1. Let \( f_i \) be the frequency of element \( i \) in the stream. Modify the Mishra-Gries algorithm to show that for a stream of length \( m \), one can compute quantities \( \hat{f}_i \) for each element \( i \) such that

\[
\frac{m}{k} \leq \hat{f}_i \leq f_i
\]

2. Given a stream with \( m \) data items and a threshold \( n/k \) for some integer \( k \), what is the minimum sample size such that if there is an item \( q \) whose frequency exceeds \( n/k \), there will be at least \( n_q \) copies of \( q \) in the sample with probability \( \geq 2/3 \).

You may assume that each element of the stream is being sampled with probability \( p \) independently (leading to an expected sample size of \( mp \)). Note that there can be \( k \) such items (exceeding the threshold), so you have to take care using the union bound. Calculate the minimum value of \( p \) using Chernoff bounds.

3. Recall the reservoir sampling algorithm described in the class. Prove by induction on \( i \) that after \( i \) steps, the random variable \( X \) is a uniformly chosen element from the stream \( \{x_1, \ldots, x_i\} \).

4. Let \( Y_1, \ldots, Y_i \) be \( t \) i.i.d. random variables. Show that the variance of \( Z \), denoted by \( \sigma^2(Z) \), is equal to

\[
\frac{1}{t} \sigma^2(Y_1).
\]

5. Suppose \( E_1, \ldots, E_k \) are \( k \) independent events, such that each event occurs with probability at most \( 1/4 \). Assuming \( k \geq 4 \log(1/\delta) \), prove that the probability that more than \( k/2 \) events occur is at most \( \delta \).

6. Let \( a_1, a_2, \ldots, a_n \) be an array of \( n \) numbers in the range \([0, 1]\). Design a randomized algorithm which reads only \( O(1/\varepsilon^2) \) elements from the array and estimates the average of all the numbers in the array within additive error of \( \pm \varepsilon \). The algorithm should succeed with at least 0.99 probability.

7. Consider a family of functions \( H \) where each member \( h \in H \) is such that \( h : \{0, 1\}^k \rightarrow \{0, 1\} \). The members of \( H \) are indexed with a vector \( r \in \{0, 1\}^{k+1} \). The value \( h_r(x) \) for \( x \in \{0, 1\}^k \) is defined by considering the vector \( x_0 \in \{0, 1\}^{k+1} \) obtained by appending 1 to \( x \) and then taking the dot product of \( x \) and \( r \) modulo 2 (i.e., you take the dot product of \( x_0 \) and \( r \), and \( h_r(x) = 1 \) if this dot product is odd, and 0 if it is even). Prove that the family \( H \) is three-wise independent.

8. Recall the setting for estimating the second frequency moment in a stream. There is a universe \( U = \{e_1, \ldots, e_n\} \) of elements, and elements \( x_1, x_2, \ldots \) arrive over time, where each \( x_i \) belongs to \( U \). Now consider an algorithm which receives \textbf{two} streams – \( S = x_1, x_2, x_3, \ldots \) and \( T = y_1, y_2, y_3, \ldots \). Element \( x_t \) and \( y_t \) arrive at time \( t \) in the two streams respectively. Let \( f_i \) be the frequency of \( e_i \) in the stream \( S \) and \( g_i \) be its frequency in \( T \). Let \( G \) denote the quantity \( \sum_{i=1}^n f_i g_i \).

   - As in the case of second frequency moment, define a random variable whose expected value is \( G \). You should be able to store \( X \) using \( O(\log n + \log m) \) space only (where \( m \) denotes the length of the stream).

   - Let \( F_2(S) \) denote the quantity \( \sum_{i=1}^n f_i^2 \) and \( F_2(T) \) denote \( \sum_{i=1}^n g_i^2 \). Show that the variance of \( X \) can be bounded by \( O(G^2 + F_2(S) \cdot F_2(T)) \).

9. You are given an array \( A \) containing \( n \) distinct numbers. Given a parameter \( \varepsilon \) between 0 and 1, an element \( x \) in the array \( A \) is said to be a near-median element if its position in the sorted (increasing order) order of elements of \( A \) lies in the range \([n/2 - \varepsilon n, n/2 + \varepsilon n]\). Consider the following randomized algorithm for finding a near-median: pick \( t \) elements from \( A \), where each element is picked uniformly and independently at random from \( A \). Now output the median of these \( t \) elements. Suppose we want this algorithm to output a near-median with probability at least \( 1 - \delta \), where \( \delta \) is a parameter between 0 and 1. How big should we make \( t \)? Your estimate on \( t \) should be as small as possible. Give reasons.