CSE 632: Analysis of Algorithms II — Fall 2017

Homework 5

Lecturer: Shi Li — **Deadline: 12/22/2017**

Instructions and Policy. Each student should write their own solutions independently and needs to indicate the names of the people you discussed a problem with. It is highly recommended that you think about each problem on your own for enough time before you discuss with others. It is recommended that you type your solutions using latex and submit the pdf file via UBLearns.

Problem 1 (50 points) We consider the streaming algorithms of deciding if a given graph $G = (V, E)$ contains a triangle or not. For simplicity we assume $V = [n]$, and we know n and m ahead of time. We are given the set of edges in stream; that is, the stream is $(u_1, v_1), (u_2, v_2), (u_3, v_3), \cdots$, where each (u_i, v_i) is an edge in E. Show that in order to decide whether G contains a triangle needs $\Omega(n^2)$ space, even for randomized algorithm that gives the answer correctly with probability at least $2/3$.

Hint: you can reduce from the index problem and apply the lower bound for the communication complexity of the randomized one-way protocol for the problem.

Problem 2 (50 points) We are given a sequence $(i_1, \Delta_1), (i_2, \Delta_2), \cdots, (i_n, \Delta_n)$ in a data stream. Initially we have $x_i = 0$ for every $i \in [m]$. Each pair (i, Δ) represents the following operation: $x_i = x_i + \Delta$, where $i \in [m]$ and $\Delta \in \{\pm 1\}$. We need to output $|x|_1 = \sum_{i \in [m]} |x_i|$.

Show that randomization is needed, even if our goal is to approximate $|x|_1$ within *an additive factor* of ϵn. That is, for some small enough constant $\epsilon > 0$, any deterministic streaming algorithm that always outputs a number $E \in \big[|x|_1, |x|_1 + \epsilon n\big]$ must use $\Omega(n)$ space, assuming $m \gg n$.