ECS223a Parallel Algorithms
Homework 1

You are encouraged to talk to other people about these problems, but please **write up the solutions by yourself**. Cite any conversations you had with others, as well as books, papers or Web sites you consulted.

Always explain the answer in **your own words**; do not copy text from the book, other books, Web sites, your friends' homework, etc. Explain your solution as you would to someone who does not understand it, for instance to a beginning graduate student or an advanced undergraduate. Do not give several solutions to one problem; pick the best one.

Please type your homework. If you know LaTeX, use that. If not, you may type your answers in any word processing system and write in mathematical notation by hand as necessary. Include pictures if appropriate; you can draw in pictures by hand or include them in the file.

1. We observed that finding an $O(n)$-sized independent set in a linked list is important for reducing the size of the problem of list ranking. Consider the following randomized algorithm for choosing an independent set in a linked list. Each node in the list chooses a random color in the set $\{0, 1\}$. Then it compares its color with those of its neighbors; if the colors of both of neighbors are larger, the node joins the independent set.

   a) What is the expected size of the independent set chosen in this manner? Recall from your extensive training in probability that for a set of random variables x_i, each of which is either zero or one, the expected value $E[\sum x_i] = \sum Pr[x_i]$, even if the x_i are not independent random variables (note that the word "independent" is being used in two different senses in this problem; independent random variables are a very different concept from independent sets).

   b) Could the expected size of the set be increased by expanding the set of colors to $\{0, 1, 2\}$? What is the best choice of the number of colors k?

   c) What is the work and depth of this randomized algorithm for choosing an independent set in a linked list?

2. Do problem 3.4 (using the $O(\lg^* n)$ coloring algorithm for list ranking). If instead of the $O(\lg *n)$ algorithm we use the algorithm from Problem 1, what is the expected work and depth of the list-ranking algorithm?

3. Do problem 3.15 (in-order traversal numbers).

4. Do problem 3.33 (choosing a root).