CS325: Analysis of Algorithms, Fall 2016

Practice Assignment 1*

Due: Tue, 10/4/16

Homework Policy:

1. Students should work on practice assignments individually. Each student submits to TEACH one set of *typeset* solutions, and hands in a printed hard copy in class or slides it under my door before the midnight of the due day. The hard copy will be graded.
2. Practice assignments will be graded on effort alone and will not be returned. Solutions will be posted.
3. The goal of the assignments is for you to learn solving algorithmic problems. So, I recommend spending sufficient time thinking about problems individually before discussing them with your friends.
4. You are allowed to discuss the problems with others, and you are allowed to use other resources, but you *must* cite them. Also, you *must* write everything in your own words, copying verbatim is plagiarism.
5. More items might be added to this list. ☺

Problem 1. For each of the following, indicate whether $f = O(g)$, $f = \Omega(g)$ or $f = \Theta(g)$.

(a) $f(n) = 2n - 5$, $g(n) = 1235813n + 2016$.

(b) $f(n) = n \log n$, $g(n) = 1235813n + 2016$.

(c) $f(n) = n^{2/3}$, $g(n) = 7n^{3/4}$.

(d) $f(n) = n^{1.00000001}$, $g(n) = n \log n$.

(e) $f(n) = n5^n$, $g(n) = 7^n$.

Problem 2. Let $f(n)$ and $g(n)$ be nonnegative functions. Use the definition of Θ-notation to prove that $\max(f(n), g(n)) = \Theta(f(n) + g(n))$.

Problem 3. Suppose you are given a stack of n pancakes of different sizes. You want to sort the pancakes so that smaller pancakes are on top of larger pancakes. The only operation you can perform is a flip – insert a spatula under the top k pancakes, for some integer k between 1 and n, and flip them all over.

*Some problems are from Jeff Erickson's lecture notes. Looking into similar problems from his lecture notes on recursion is recommended. Other problems are from the book "Introduction to algorithms" by Coremen, Leiserson, Rivest, and Stein. Fore more on asymptotic running time analysis, see Chapter 2 of this book.

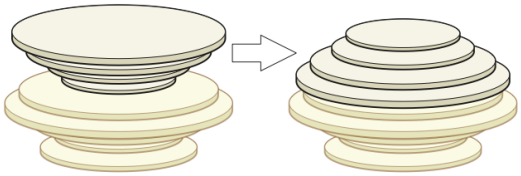

Figure 1: Flipping the top four pancakes.

(a) Describe an algorithm to sort an arbitrary stack of n pancakes using as few flips as possible. Exactly how many flips does your algorithm perform in the worst case?

(b) Now suppose one side of each pancake is burned. Describe an algorithm to *sort* an arbitrary stack of n pancakes, so that the burned side of every pancake is facing down, using as few flips as possible. Exactly how many flips does your algorithm perform in the worst case?

Do *not* submit solutions for the following problems, they are just for practice.

Practice Problem A. Prove that $\log(n!) = \Theta(n \log n)$. (Logarithms are based 2)

Practice Problem B.

(a) Read tree traversal from wikipedia: `https://en.wikipedia.org/wiki/Tree_traversal`, the first section, Types.

(b) Prove that there is no algorithm to reconstruct an arbitrary binary tree from its preorder and postorder node sequences

(c) Recall that a binary tree is *full* if every non-leaf node has exactly two children. Describe and analyze a recursive algorithm to reconstruct an arbitrary full binary tree, given its preorder and postorder node sequences as input.

(d) Describe and analyze a recursive algorithm to reconstruct an arbitrary binary tree, given its preorder and inorder node sequences as input.

(e) Describe and analyze a recursive algorithm to reconstruct an arbitrary *binary search tree*, given only its preorder node sequence. Assume all input keys are distinct.

In parts (b), (c), and (d), assume that all keys are distinct and that the input is consistent with at least one binary tree.