Problem Set No. 2

1. Consider the list-based (fast-find) method for disjoint set union, as compared to the tree-based (fast-union) method, with path compression. Consider any intermixed sequences of links and finds. Suppose both methods are used to perform the sequence concurrently, and that they both do the links in the same way, in the sense that throughout the process the canonical elements are the same for both methods. Prove that the list-based method makes at least as many changes of canonical-element pointers as the tree-based method makes of parent pointers.

2. Prove that the tree-based method with compression but without linking by rank takes $O(m \log n)$ time to perform m finds intermixed with links, if n is the number of initial singleton sets and $m \geq n$.

   Hint: Mimic the potential-based proof of the $O(m\alpha(n))$ time-bound proof for compression with linking by rank. Initialize all ranks to zero. When performing a link that makes root x point to root y (x becomes a nonroot), replace $r(y)$ by $\max\{r(x)+1, r(y)\}$. For a nonroot node x, define the *level* of x by $k(x) = \lg\lfloor r(p(x)) - r(x)\rfloor$. Define the potential of a node x by $\phi(x) = 0$ if x is root, $\phi(x) = \lfloor \lg(n-1)\rfloor - k(x)$ if x is a nonroot. Define the total potential to be the sum of all the node potentials. Prove that the initial potential is zero, the potential is always nonnegative, and the amortized cost of both a find and a link is $O(\log n)$.

3. (Extra Credit) Define a two-variable version of the inverse Ackermann function α by $\alpha(n,x) = \min\{k | A_k(x) \geq n\}$. Note that as x increase, $\alpha(n,x)$ decreases or stays the same.

   Prove that, beginning with n singleton sets and performing m finds and intermixed links, with m not necessarily at least n, the total time using path compression and union by rank is $O(n + \alpha(n, \lceil m/n \rceil))$.