CSC 282 - Fall 2007 - http://www.cs.rochester.edu/~stefanko/Teaching/07CS282/
Homework #5 - due: **Tuesday, Oct. 9, before class**
(do all the problems, including the ones you solved in the quiz)

1. (4 points) Compute the convolution of $(1, 0, 1, 0, 1, 0, 1)$ with $(1, 1, 1)$.

2. (4 points) Compute the Fourier transform of $(1, 1, 1, 1)$ (for $\omega = i = \sqrt{-1}$).

3. (6 points) Find the unique polynomial p of degree 2 such that $p(-1) = 8$, $p(0) = 5$, and $p(1) = 18$. Write your answer in the coefficient representation. Show your work.

4. (4 points) Let p be a polynomial of degree 10. Let q be a polynomial of degree 7. Suppose we want to compute the product $r = pq$ using the Fourier transform. That is, we will evaluate both p, q on 2^n points, obtain evaluation of r on these points, and then recover the coefficient representation of r using the inverse Fourier transform. What is the smallest value of n we can use? Argue why smaller value of n does not work.

5. (4 points) The convolution of the sequence $(1, 2, 3, 4)$ with an unknown sequence S is the sequence $(5, 11, 17, 23, 4)$. Compute the sequence S.

6. (6 points) Let $m \geq n$. You are given two sequences $A = (a_1, \ldots, a_n)$ and $B = (b_1, \ldots, b_m)$. Assume that there exists a sequence S such that the convolution of A with S is B. Give an $O(m \log m)$ algorithm to find S. You can assume that operations on complex numbers can be performed in time $O(1)$.

7. (4 points) Let A be an $n \times n$ matrix with m non-zero entries. Let v be a vector of length n. The matrix A is given as a list of m triples (j_i, k_i, b_i), for $i = 1, ..., m$ (the i-th triple encodes the information that the entry in row j_i, column k_i of A is b_i). Write a pseudocode for an algorithm which computes Av in time $O(m + n)$.

8. (6 points) Suppose that there exists an algorithm which squares n-bit numbers in time $O(T(n))$. Use this algorithm to give an algorithm which multiplies n-bit numbers in time $O(T(n))$.

9. (4 points) Suppose that there exists an algorithm which cubes n-bit numbers in time $O(T(n))$ Use this algorithm to give an algorithm which squares n-bit numbers in time $O(T(n))$.

10. (6 points) Prove that the solution of recurrence $T(n) = 2T(n/2) + 1$ is $O(n)$.

11. (6 points) Prove that the solution of recurrence $T(n) = T(n/2) + T(n/4) + T(n/8) + n$ is $O(n)$.

12. (5 points) Write the pseudocode of the Karatsuba-Offman integer multiplication algorithm (this is the $O(n^{\log_2 3})$ algorithm described in the book and covered in class).

13. (4 points) Solve the following recurrence: $T(n) = T(\sqrt{n}) + O(1)$. Prove your answer.

14. (6 points) Solve the following recurrence: $T(n) = T(n/2) + T(n/3) + T(n/4) + O(n)$. Prove your answer.

15. (4 points) State the recurrence for the running time of the merge-sort algorithm. Do not solve the recurrence.

16. (6 points) We are given an array of integers $A[1..n]$. We would like to determine whether there exists an integer x which occurs in A more than $n/5$ times. Give an algorithm which runs in time $O(n)$. You can use known linear-time algorithms as subroutines. Argue that your algorithm is correct and runs in linear time. Important: no argument - no points.

17. (8 points) We are given n positive numbers $a_1, \ldots, a_n$ (the numbers are not necessarily integers (and hence you cannot use counting sort, radix sort, etc.)). We are given another positive number B. We want to find the size of the largest subset of $a_1, \ldots, a_n$ whose average is at least B. Give an $O(n)$ algorithm for the problem. For example if the numbers are $4, 1, 2, 4$ and $B = 3$ then we can select $4, 1, 4$ (their average is $(4 + 1 + 4)/3 = 3 \geq 3$) but we cannot select $4, 1, 2, 4$ (their average is $(4 + 1 + 2 + 4)/4 = 11/4 < 3$). (Thus the answer for this example is 3.)