CS 530 - MIDTERM - SPRING 2013

DIRECTIONS: Do any three of the following four problems. Each problem is worth 8 points. Write all of your answers in your blue book. The test is open notes.

Please write the number of the problem you are not doing on your blue book.

1. In this problem we compare the work needed to multiply two matrices, to multiply two lower triangular matrices and to find the inverse of a lower triangular matrix.

   Let M(n) be the time to multiply together 2 n x n matrices.
   Let MT(n) be the time to multiply together 2 n x n lower triangular matrices.
   Let INVT(n) be the time to compute the inverse of an n x n lower triangular matrices.

   You may use reasonable assumptions about M(n), MT(n), and INVT(n) so long as you state them. For example, you can assume that all three of these functions are in $\Omega(n^2)$.

   i. Show that M(n) is in O(MT(n)). That is, if you can multiply 2 lower triangular matrices in time MT(n) then you can multiply any two n x n matrices in time O(MT(n)).

   ii. Show that INVT(n) is in O(M(n)). State clearly the logic of what you are doing here in order to show this.

   Note: This one is harder. You can use divide and conquer together with the fact that,

   For a 2 by 2 lower triangular matrix $B =$

   a 0
   c d

   the inverse, B^{-1}, can be written as

   a^{-1} 0
   -d^{-1}ca^{-1} d^{-1}

   Here a,c and d may be numbers but they also may be (equal sized) square arrays of numbers.

2. i. Compute the convolution C of the two vectors A and B, where
   A = (6 5 4 3 2 1) and B = (1 1 0 0 1 0).

   ii. What is the complexity of computing the convolution of two n x 1 vectors directly from the definition ?

   That is, how many arithmetic operations does it take to compute the convolution of two vectors ?

   iii. Let w be the principal 10^{th} root of unity. So the 10 tenth roots of unity are $w^0, w^1, w^2, ..., w^9$.

   Which of these 10^{th} roots of unity is -$(w^8)^2$?

   iv. List which of these 10^{th} roots of unity are principle (that is, can be used to generate all of the others by taking their powers) ? w is on this list, are there others as well ?

3. A T-matrix (or Toeplitz matrix) is an n x n matrix A = (a_{ij}) such that $(a_{ij}) = (a_{i-1,j-1})$, for i = 2, 3,...n and j = 2, 3,...n.

   An example of a 4 x 4 T-matrix of integers where the first column is 3,4,5,6 is on the blackboard.

   (a) Describe how to represent an n x n T-matrix so that two such Toeplitz matrices can be added in O(n) time. Briefly describe the add algorithm for these matrices.

   (b). Show how to compute the convolution of 2 n x 1 vectors V and W by matrix multiplication. Use a T-matrix containing V as part of an extended matrix L (See the "Idea" below.) and multiply L by the vector W to obtain the convolution.

   What is the extended matrix L for the T-matrix I've drawn on the blackboard? How many arithmetic operations does this take convolution computation take?

   Idea to obtain the extended T-matrix L:
   Start with an n x n T-matrix where the first column consists of the vector V and the first row is all 0's except in the first column. Now extend this T-matrix to a (2n-1) x n matrix by making every element of V appear on its diagonal all the way across to the last column. Then use this extended matrix and multiply it by W to obtain the convolution. regular matrix multiplication to calculate the convolution of two length n vectors V and W.

4. (A Treasure Hunt)
   You are given weighted grid graph G (below), that is a an n by m grid where the vertices are the grid points and the edges go horizontally, vertically or diagonally downward between neighboring points. The weights are natural numbers. So for example,

   G:

   Think of this graph as representing a small part of a city where the edges are roads. The weight on the edges represent the number of places on that road segment where treasure (A gold coin) can be found. The starting point of the treasure hunt is the upper left corner and the end point is the lower right corner of the grid. Each move along the graph is either to the right or down, or along a diagonal when one is present in the graph.

   (i). Give an efficient algorithm to find largest number of coins that can be collected while going from start to end and the path which gives this largest amount of coins. And find the algorithm's complexity.

   (ii). Carry out your algorithm on the graph G above and also find a best path.