Design and Analysis of Algorithms
CS575, Spring 2024

Theory Assignment 2.1

Due on 2/27/2024 (Tuesday)

1. (16 points) Use the iteration method or recursion tree method to solve the following recurrence equation.
   a) (8 points)

$$T(n) = \begin{cases} T(n-1) + n & if\ (n > 1) \\ 1 & if\ (n = 1) \end{cases}$$

   b) (8 points) You can assume $n^{1/2^k} = 2$ for some integers *n* and *k*.

$$T(n) = \begin{cases} 0 & if\ n = 2 \\ T(\sqrt{n}) + 1 & if\ n > 2 \end{cases}$$

2. (6 points) Use Master method to solve $T(n) = 4T(n/2) + n^2$ and T(1)=1.

3. (10 points) Professor Caear wishes to develop a matrix-multiplication algorithm that is asymptotically faster than Strassen's Algorithm. His algorithm will use divide-and conquer method, dividing each matrix into pieces of size n/4× n/4, and the divide and combine steps together will take $\theta(n^2)$ time. He needs to determine how many subproblems his algorithm has to create in order to beat Strassen's algorithm. If his algorithm creates a subproblems, then the recurrence for the running time T(n) becomes T(n) = aT(n/4)+ $\theta(n^2)$. What is the largest integer value of a for which Professor Caesar's algorithm would be asymptotically faster than Strassen's algorithm?