MA/CSSE 473 – Design and Analysis of Algorithms

Homework 2 – (10 problems, 77 points total) Updated for Fall, 2014

These are to be turned in to a drop box on Moodle. You may write your solutions by hand and scan them if you wish.   When a problem is listed by number, it is from Levitin. 1.1.2 means "Exercise 2 from section 1.1" .

Timing suggestion: This is a very long assignment; HW 1 is not so long. I recommend that you do part of this assignment *before* the due date for HW1. Because some students will take a while to "get into" this course, I chose to make HW 1 shorter and HW2 longer. If you want to make the workload more even, schedule you time pretending like a few HW2 problems are really in HW1.

Problems for enlightenment/practice/review (not to turn in, but you should think about them):
How many of them you need to do serious work on depends on you and your background. I do not want to make everyone do one of them for the sake of the (possibly) few who need it. You can hopefully figure out which ones you need to do.

2.1.7 [2.1.7]   (and 2.1.8. Effect of changing problem size on runtime)
2.1.10a [2.1.10]   (chess-board doubling)
2.2.1 [2.2.1]   (efficiency of sequential search)
2.2.2 [2.2.2]   (informal definitions of asymptotic notations)
2.2.6 [2.2.6]   (orders of growth for polynomials and exponentials)
2.2.9 [2.2.9]   (effect of presorting on running time)
2.3.1 [2.3.1]   (summation practice)
2.3.5 [2.3.5]   (Secret algorithm)
2.3.6 [2.3.6]   (Enigma algorithm)
2.3.12 [2.3.11]   (von Neumann's Neighborhood)
2.3.13 [not in 2nd ed] (Page numbering). How many decimal digits altogether in the numbers 1-1000?
2.4.13[not in 2nd ed] (Frying Hamburgers)
2.5.3 [2.5.4]   (Climbing stairs)
Another good practice problem to prepare for this assignment: The "growable array" exercise from 230.
See the three files from days 01 and 02 in the 230-materials folder.

Problems to write up and turn in:

1. 2.1.4   (6 points)   (socks and gloves)

2. 2.1.5   (9 points)   (number of digits in the representation of a positive integer))
                                             Note that there are four parts of this problem. If you have the 2nd edition of
                                             Levitin, see the "Problems" document. Points: (3, 3, 1, 2)

3. 2.2.3   (10 points)   (big-theta of specific functions with proofs)
                                             For parts a&b, use limits;
                                             for e, use formal definitions of O and Ω;
                                               you should probably find specific values for the c and n_0 in the formulas on
                                               pages 53-55.
                                             for c & d, you can use the theorem on p 56.

4. 2.2.7a,d   (4 points)   (proof or disproof of properties using the formal definition)

5. 2.2.12 [2.2.10]   (6 points)   (door in a wall). Show that your algorithm is actually O(N).
6. 2.3.2   (8 points)   (big-theta for various summations).

7. 2.3.11 [2.3.10]   (10 points)   (GE Algorithm – yeah, it's a big secret what GE stands for ☺) Include a
                                             quantitative indication of how much time is gained by removing the glaring
                                             inefficiency.

8. 2.4.14 [5.3.10] (10 points)   (Celebrity identification) **be sure to start this one early!** "Efficient" in this case means making the number of questions for N people as small as you can, and you should say how many questions (as a function of N) are required by your approach in the worst case.

9. Master Theorem Proof (8 points).   These questions refer to the proof of the Master Theorem in Weiss section 7.5.3 (available on Moodle). **Later we will have homework problems that *use* the Master Theorem**.

This proof is sometimes part of CSSE 230, but it happens during the time of the big team project, so many students don't focus on it enough to really understand it. Here is your chance to do so! If you do not already feel comfortable with telescoping as a method of solving recurrence relations, read the early parts of the section carefully. If you do, you can start with the page before the proof.

You'll probably find one or more of the formulas form Levitin's Appendix A to be helpful here.

(a) (5 points) (7.11) Why is it $O(A^M)$? Why is the second equation true ($: O(A^M) = O(N^{\log_B A}).$)?
(b) (3 points) Sentence after (7.11). Why does the sum contain that many terms? Why does $A = B^k$ imply $A^M = N^k$?

10. Dasgupta questions (6 points, 2 for each part). Refers to an excerpt from Dasgupta's book (on Moodle)
    (a) What does Dasgupta say are the two main ideas that changed the world? Do you agree? What else might you include in the list?

    (b)Why is the simple algorithm at the bottom of page 4 actually not $O(n)$?
    (c) Show how to use al-Khwarizmi's technique to multiply 9 (first column) by 15 (second column).

    Note on problem 10. These are easy "did you read this?" problems. On later assignments there will be harder problems based on the Dasgupta excerpt.