CS314: Algorithms

Homework 0, due Friday, January 16 at the beginning of class

This homework tests your familiarity with the prerequisite material from Data Structures and Discrete Math, primarily to help you identify gaps in your knowledge. **You are responsible for filling those gaps on your own.**

Before you do anything else, read the Course Policies on the webpage. This web page gives instructions on how to write and submit homeworks—staple your solutions together in order, write your name on every page, don't turn in source code, analyze everything, use good English and good logic, and so forth.

Required Problems

1. RECURRENCES (20 points)

   Solve the following recurrences. State tight asymptotic bounds for each function in the form $\Theta(f(n))$ for some recognizable function $f(n)$. You do not need to turn in proofs (in fact, please *don't* turn in proofs), but you should do them anyway just for practice. Assume reasonable but nontrivial base cases if none are supplied. More exact solutions are better.

   (a) $A(n) = 2A(n/2) + \lg n$

   (b) $B(n) = 3B(n/2) + n$

   (c) $C(n) = 2C(n/2) + n^2$

   (d) $D(n) = 2D(n-1) + 1$

   (e) $E(n) = \max_{1 \le k \le n/2}(E(k) + E(n-k) + n)$

   (f) $F(n) = 2F(\lfloor n/3 \rfloor + 9) + n^2$

   (g) $G(n) = 2G(n-1)/G(n-2)$

   (h) $H(n) = \log(H(n-1)) + 1$

   (i) $I(n) = 2I(\sqrt(n)) + 1$

   (j) $J(n) = J(n/2) + 1$

2. Sorting functions (20 points)

Sort the following 25 functions from asymptotically smallest to asymptotically largest, indicating ties if there are any. You do not need to turn in proofs (in fact, please *don't* turn in proofs), but you should do them anyway just for practice.

1	n	n^2	$\lg n$	$1 + \lg\lg n$
$\cos n + 2$	$n^{\lg n}$	$(\lg n)!$	$(\lg n)^{\lg n}$	F_n
$\lg^{1000} n$	$2^{\lg n}$	$n \lg n$	$\sum_{i=1}^{n} i$	$\sum_{i=1}^{n} i^2$
$n!$	$\lg(n^{10000})$	$\lfloor \lg\lg(n) \rfloor$	$2^{2\log n}$	$15n^2 - 12n + 8\lg n + 4$

To simplify notation, write $f(n) \ll g(n)$ to mean $f(n) = o(g(n))$ and $f(n) \equiv g(n)$ to mean $f(n) = \Theta(g(n))$. For example, the functions n^2, n, $\binom{n}{2}$, n^3 could be sorted either as $n \ll n^2 \equiv \binom{n}{2} \ll n^3$ or as $n \ll \binom{n}{2} \equiv n^2 \ll n^3$. *[Hint: When considering two functions $f(\cdot)$ and $g(\cdot)$ it is sometime useful to consider the functions $\ln f(\cdot)$ and $\ln g(\cdot)$.]*

3. Trees, Fibonacci numbers, and Induction (20 points)

The n^{th} Fibonacci binary tree $\mathcal{F}_n$ is defined recursively as follows:

- $\mathcal{F}_1$ is a single root node with no children.
- For all $n \geq 2$, $\mathcal{F}_n$ is obtained from $\mathcal{F}_{n-1}$ by adding a right child to every leaf node and adding a left child to every node that has only one child.

(a) Prove that the number of leaves in $\mathcal{F}_n$ is precisely the n^{th} Fibonacci number: $F_0 = 0, F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for all $n \geq 2$.

(b) How many nodes does $\mathcal{F}_n$ have? For full credit, give an *exact* closed form answer in terms of the Fibonacci numbers and prove that your answer is correct.

(c) Prove that for $n \geq 2$, the left subtree of $\mathcal{F}_n$ is a copy of $\mathcal{F}_{n-2}$. (Hint: This is easier than it sounds!)

4. Fractions and Pigeonholes (20 points)

The *fractional part* of x is the amount by which x exceeds its floor, $\lfloor x \rfloor$. For $x \in \mathbb{R}$ and $n \in \mathbb{N}$, let $S = \{x, 2x, \ldots, (n-1)x\}$.

(a) Prove that if some pair of numbers in S have fractional parts that differ by at most $1/n$, then some number in S is within $1/n$ of an integer.

(b) Use part (a) to prove that some number in S is within $1/n$ of an integer.