CSE332 Data Abstractions, Spring 2012
Homework 1

Due: **Friday, April 6, 2012** at the beginning of class. Your work should be readable as well as correct.

This assignment has **six** problems.

Problem 1. Some important sums

A certain style of sums appear repeatedly in analyzing the running time of different algorithms. In this problem, you will compute two of these sums and prove a third to be true.

(a) Weiss 1.8a

(b) Weiss 1.8b

(c) Weiss 1.12a

(For problems 1.8a and 1.8b make sure to not only evaluate the sum, but also show how you performed this evaluation. All problems are the same in the 2nd and 3rd editions.)

Problem 2. Recurrence Relations

Consider the following recurrence relation: $T(1) = 5$, and for $n > 1$, $T(n) = 1 + 2T\lfloor n/2 \rfloor$
Note: $\lfloor n/2 \rfloor$ is the "floor" of $n/2$: it rounds down to the next integer.

(a) Give $T(n)$ for $n =$ integers 1 through 8.

(b) Expand the recurrence relation to get the closed form. Show your work; do not just show the final equation. For arithmetic simplicity, you may assume n is a sufficiently large power of 2 that the floor function does not lead to rounding.

Problem 3. Using a Time Budget

This problem gives an orthogonal view of comparative running times from that given in lecture. Be sure to look at the patterns in your table when you have completed it. For each function $f(n)$ and time t in the following table, determine the largest size n of a problem that can be solved in time t, assuming that the algorithm to solve the problem takes $f(n)$ **micro**seconds. For large entries (say, those that warrant scientific notation), an estimate is sufficient. For one of the rows, you will not be able to solve it analytically, and will need a calculator, spreadsheet, or small program.

$f(n)$	1 second	1 minute	1 hour	1 day	1 month	1 year
$1000 \log_2 n$						
$100n$						
$100n \log_2 n$						
$10n^2$						
n^3						
$\frac{1}{10}2^n$						

Problem 4. Fun with Induction

The following statement is clearly not true. Can you spot the error in the inductive "proof" below? Specify which of the following 5 numbered lines are wrong, and clearly describe the error.

All jelly beans are the same color

"Proof": The proof is by induction on n, the number of jelly beans:

Base case ($n = 1$):

(a) If there is only one jelly bean in the set, then the statement trivially holds.

Induction step: ($n = k + 1$). Assume the statement holds for $n = k$. Now suppose you have $k + 1$ jelly beans.

(b) Set the first one aside. The remaining k must be the same color (let's say red).

(c) All we have to do now is show that the first one is also red.

(d) To do this, remove a second jelly bean and put the first jelly bean back in to form a new set of size k. By the inductive hypothesis, all the jelly beans in the new set are also the same color.

(e) Since this set contains $k - 1$ jelly beans that we already know are red, it follows that they are all red (including the first).

Problem 5. Big-O, Big-Θ

For each of the following statements, use our formal definitions of Big-O, Big-Θ, and Big-Ω either to prove the statement is true or to explain why it is false.

(a) If we have an algorithm that runs in $O(n)$ time and make some changes that cause it to run 10 times slower for all inputs, it will still run in $O(n)$ time.

(b) If $f(n) = O(g(n))$ and $h(n) = O(k(n))$, then $f(n) - h(n) = O(g(n) - k(n))$.

(c) If $f(n) = O(g(n))$ and $h(n) = O(k(n))$, then $f(n) + h(n) = O(g(n) + k(n))$.

(d) $\left(2^{n+3}\right) = \Theta(2^n)$

(e) $(2^n)^{1/3} = \Theta(2^n)$

Problem 6. Algorithm analysis

(a) Weiss 2.7a (give the best big-O bound you can for each of the 6 program fragments; you do not need to explain why)

(b) Weiss 2.11

(All problems are the same in the 2nd and 3rd editions.)