Randomized Algorithms. Exercises for 22.09

September 17, 2009

We will discuss exercises 1.2-1.4 (in the main text) and problems 1.1-1.4 (at the end of the chapter) of the book. Please study these exercises at home, write down a solution to at least one exercise of your choice and prepare to present a solution to one more of the exercises in the class on Tuesday.

For those of you who do not (yet) have a copy of the book, the following is a "short version" of these exercises.

ex.1.2 Consider a version of the algorithm for MinCut, where in each step we choose a random pair of vertices to merge. Show a graph on which this strategy has a low probability of finding a minimal cut.

ex.1.3 Show that if we have a Monte Carlo algorithm and a method to efficiently check the correctness of the solution, then we may construct a Las Vegas algorithm.

ex.1.4 Suppose $0 < \epsilon_2 < \epsilon_1 < 1$, and that you have an algorithm that produces correct solution with pr. $1 - \epsilon_1$. How many times you need to execute it to get success probability $1 - \epsilon_2$?

pr.1.1 Describe how to use a biased coin (e.g. a coin that comes up heads with pr. p) to "construct" an unbiased coin. Can you make it more efficient when generating more random bits at once?

pr.1.2 Describe how to use random bits to generate uniformly random elements from a given set. (note that the cardinality of the set is not necessarily a power of 2)

pr.1.3 Use random bits to sample numbers from $\{2, \ldots, 12\}$ such that the probability of drawing a number is as if one would roll two dices and take the sum of the outcomes.

pr.1.4 Describe how to use random bits to generate random permutations.