Your Name	October 22, 2024
6.5240 Problem Set 4	
Collaborators: *Firstname Lastname*	Due: October 30, 2024, 10pm

Homework guidelines: You may work with other students, as long as (1) they have not yet solved the problem, (2) you write down the names of all other students with which you discussed the problem, and (3) you write up the solution on your own. No points will be deducted, no matter how many people you talk to, as long as you are honest. If you already knew the answer to one of the problems (call these "famous" problems), then let us know that in your solution writeup – it will not affect your score, but will help us in the future. It's ok to look up famous sums and inequalities that help you to solve the problem, but don't look up an entire solution.

1. **Testing on strings: concatenations of palindromes.** Let $L = \{uu^r vv^r | u, v \in \{0,1\}^*, 2(|u| + |v|) = n\}$. We saw in class that given a string x, distinguishing $x \in L$ from x that is ϵ-far (meaning that $> \epsilon n$ bits of x need to be changed in order to make x a member of L) requires $\Omega(\sqrt{n})$ queries. Give an algorithm for this problem that uses $O(\sqrt{n}\log n/\text{poly}(\epsilon))$ queries to the input. The running time does not have to be sublinear.

2. **Lower bounds for estimating the weight of a maximum matching.** Give a lower bound on computing a multiplicative estimate on the maximum matching size of a graph G: for any $d \leq n/10$, give two distributions over unweighted graphs of *average degree* d such that

   a) graphs in one distribution have a matching of size at least twice the maximum matching size of graphs in the other distribution

   b) in order to distinguish the two distributions with constant probability of success, one must make at least $\Omega(d)$ queries. Queries can be neighbor queries, pair queries, or degree queries.

   For full credit, you need only prove this lower bound against *nonadaptive* algorithms (algorithms that specify all queries before reading any of the answers). If you like, you may optionally also prove a lower bound against adaptive algorithms, as the statement is true but the analysis is a bit tedious.

3. **Vertex covers and distance to monotonicity.**

   A *vertex cover* V' of a set of edges E' is a set of nodes such that every edge of E' is adjacent to one of the nodes in V'.

For graph $G = (V, E)$, let the *transitive closure graph* $TC(G)$ be the graph $G^{(tc)}(V, E^{(tc)})$ where $(u, v) \in E^{(tc)}$ if and only if there is a directed path from u to v in G.

Let $f : V \to \{0, 1\}$ be a labeling of the vertices of a known directed acyclic graph G by 0 and 1. For any pair of nodes x and y, we say that $x \leq_G y$ if there is a path from x to y in G. We say that f is *monotone* if for all $x \leq_G y$, $f(x) \leq f(y)$. The *minimum distance of f to monotone* is the minimum number of nodes that must be relabeled in order to turn f into a monotone function.

Let E' be the set of violating edges in $TC(G)$ according to f. Show that the minimum distance of f to monotone is equal to the minimum size of a vertex cover of E'.