

# Assignment 3

- 1 Give asymptotic upper and lower bounds for  $T(n)$  in each recurrence. Assume  $T(n)$  is constant for  $n \leq 2$ . Make bounds as tight as possible

a.  $T(n) = 16T(n/4) + n^2$

b.  $T(n) = 7T(n/3) + n^2$

c.  $T(n) = 7T(n/2) + n^2$

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

- 1.1 Asymptotic Upper Bound

- 1.2 Asymptotic Lower Bound

- 2 Prove the substitution method that the solution to the following recurrence is  $S(n) = O(n \lg^2(n))$

$$S(n) = 2S(n/2) + n \lg n$$

$$S(1) = 1$$

- 2.1 Substitution Method
- 3 Write the pseudocode for Strassen's Algorithm.
- 3.1 Strassen's Algorithm
- 4 Draw the recursion tree and show a good asymptotic upper bound on the recurrence  $T(n) = T(n/2) + T(n/3) + T(n/6) + n$ .
- 4.1 Recursion Tree
- 4.2 Asymptotic Upper Bound
- 5 Write the pseudocode for an algorithm that finds a local minimum of an array in  $O(\lg n)$  time, and explain the algorithm's run time.
- 5.1 Local Minimum