Assignment 4

Timothy Yong, Eric Bronner, Aedan Dispenza, Jason Davis

10/6/14

PROBLEM 1

a) Algorithm: Run the Dueling Algorithm for constructing a cartesian tree, but if a node is about to be inserted under a node of the same value, do not insert it.

b) Counter Example: $\exists u, v$, both of which are subtrees of T, s.t. $h(u) < h(v)$. Given that u is visited first by the Eulerian Tour, and a Cartesian Tree is thus constructed from this Eulerian Tour, after doing the Cartesian Update from v, $\exists$ at least one edge which exists in the Cartesian Tree, but not in T.

PROBLEM 2

Given that k is the amount of bits in the representation of a number, and $logn$ is the amount of bits that can be evaluated in constant time, then this algorithm is $O(1)$ if $k \leq logn$, or $O(\frac{k}{logn})$ if $k > logn$.

This algorithm runs in $\frac{logb}{logn} = log_n b$ time. Given that $b < n$, this algorithm is $O(1)$.

PROBLEM 3

Algorithm $LCA_T(V')$:

1 Do a preorder traversal on $T(v)$ to get one of the outermost leaves p.
2 Do a postorder traversal on $T(v)$ to get one of the outermost leaves q.
3 Run $LCA(p, q)$.

Algorithm M:

1 Get $S(v)$ by running a DFS on the tree rooted at v.
2 $LCA_T(S(v))$

PROBLEM 4

If we do the set cover of all edges in graph G, in Polynomial time, then we have determined that for the input graph G, $\exists V' \leq K$, s.t. V''s edges comprise all of the edges inside of graph G, therefore solving Vertex Cover.