Tutorial 1: CS21003 Algorithms 1

Prof. Partha Pratim Chakrabarti and Palash Dey
Indian Institute of Technology, Kharagpur

January 14, 20201

1. Arrange the following functions

$$n, 10, n^{\ln n}, \ln\ln^2 n, (\ln n)^{\ln n}, 2^{\sqrt{n}}, n!, n\ln^{1000} n, n^{1.0001}, n^{.999}\ln^{100000} n, \sqrt{n}^{\sqrt{n}}, \frac{1}{n}, \binom{n}{\frac{n}{2}}$$

in a sequence such that if $f_1(n)$ appears in the left of $f_2(n)$, then $f_1(n) = \mathcal{O}(f_2(n))$.

2. This question has two parts.
   (a) Given two sequences of positive integers, S and T, each of whose elements are in ascending (non-decreasing) order, develop a recursive algorithm to put them into a single sequence of elements which are in non-decreasing order. Present the working on an example. Prove the correctness of your algorithm. Analyze the time complexity of your algorithm using a recurrence equation. Is it optimal?
   (b) Generalizing the first part above, given n such sequences of integers, each in non-decreasing order, find the best algorithm to combine them into a single sequence in non-decreasing order. Analyze your method in details to establish the goodness of your proposed algorithm.

3. Compute asymptotic complexity of $T(n, n)$ in terms of Θ where

$$T(x, y) = \begin{cases} x & \text{if } y \leqslant 50 \\ y & \text{if } x \leqslant 50 \\ x + y + T\left(\frac{2x}{3}, \frac{y}{2}\right) & \text{otherwise} \end{cases}$$

4. Let there be n coins of value denominations $C = \{c_1, c_2, \ldots . c_n\}$. Given a required value V, you are to find the minimum number of coins from C whose total value exactly equals V. Write a recursive definition for the problem. Prove the correctness of your algorithm. Check whether identical sub-problems are generated. Analyze the time complexity based on recurrence equations. Develop an appropriate final algorithm with any required data structures to solve the problem efficiently. Show the working on the example where: $C = \{5, 8, 2, 1, 6, 12\}, V = 11$.

5. Compute asymptotic complexity of $T(n)$ in terms of Θ where
   (a) $T(n) = \begin{cases} 3T(\lceil n/2 \rceil) + n\log_2 n & \text{if } n > 50 \\ 1 & \text{otherwise} \end{cases}$
   (b) $T(n) = \begin{cases} T(\lceil n/5 \rceil) + 9n & \text{if } n \geqslant 50 \\ 1 & \text{otherwise} \end{cases}$
   (c) $T(n) = \begin{cases} T\left(\lceil \frac{n}{2} \rceil + 7\right) + \left(\lfloor \frac{n}{2} \rfloor + 11\right) & \text{if } n \geqslant 50 \\ 1 & \text{otherwise} \end{cases}$
   (d) $T(n) = \begin{cases} T(\lceil \sqrt{n} \rceil) + 13\lg n & \text{if } n \geqslant 50 \\ 1 & \text{otherwise} \end{cases}$

6. Give an example of two functions $f, g : \mathbb{N} \longrightarrow \mathbb{N} \setminus \{0\}$ such that we have neither $f = \mathcal{O}(g)$ nor $f = \Omega(g)$.