Homework 1

(Network Optimization by Randomization)

1. (10 pts.) Comment the following assertion due to Democritos (ca. 460 BC - ca. 370 BC) "The randomness is the unknown, and that the nature is determined in its fundamentals."

2. (30 pts.) Which of the following statements are true? Justify your answers.

   (a) $n \log_2 n = O\left(n \log_{10} n\right)$ (Note: 2 and 10 are the respective bases of the logarithms.)

   (b) $n \log_2 n = o\left(n \log_{10} n\right)$

   (c) $2^{n+1} = O\left(2^n\right)$

   (d) $2^n = o\left(2^{2n}\right)$

   (e) $n^n = o(n!)$ (Note: $n!$ denotes the factorial.)

3. (20 pts.) Let $T(n)$ denote the running time of an algorithm on an input of size n.

   (a) If $T(n) = 4T\left(\frac{n}{2}\right) + cn$ for all $n \geq 1$, and c is a constant, determine the asymptotic growth of $T(n)$ using the big-Oh notation.

   (b) Same as in (a), but for the recursion $T(n) = 3T\left(\frac{n}{2}\right) + cn$.

4. (20 pts.) Write an algorithm performing Karatsuba's multiplication for two binary numbers $a = a_1 a_2 \ldots a_n$ and $b = b_1 b_2 \ldots b_n$, where $n = 2^k$, $k \geq 0$, and $a_i, b_i \in \{0, 1\}$ for all $i = 1, 2, \ldots, n$.

5. (20 pts.) Consider a vector A of size n and the following sorting algorithm:

   stupidSort(A[1..n])
     while(A is not sorted)
       randomly permute the elements of A
     end
     return A

   Assuming that all the possible permutations can occur with equal probabilities, determine the running time of the algorithm using the big-Oh notation.

6. (Bonus: 25 pts.) During the lecture we sketched Karatsuba's algorithm for multiplying two numbers, each with $n = 2^k$ digits. Extend this algorithm to the multiplication of two square matrices with $n = 2^k$ rows/columns. Points are awarded if the running time $T(n)$ of the algorithm satisfies $T(n) = o\left(n^3\right)$.

Note: Homework due on April 28^{th}.