Problem Set 4

Data Structures and Algorithms, Fall 2019

Due: October 10, in class.

Reading Assignment

Chapter 7 of "Discrete Mathematics and Its Applications (7ed)" provides a quick introduction to discrete probability theory. Appendix C of CLRS serves the same purpose, but is more concise. Please read these material—especially Section 7.2 and 7.4 of Rosen's book, and Appendix C.2 to C.4 of CLRS—to get yourself familiar with concepts like "random variable" and "expectation".

I *will* assume you have read these material and are familiar with basic discrete probability theory starting from the class on October 10^{th}. We will use these tools extensively throughout this course. To begin with, we will analyze the average case performance of `RandomizedQuickSort`.

From CLRS

Exercise 7.3-2, 7.4-2 (be formal and rigorous), 7.4-5. Problem 7-2, 7-5. Exercise 8.1-1, 8.1-4, 8.2-4, 8.4-4. Problem 8-3, 8-6.

Additional Problem One

[This is NOT a bonus problem, and you ARE required to solve it.]

We are given n bolts and n nuts of different sizes, where each bolt exactly matches one nut. Our goal is to find the matching nut for each bolt. The nuts and bolts are too similar to compare directly; however, we can test whether any nut is too big, too small, or the same size as any bolt.

(a) Prove that in the worst case, $\Omega(n \log n)$ nut-bolt tests are required to correctly match up the nuts and bolts.

(b) Now suppose we would be happy to find most of the matching pairs. Prove that in the worst case, $\Omega(n \log n)$ nut-bolt tests are required to find $n/2$ arbitrary matching nut-bolt pairs.

(c) Prove that in the worst case, $\Omega(n + k \log n)$ nut-bolt tests are required to find k arbitrary matching pairs.

(d) Describe a randomized algorithm that finds k matching nut-bolt pairs in $O(n + k \log n)$ expected time. You should also briefly argue your algorithm indeed has the desired expected runtime.