Algorithms for Big Data – no. 22934

Problems Set 11

General instructions: Please keep your answers short and easy to read. You can use algorithms, results, calculations or notation that appear in our course material without repeating them, unless asked explicitly to redo them.

1. Show that given any algorithm that runs in time on inputs of size with probability of error , one can convert it into a new algorithm that runs in time with probability of error at most . Hint: run times and take the majority answer. Use Chernoff bounds.

2. You are given an approximation scheme for such that
   —— , and runs in time polynomial in and
   . Construct an approximation scheme for such that
   —— , and runs in time polynomial in ,
   and –.

3. (Coupon Collector Problem). Given a die with sides. What is the expected number of times you need to roll the die in order to see each of the sides? Hint: Given that you saw sides, how many times do you need to roll the die to see the side? Then use linearity of expectation.

4. Recall that stands for the length of the input stream (i.e. the number of the tokens) and is the number of different possible tokens. Assume that the value of every token is between 1 and , i.e. . Design a streaming algorithm that at any point (not known in advance) receives a query (i.e. a subset of tokens values) and outputs and estimate what fraction of tokens in the stream belong to within additive error . Note that is given only at query time (not in advance). Hint: Maintain random samples and use them to estimate the fraction in .

5. a. Suppose we are guaranteed that some token in the stream appears more than half the time, i.e., there exists (unknown) token with frequency . Design a streaming algorithm with space complexity bits that finds this item . Hint: Store only two items.

   b. Next, extend your algorithm to output also a approximation to its frequency . Make sure to clearly state the space complexity of your algorithms. Hint: Use ideas of question 4.