1. In the following questions, assume that all input graphs start out with unique IDs.

(a) Given a graph of max degree at most $\Delta$, show that the edges can be decomposed into at most $\Delta$ oriented forests (where each node has outdegree at most 1, the roots have outdegree 0, and edges point along the path to a root). Show that given a node, the edge in oriented forest $i$ and the direction of the edge, can be computed in $O(\Delta)$ sequential time.

(b) Give a distributed algorithm for 6-coloring trees. Assume that the tree can be viewed as a rooted tree in which children know who their parent is. For full credit, your algorithm should run in $k = O(\log^* n)$ rounds. Note that this gives an LCA for 6-coloring trees which runs in $2^{O(\log^* n)} = O(\log^* n)$ probes. Hint: Consider algorithms in which a node $u$ looks at its parent $v$ and recolors itself based on the location of the first bit which differs between $u$ and $v$.

(c) Given graph $G$ along with a $c$-coloring of the nodes (assume you can query the coloring of any node in 1 step). Show how to find an MIS in $c$ distributed rounds.

(d) Combine the above to give an LCA for $6^\Delta$ coloring a degree at most $\Delta$ graph $G$.

2. In class, we gave an LCA for the spanner problem that works for graphs of max degree at most $n^{3/4}$. Show how to construct an LCA for the spanner problem for any graph. For full credit, your runtime should still be $O(n^{3/4})$ per query.

Hint: (1) Handle the nodes that have degree between $\sqrt{n}$ and $n^{3/4}$ with a different setting of parameters for determining centers. (2) For nodes of degree at least $n^{3/4}$, partition the edges into groups of size $n^{3/4}$, and add a rule 3 edge $(u,v)$ whenever $v$ introduces $u$ to a new cluster within its partition (this will allow more edges in the final graph, but show that it won’t destroy the sparsity of the spanner).

3. Say that $f : \{0,1\}^n \rightarrow \{0, \ldots, n\}$ is monotone if for all $x, y$ such that $x_i \leq y_i$ for $i = 1, \ldots, n$, then $f(x) \leq f(y)$. Show that distinguishing whether $f$ is monotone from the case that $f$ is $\epsilon$-far from monotone (i.e., there is no monotone $g$ such that $f$ and $g$ differ on at most $\epsilon$-fraction of the domain $\{0,1\}^n$) requires $\Omega(n)$ queries. Hint: reduce from the communication complexity problem of disjointness. Another hint: Let $|x|$ be the number of 1’s in $x$. Let Alice define $p(x)$ to be $-1$ if the parity of the input bits in her set is 1, and 1 if the parity is 0. Let Bob define $q(x)$ similarly. Let them compute $h(x) = 2|x| + p(x) + q(x)$. 