University of Houston

COSC 3320: Algorithms and Data Structures
Spring 2017

Homework 2

Due February 9, at the start of class

1. (a) Prove that the function $f(n) = na^{\log n}$, where a is a constant greater than 1, is $\Theta(n^c)$ for some constant c.

   (b) Prove that $f(n) = n/\log n$ is $o(n^a)$ for any constant $a \geq 1$.

   (c) Order the following functions by order of growth, that is, find an arrangement $f_1, f_2, \ldots, f_{20}$ of the functions such that $f_1 = O(f_2), f_2 = O(f_3), \ldots, f_{19} = O(f_{20})$. (Recall that $\log n$ means $\log_2 n$.)

$\dfrac{\log n}{n}$	$\dfrac{1}{\log n}$	$n^{4/5}$	1.5^n	$2^{\log n}$
$n \log\log n$	$\sqrt{\log n}$	$n^{\log_2 3}$	5	$n^{1/\log n}$
$\log n!$	$\log^{11/6} n$	$\dfrac{n}{\log n}$	$\log\log n^2$	$\sqrt{n^5}$
$2^{\sqrt{\log n}}$	$e^{\sqrt{n}}$	$\log\left(\dfrac{n}{\log n}\right)$	n^2	$n!$

2. Given an array A of n numbers, you are asked to output a new array B where $B[j]$ is equal to the multiplication of all the elements of A except $A[j]$. That is, for any $j \in \{1, 2, \ldots, n\}$,

$$B[j] = \prod_{\substack{i \in \{1,2,\ldots,n\} \\ \text{and } i \neq j}} A[i].$$

   (a) Solve this problem in linear time.

   (b) Solve this problem in linear time and without using the division operator. (Hint: it might be helpful to use a couple of additional arrays.)

3. Design recursive algorithms for the following problems:

   (a) Compute the n-th Fibonacci number F_n. Recall that the n-th Fibonacci number is defined as follows.

$$F_n = \begin{cases} 1 & \text{if } n = 0, 1, \\ F_{n-1} + F_{n-2} & \text{if } n \geq 2. \end{cases}$$

   (b) Compute the n-th power of a number x, x^n, with n non-negative integer. The algorithm should be designed in such a way that it is possible to write a recurrence relation for the total number of multiplications executed by the algorithm for which the Master Theorem applies. Write such a recurrence and apply the Master Theorem to obtain an asymptotic bound for it.

4. When possible, apply the Master Theorem to give asymptotic bounds for $T(n)$ for the following recurrences:

(a)

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ 3T(n/3) + n/2 & \text{if } n > 1. \end{cases}$$

(b)

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ \log n \cdot T(n/2) + n^2 & \text{if } n > 1. \end{cases}$$

(c)

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ T(n/2 + 2) + n^2 & \text{if } n > 1. \end{cases}$$

(d)

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ 4T(n/2) + n/\log n & \text{if } n > 1. \end{cases}$$

(e)

$$T(n) = \begin{cases} 4 & \text{if } n = 1, \\ 4T(n/2) + 16n^{15/7} & \text{if } n > 1. \end{cases}$$

(f)

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ 2T(n/2) + n/\log n & \text{if } n > 1. \end{cases}$$