Problem 1. We have learned that the count-min sketch allows us to answer point queries with a probabilistic guarantee. However, currently, such a guarantee holds only for one query. In this problem, we will see how to extend the guarantee to all queries simultaneously.

Let us consider a cash-register stream. Recall that the underlying dataset is an array \(A\) of \(n\) real values. Initially, all the elements of \(A\) are 0. Each update has the form \((i, v)\), indicating that we should increase the \(i\)-th (\(1 \leq i \leq n\)) element \(A[q]\) in \(A\) by \(v\). Given an array index \(q\) (i.e., \(1 \leq q \leq n\)), a point query returns a value \(\hat{A}[q]\) such that \(A[q] \leq \hat{A}[q] \leq A[q] + \epsilon \|A\|\), where \(\|A\| = \sum_{i=1}^{n} A[i]\). Note that there are \(n\) different point queries (i.e., \(n\) choices for \(q\)).

Describe a data structure that uses \(O\left(\frac{1}{\epsilon} \log \frac{n}{\delta}\right)\) words, such that with probability at least \(1 - \delta\), we are able to give correct answers to all \(n\) point queries (simultaneously).

Problem 2. In probabilistic algorithms, a success probability of at least \(1 - \frac{1}{n^c}\) (where \(c > 0\) is a constant) is termed a high probability, when the input set has size \(n\). In the setup of Problem 1, prove that there is a structure that uses \(O\left(\frac{1}{\epsilon} \log n\right)\) words such that with high probability, we are able to give correct answers to all \(n\) point queries (simultaneously) – this is true regardless of \(c\).

Problem 3. Recall that a chief motivation of applying the reservoir algorithm is to ensure that the sample set should have a specific size. If, on the other hand, we only want to have a probabilistic control over the size, then usually there is a simpler way to sample (without replacement), as we will find out in this problem.

Let \(S\) be a set of \(n\) elements. We obtain a sample set \(R\) by including each element of \(S\) independently with a probability \(p\), where \(p\) satisfies \(p \geq \frac{\log 2}{n}\). Prove that when \(n\) is larger than a certain constant, the size of \(R\) is at most \((1 + \epsilon)np\) with probability at least \(1 - 1/n\), where \(\epsilon\) can be any arbitrarily small constant.