

CS 6550 – Design and Analysis of Algorithms – Fall 2007
Homework 2 – Due Tuesday, September 25

- a) We showed in class that if n balls are thrown (independently and randomly) into n bins, then the expected number of empty bins approaches n/e . What is the expected number of bins when m balls are thrown into n bins?
 - b) Again, we are throwing m balls into n bins. Give the best bound you can on m to ensure that the probability of there being a bin containing at least two balls is at least $1/2$.
2. Suppose you are given a biased coin that has $\Pr[\text{heads}] = p \geq a$, for some fixed $a > 0$, without being given any other information about p . Devise a procedure for estimating p by a value \tilde{p} such that you can guarantee that $\Pr[|p - \tilde{p}| > \epsilon p] < \delta$, for any choice of the constants $\epsilon > 0$ and $\delta > 0$. Let N be the number of times you need to flip the biased coin to obtain the estimate. What is the smallest value of N for which you can still give this guarantee?
3. In class we analyzed the scenario where you play repeated games with a friend where you have a $2/3$ chance of winning each game. If you play 30 games, what is the chance that you win 5 or fewer games? Work out the bound using both Chebyshev's inequality and one of the Chernoff bounds.
4. Let a_1, \dots, a_n be n real numbers chosen independently and uniformly from the range $[0, 1]$.
 - a) Describe an algorithm for sorting these numbers that has linear expected running time.
 - b) Show that the linear running time occurs with high probability.
5. Suppose you are playing a game against an evil all-knowing adversary.

The game is made out of a 2×2 chessboard, with coins on each of the 4 squares. At each round, you can decide to flip either one or two of the coins by specifying which squares you want flipped (i.e., the bottom left and the upper right coins). The adversary then rotates the board, either by 0, 90, 180, or 270 degrees.

You win when all the coins on the board are heads or when they are all tails. Easy, eh? The catch is that all the coins are covered, and the initial configuration of coins is unknown to you.

 - a) Describe a *deterministic* algorithm you can use to guarantee that you win in a finite number of steps. How many rounds are required by your algorithm?
 - b) We now generalize this game to n coins sitting around a circle. You can flip any number of coins in each move, and now the adversary gets to rotate the board to any of the n positions during his turn.

Show that there is no longer a deterministic strategy for winning this game when $n = 3$.
 - c) Show a randomized strategy you can use that is guaranteed to terminate with a win (with probability approaching one as the game continues). What is the expected number of rounds following this strategy?

d) In the last variant, we make two changes. This time you are only allowed to flip at most one coin in each turn. However, to help you, you get to see the current value of the coin you picked and then, after seeing whether it is heads or tails, you get to decide whether to flip that coin or not. Describe your new randomized strategy and compute the expected number of rounds it requires before you win.