MA/CSSE 473 – Design and Analysis of Algorithms

Homework 2 – 50 points

These are to be turned in to a drop box on ANGEL. You may write your solutions by hand and scan them if you wish. There is an easy-to-use network scanner in F-217. It will email the scan to you.

When a problem is listed by number, it is from the textbook (Levitin, 2nd edition).
1.1.2 means "problem 2 from section 1.1" .

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	(and 2.1.8. Effect of changing problem size on runtime)
2.1.10	(chess-board doubling)
2.2.1	(efficiency of sequential search)
2.2.2	(informal definitions of asymptotic notations)
2.2.6	(orders of growth for polynomials and exponentials)
2.2.9	(effect of presorting on running time)
2.3.1	(summation practice)
2.3.5	(Secret algorithm)
2.3.6	(Enigma algorithm)

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   (6 points)   (number of digits in the representation of a positive integer))
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 give specific values for the c and n_0 in the formulas on pages 53-54.
                         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. 2.2.10   (6 points)   (door in a wall). Show that you algorithm is O(N).
6. 2.3.2   (8 points)   (big-theta for various summations)
7. 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.