

Homework #9
Algorithms I
600.463
Spring 2017

Due on: Thursday, May 4th, 11:59pm

Late submissions: will NOT be accepted

Format: Please start each problem on a new page.

Where to submit: On Gradescope, under HW9

Please type your answers; handwritten assignments will not be accepted.

To get full credit, your answers must be explained clearly,
with enough details and rigorous proofs.

May 4, 2017

Problem 1 (20 points)

You are given a biased coin but you don't know what the bias is. Can you simulate a fair coin? Please prove the correctness of your solution.

Answer Problem 1

The idea is to still use the biased coin to come up with fair results. We want to "remove the bias" by altering the procedure and which results we take into as legitimate and fair. Usually with a fair coin, where there is a 50-50 chance of flipping either heads or tails, we only need to flip it once to trust the result. But with a biased coin, what we need to do is figure out where the bias occurs. If the biased coin, however, has a side where there is 0-percent probability for a successful flip, then this method will not work. We can do this by tossing the coin twice. If the result of each toss is the same (both heads, or both tails), then ignore the results of that pair of simulation and start again. If we toss the biased coin twice and we get heads for one and tails for the other, then we should record the result of the first coin flip, and omit the second. What we are exploiting in this biased coin is the fact that the bias doesn't change between the flipping of the coins, as the coin flips are independent. What this means is we can successfully simulate coin tosses where we can have two outcomes that have the same probability- which we do by omitting the scenarios of two consecutive flips of heads or tails. This also works with 2 tosses, 4 tosses, 8 tosses, etc.. (for all 2^k tosses)...

Correctness of Solution:

Let the variable p represent the probabilities where we flip the biased coin twice and we get two different values (HT or TH but NOT HH and TT). Let n be the number of times we have to repeat this procedure (or, in other words, let it be the number of trials conducted until we reach a "fair" start to our procedure). Suppose, like in the above explanation, that after two flips of our biased coin we get the same results (either HH or TT). We can model this in the following equation: $n = 2p + (1 - p)(2 + n)$. Upon expanding this equation we get, $t = 2p + 2 + n - 2p - np$. Then simplifying we are left with: $t = 2/n$, where $n = 2x(1 - x)$ (this is derived from when $x(1 - x)$ is the probability of getting two flips of heads and $(1 - x)x$ is for getting two flips of tails, as such the probability in two flips of getting two different results would yield $x(1 - x) + (1 - x)x$. And such we would expect that the following number of flips will result in a fair coin simulation: $1/x(1 - x)$).

Problem 2 (20 points)

The following approach is often called *reservoir sampling*. Suppose we have a sequence of items passing by one at a time. We want to maintain a sample of one item with the property that it is uniformly distributed over all the items that we have seen at each step. Moreover, we want to accomplish this without knowing the total number of items in advance or storing all of the items that we see. Consider the following algorithm, which stores just one item in memory at all times. When the first item appears, it is stored in the memory. When the k -th item appears, it replaces the item in memory with probability $1/k$. Explain why this algorithm solves the problem. (Exercise 2.18 from the Book “**Probability and Computing: randomized algorithms and probabilistic analysis**”, Mitzenmacher and Upfal 2009. https://catalyst.library.jhu.edu/catalog/bib_5738970)

Answer Problem 2

Correctness of Solution/Proof By Induction:

We need to prove the following: that at some time t , the probability that some random variable (the k -th item appears that replaces the item in memory) is $1/k$, for all values $k \geq i \geq 1$. So let's say there are a series of items, x_1, x_2, \dots, x_n observed at some time x_t . Then let's say that the variable R_t represents the k -th item that replaces an item in memory.

Base Case: $k = n$, for some $t = 1$

TRUE because $R_t = x_1$ where there is a probability of 1 (as the time factor is 1, and such a replace in memory will most certainly happen, as $1/k = 1/1 = 1$). Then let's go ahead and state that this will also hold true for some $k = n$. So, after some n -th item replaces the item (with probability of $1/n$), and such the probability of the i -th value is the item that is in memory is the probability $1/n$ (for $i = 1, 2, \dots, n$).

Inductive Step: $k = n + 1$

This step is when after the $n + 1$ item replaces whatever item in memory, the probability will have to be $1/n + 1$, and such the probability that some i -th value is the item that is in memory is the probability $1/n + 1$ for some $i = 0, 1, \dots, n + 1$. After the replacement, these are the following cases that we can have: either the newest item, (the $n + 1$ item) and its probability being $(1/n + 1)$ OR that after being replaced, the item in memory is the same item as the previous where the probability will then be the following: $1 - 1/n + 1 = n/n + 1$. As such, we must recall that previous item can be an item out of the following $(1, \dots, n)$, and the probability for that will be $(\frac{1}{n} \times \frac{n}{n+1} = \frac{1}{n+1})$. As such, the probability that the i -th item is the one in memory is $1/n + 1$ for some $i = 1, \dots, n + 1$.

Therefore, as we can see through induction, we have proved the inductive step works, and so it is evident that when the k -th item appears, it does indeed replace the item in memory with the probability of $1/k$.