Streaming.

**Sampling:** Given a number $k$, you want to maintain a random sample of size $k$ from the stream. I.e., for each $n \geq k$, the set you have at time $n$ should be a random subset of the prefix $a[1:n]$, each of the $\binom{n}{k}$ subsets of size $k$ from this prefix should be equally likely.

1. For $k = 1$, show that the algorithm: pick the first element. When faced with the $n^{th}$ element, with prob. $1/n$ discard the element in your hand and pick the new element, and with prob. $1 - 1/n$ keep the element in hand.

2. Give an algorithm for general $k$. (What would you do when faced with the $n^{th}$ element? With what probability should you pick this element? Which element should you drop?)

**Missing Numbers:** Suppose I give you a stream of $n - 1$ elements, which contains all the numbers from 1 thru $n$ except one of them. (The numbers do not appear in sorted order.) Clearly you can figure out the missing number by storing all $n - 1$ numbers and looking for the missing number. How can you output the missing number with only $O(\log n)$ space? What if there are two missing numbers: can you again use only $O(\log n)$ space?
Jaccard Similarity.
Suppose that we have two nonempty sets $A$ and $B$ that are subsets of the same universe $U$. We can estimate how similar they are with Jaccard similarity, defined as

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Note that this value will always be between 0 (when $A$ and $B$ are disjoint) and 1 (when $A = B$).

Suppose that we are being streamed the two sets $A$ and $B$ from the universe $U$ and wish to use a constant amount of space. Also suppose we have a constant-space, perfect hash family $H : U \rightarrow \mathbb{Z}$ with the additional property that $P[h(a) < h(b) < h(c)] = \frac{1}{6}$ for all $a, b, c \in U$, $a \neq b \neq c$. (In other words, we have that $h(a) = h(b) \Rightarrow a = b$, and that any permutation of hash orderings is equally likely.)

Propose an algorithm for estimating the Jaccard similarity. (That is, give an algorithm that outputs a number on the range $[0, 1]$, and outputs the Jaccard similarity in expectation.) How can we improve the expected error in our estimation?

Fingerprinting.

Many Patterns: You are given a set of patterns $P_1, P_2, \ldots, P_k$ of equal length (all of them having length $n$) and a text $T$ of length $m$. Give an algorithm to find all the locations $i$ such that some pattern $P_j$ occurs as a substring of $T$ starting at location $i$. The expected runtime should be $O(kn + m)$, and the probability of error is at most 0.01.  

\[\text{1} \text{ Assume you can do arithmetic operations on numbers of size } O(\log(kmn)) \text{ in constant time, even modulo a prime.}\]