CS 223: Homework 1. Due February 2.

Do the following exercises from Mitzenmacher and Upfal (blue book version) (5 points each): 1.22, 3.15, 3.19.

Do the following exercises from Mitzenmacher and Upfal (blue book version) (10 points each): 4.12, 4.13.

Additional problem, using the algorithm with predictions framework (15 points).

Let us suppose that we have N jobs to run on our system. Of these jobs, n are short jobs, that each take time s, and m are long jobs, that each take time $\ell > s$. Here $N = n + m$. Jobs run one at a time, and a scheduler schedules the jobs in some order. A job's waiting time it waits before being served. So, for example, a short job that is scheduled after two long jobs has waiting time 2ℓ.

- If we know nothing about the jobs, it makes some sense to just schedule the jobs randomly (according to a random permutation). What is the expected waiting time of a job in this case, as a function of n, m, s, ℓ. (You may want to find the expected waiting times for small and large jobs separately, and then combine them.)
- If we know which jobs are small and which jobs are long, then we should schedule short jobs before long jobs. (You don't have to prove this, but you may think about how you would prove it.) In this case, there is no randomness, so you can deterministically calculate the average waiting time over all jobs. Do this computation.
- Of course, even if we schedule all short jobs before all long jobs, we could calculate the average waiting time with probabilistic analysis as follows. Assume the short jobs are in a random order, and similaly the long jobs are in a random order. Find the expected waiting times for small and large jobs separately, and then combine them to determine the average waiting time. (Does your answer match what you got in the previous step?)
- We might not know which jobs are small and which jobs are long precisely. Suppose the scheduler can predict whether a job is short or long, but it makes some mistakes. In particular, a short job is classified as a long job with some probability p, $0 < p < 1$ (independently for each job), and similarly a long job is classified as a short job with some probability q, $0 < q < 1$. The scheduler then orders the jobs that are classified as short in a random order, and the jobs that are classified as long in a random order. What is the expected waiting time of a job in this case, as a function of n, m, s, ℓ, p, q. (There are many way to calculate this; you might want to use variables $X_{i,j} = 1$ if the job i is scheduled after job j, where the jobs are initially names job 1, job 2, etc.)
- (Open-ended) Suppose that $n = m$, that is there are as many short jobs as long jobs. Under what conditions does the predictor perform better than just randomly ordering all the jobs? Can you explain this condition?