Problem Set 3

Due dates: Electronic submission of .tex and .pdf files of this homework is due on **2/7/2014 before 11:00am** on csnet.cs.tamu.edu, a signed paper copy of the pdf file is due on **2/7/2014** at the beginning of class.

Name: (put your name here)

Resources. (All people, books, articles, web pages, etc. that have been consulted when producing your answers to this homework)

On my honor, as an Aggie, I have neither given nor received any unauthorized aid on any portion of the academic work included in this assignment. Furthermore, I have disclosed all resources (people, books, web sites, etc.) that have been used to prepare this homework.

Signature: _____

Make sure that you describe all solutions in your own words, even though many of the exercises were part of team explorations!

Read chapters 4 in our textbook.

Problem 1. (15 points) Exercise 4.3-5 on page 87 in our textbook.

Problem 2. (10 points) Suppose that you want to improve upon Karatsuba's integer multiplication algorithm using a similar recursive method. Your idea is to split the n-bit input integers each into three parts instead of two. (a) What is the maximum number of multiplications of $n/3$ bit integers that you can afford in order to get a divide-and-conquer algorithm for integer multiplication that outperforms Karatsuba's method. (b) What divide-and-conquer algorithms using $n/3$-bit integers can you find in the literature for multiplication of n-bit integers. How many multiplications of $n/3$ bit integers are used there?

Problem 3. (a) (10 points) Give pseudo-code for Strassen's matrix multiplication (use the version from the slides). What is the idea behind the algorithm?

(b) (20 points) Prove that Strassen's algorithm is correct. Use induction. Key points: Make sure the correct result is established for each of the four quadrants. Use the algorithm given on the slides, not the one given in our textbook.

Solution.

Read Chapter 30.

Problem 4. (a) (10 points) Suppose that you are given a polynomial

$$A(x) = \sum_{k=0}^{n-1} a_k x^k.$$

The input to the FFT of length n is given by an array containing the coefficients $(a_0, \ldots, a_{n-1})$. Describe the output of the FFT in terms of the polynomial $A(x)$.

(b) (10 points) Let ω be a primitive nth root of unity. The fast Fourier transform implements the multiplication with the matrix

$$F = (\omega^{ij})_{i,j \ in[0..n-1]}.$$

Show that the inverse of the F is given by

$$F^{-1} = \frac{1}{n}(\omega^{-jk})_{j,k \ in[0..n-1]}$$

[Hint: $x^n - 1 = (x-1)(x^{n-1} + \cdots + x + 1)$, so any power $\omega^\ell \neq 1$ must be a root of $x^{n-1} + \cdots + x + 1$.] Thus, the inverse FFT, called IFFT, is nothing but the FFT using ω^{-1} instead of ω, and multiplying the result with $1/n$.

(c) (10 points) Describe how to do a polynomial multiplication using the FFT and IFFT for polynomials $A(x)$ and $B(x)$ of degree $\leq n-1$. Make sure that you describe the length of the FFT and IFFT needed for this task.

(d) (15 points) How can you modify the polynomial multiplication algorithm based on FFT and IFFT to do multiplication of long integers in base 10? Make sure that you take care of carries in a proper way.

Solution.

Discussions on ecampus are always encouraged, especially to clarify concepts that were introduced in the lecture. However, discussions of homework problems on piazza should not contain spoilers. It is okay to ask for clarifications concerning homework questions if needed.

Checklist:

□ Did you add your name?

□ Did you disclose all resources that you have used?
(This includes all people, books, websites, etc. that you have consulted)

□ Did you sign that you followed the Aggie honor code?

□ Did you solve all problems?

□ Did you submit (a) the pdf file derived from your latex source file of your homework?

□ Did you submit (b) a hardcopy of the pdf file in class?