CS 466/666 Algorithm Design and Analysis **Last Updated:** May 23, 2022
Homework 2 **Due Date: June 6, 2022**

You are allowed to discuss with others but not allowed to use any references except the course notes and the books "Probability and Computing" and "Randomized Algorithms". Please list your collaborators for each question. This will not affect your marks. In any case, you must write your own solutions.

There are totally 50 marks, and the full mark is 40. This homework is counted 8% of the course. The extra marks will not be carried to other parts of the course (also not to other assignments). Please read the course outline for the late submission policy.

---

1. **Streaming Sampling**

   (10 marks) Suppose we have a long sequence of numbers coming one at a time. We would like to maintain a set of numbers of size k with the property that for each number we have seen so far, the probability that the number appears in the set are equal.

   We want to accomplish this without knowing the total number of items in advance or storing all of the items that we have seen.

   Prove that the following simple algorithm works. When the first k numbers come, we put it in the set. After that when the m-th number appears, with probability k/m, we replace a random number in the set with the m-th number.

2. **Coupon Collectors**

   (10 marks) Now we know that it requires $\Theta(n \log n)$ coupons to collect n different types of coupons, with at least one coupon per type. We wonder whether it would be more efficient if a group of k people cooperate, such that each person buys cn coupons and as a group they have at least k coupons per type (so that everyone gets a complete set of coupons).

   Prove the best bound you can on k to ensure that, with probability at least 0.9, each person only needs to buy at most $10n$ coupons. You will get full marks if k is of the correct order in terms of n.

3. **Maximum Load by Universal Family**

   (10 marks) We have shown that the maximum load when n items are hashed into n bins using a hash function chosen from a 2-universal family of hash functions is at most $\sqrt{2n}$ with probability at least $1/2$. Generalize this argument to k-universal hash functions. That is, find a value such that the probability that the maximum load is larger than that value is at most $1/2$. Then, find the smallest value of k such that the maximum load is at most $3 \ln n / \ln \ln n$ with probability at least $1/2$ when choosing a random hash function from a k-universal family.

4. **Approximate Median in Sublinear Time**

   (10 marks) We would like to find an approximate median of n distinct numbers in sublinear time. To do so, we sample $m \ll n$ numbers with replacement, find the median c of these m numbers, and report c as the approximate median of the n numbers. Let the sorted list of the n numbers be $\{x_1, x_2, \ldots, x_n\}$ and so the true median is $x_{n/2}$. The approximate median is said to be a $\pm k$-approximation if $c \in [x_{\frac{n}{2}-k}, x_{\frac{n}{2}+k}]$. Suppose we want the algorithm to succeed to find a $\pm k$-approximation with probability at least 0.9999. What is the tradeoff between m and k? How large should we set m if we want $k \leq \epsilon n$ for some small constant ϵ? How large should we set m if we want $k \leq n^{1-\eta}$ for $\eta \leq 1/2$?

5. ***k*-Wise Independence Bits** (Bonus)

   Suppose that we are given m vectors $v_1, \ldots, v_m \in \{0,1\}^b$ such that any k of the vectors are linearly independent modulo 2. Let $v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,b})$ for $1 \leq i \leq m$. Let X be chosen uniformly at random from $\{0,1\}^b$. Let $Y_i = (\sum_{j=1}^{b} v_{i,j} X_j) \bmod 2$. Show that the Y_i are uniform, k-wise independent random bits.