Homework 1*

Algorithms
Spring 2020 CS207@IITG

(1) Solve the following problems from [CLRS]: 4.5-1 (page 96), 4.5-4 (page 97).

(2) Write the formal proof of correctness of algorithm discussed for counting inversions in a given permutation.

(3) Is there another non-trivial way to organize submatrix multiplication/additions/subtractions in Stressens algorithm, while achieving $o(n^3)$ time for the multiplying two matrices of order $n \times n$? How?

(4) Determine which could get affect in the algorithm for closest pair of points presented in class if either or all of the three assumptions are removed: no two points have the same x-coordinate, no two points have the same y-coordinate, and the pairwise distances' are distinct.

Adjust the algorithm accordingly and formally prove the correctness of the suggested algorithm.

(5) In computing $A(x)$ at twiddle factors $\omega_{0,2n}, \omega_{1,2n}, \ldots, \omega_{2n-1,2n}$, we used $A_{\omega_{j,2n}} = A_{even}(\omega_{j,2n}^2) + \omega_{j,2n} A_{odd}(\omega_{j,2n}^2)$ for $j = 0, \ldots, n-1$, and $A_{\omega_{j+n,2n}} = A_{even}(\omega_{j,2n}^2) + \omega_{j+n,2n} A_{odd}(\omega_{j,2n}^2)$ for $j = 0, \ldots, n-1$. In specific, we had shown in class that computing both the $A_{even}(\omega_{j,2n}^2)$ and $A_{odd}(\omega_{j,2n}^2)$ at $j = 0, \ldots, n-1$ suffice to compute $A(x)$ at $2n$ twiddle factors $\omega_{0,2n}, \omega_{1,2n}, \ldots, \omega_{2n-1,2n}$.

Analogously, with the appropriate twiddle factors, prove that $A_{even}(x^2)$ (resp. $A_{odd}(x^2)$) can be evaluated at n points in $T(\frac{n}{2})$ time using $\frac{n}{2}$ points computed at each of its children.
(Note that $T(n)$ denotes the number of operations required to evaluate $A(x)$ of degree $n-1$ at $2n$ points.)

(6) Devise a $O(n \lg n)$ time algorithm for computing the inverse fast Fourier transform of Y vector in $VD = Y$. Here, V is the Vandemonde matrix of order $2n \times 2n$ with $(j,k)^{th}$ entry of V equals to $\omega_{j,2n}^k$, D is a matrix of order $2n \times 1$, and Y is a matrix comprising of complex values. (Note that D comprises of the set of variables to be determined.)

(7) Let x, y be a positive integers. Also, let $y0$ be y in binary concatenated with bit value 0, and let $y1$ be y in binary concatenated with bit value 1. Then prove that $x^{y0} = (x^y)^2$ and $x^{y1} = (x^y)^2 x$.

Assuming RAM model of computation, write a recursive algorithm that uses this property of exponentiation; further, analyze the time complexity of your algorithm.

(8) Prove the correctness of the greedy algorithm with the stays ahead argument: n jobs are to be scheduled; no preemption involved; the objective is to maximize the number of mutually compatible jobs that can be scheduled on a single machine.

*Prepared by R. Inkulu, Department of Computer Science, IIT Guwahati, India. http://www.iitg.ac.in/rinkulu/