Individual Assignment 1

Each problem is worth 10 points.

1. Remember the stock market problem we talked about in class. We have an array A of numbers, with A indexed from 0 to n-1. We want to find two integers t_1 and t_2 that satisfy the following conditions:

$0 \leq t_1 \leq t_2 \leq n - 1$

$A[\ t_2\ ] - A[\ t_1\ ]$ is as large as possible.

Write in pseudocode an algorithm for finding t_1, t_2, and the value $A[\ t_2\ ] - A[\ t_1\ ]$ that will operate in linear time instead of n*lg(n) as we did in class. Hint: Think about other simple problems where you had to find a maximum number.

2. Prove that your solution to problem 1 is in $\Theta(n)$. Since the algorithm is not recursive, it is not necessary to use a recurrence equation. You should be able to explain why the algorithm takes a time proportional to n directly from the structure of the pseudocode.

3. Let A be the matrix $\begin{vmatrix} 1 & 2 \\ 5 & 6 \end{vmatrix}$ and B be the matrix $\begin{vmatrix} 3 & 4 \\ 7 & 8 \end{vmatrix}$. Use Strasson's algorithm to compute the matrix product AB. Show your work and verify that the answer is correct.

4. Show by the substitution method that the function T satisfying the recurrence equation

$$T(n) = T(n/2) + 1$$

is $\Theta(\lg(n))$. By "show" I mean prove. Don't worry about n or n/2 not always being an integer.

5. Do Exercise 4.5-1. That is, use the Master Method to give tight asymptotic bounds to the following recurrences:

a. $T(n) = 2T(n/4) + 1$.

b. $T(n) = 2T(n/4) + \sqrt{n}$.

c. $T(n) = 2T(n/4) + n$.

d. $T(n) = 2T(n/4) + n^2$.

For each recurrence, explain how you used the Master Method to get the bound.