CS 263 – Analysis of Algorithms

Homework 3, 50 Points
Due: Sunday, February 1, 11:00pm

This homework must be turned in electronically using the course GradeScope website.

Solutions **must** be submitted as a PDF file.

1. Suppose that Bob wants a constant-time method for implementing the random(k) method, which returns a random integer in the range [0, k − 1]. Bob has a source of unbiased bits, so to implement random(k), he samples $\lceil \log k \rceil$ of these bits, interprets them as an unsigned integer, K, and returns the value K mod k. Show that Bob's algorithm does not return every integer in the range [0, k − 1] with equal probability.
2. Suppose you have a collection, S, of n distinct items and you wish to select a random sample of these items of size exactly $\lceil n^{1/2} \rceil$. Describe an efficient method for selecting such a sample so that each element in S has an equal probability of being included in the sample.
3. Suppose you have a collection, S, of n distinct items and you create a random sample, R, of S, as follows: For each x in S, select it to belong to R independently with probability $1/n^{1/2}$. Derive bounds on the probability that the number of items in R is more than $2n^{1/2}$ or less than $n^{1/2}/2$.
4. Show that the randomized quick-sort algorithm runs in O(n log n) time with probability at least 1-1/n.
5. Suppose that we have n jobs to distribute among m processors. For simplicity, we assume that m divides n. A job takes 1 step with probability p and k > 1 steps with probability 1 − p. Use Chernoff bounds to determine upper and lower bounds (that hold with high probability) on when all jobs will be completed if we randomly assign exactly n/m jobs to each processor.