

(i) Let the indicator random variable X_i denote the event that the i-th person gets back his own coat. Compute $E[X_i]$, for any i.
(ii) Let the random variable X denote the number of persons that get back their own coat. Using (i) and linearity of expectation, compute $E[X]$.