Assignment - 2

Name: Inchara Raveendra

SCU ID: 00001653600

3.1-2
Show that for any real constants a and b, where $b > 0$,

$(n + a)^b = \Theta(n^b)$.

Let $c = 2^b$ and $n_0 \geq 2a$

For all $n \geq n_0$, we have $(n+a)^b \leq (2n)^b = cn^b$

Let $n_0 \geq \dfrac{-a}{1 - \frac{1}{2}^{1/b}}$ and $c = 1/2$

then, $n \geq n_0 \geq \dfrac{-a}{1 - \frac{1}{2}^{1/b}}$

$\Rightarrow \quad n - \dfrac{n}{2^{1/b}} \geq -a$

$n + a \geq (1/2)^{a/b} n$ and $(n + a)^b \geq cn^b$

$$\left[(1/2)^{a/b} n \right]^b \geq cn^b$$

$$(1/2)^a n^b \geq cn^b$$

$$\therefore (n + a)^b = \Theta(n^b)$$

3.1-4
Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$?

Is $2^{n+1} = O(2^n)$?

$$2^{n+1} = 2^n * 2$$

For $n \geq 1$ and any constant $c \geq 2$,

$$0 \leq 2^{n+1} \leq c \cdot 2^n$$

$$\therefore \quad 2^{n+1} = O(2^n)$$

Is $2^{2n} = O(2^n)$?

$$2^{2n} = 2^n * 2^n$$

There would exist n_0 and c such that

$$n \geq n_0 \Rightarrow 2^n \cdot 2^n = 2^{2n} \leq c^{2n}$$

So, $2^n \leq c$ for $n \geq n_0$ which is not possible

3.2-3
Prove equation (3.19).

Equation 3.19, $\log(n!) = \Theta(n \log n)$

To prove 3.19, apply Stirling's approximation

For large values of n, $\Theta(1/n)$ will be very small compared to 1

$$\therefore \log(n!) \approx \log\left(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n\right)$$

$$= \log\sqrt{2\pi n} + \log\left(\frac{n}{e}\right)^n$$

$$= \log\sqrt{2\pi} + \log\sqrt{n} + n\log\left(\frac{n}{e}\right)$$

$$= \log\sqrt{2\pi} + \frac{1}{2}\log n + n\log n - n\log e$$

$$= \Theta(1) + \Theta(\log n) + \Theta(n\log n) - \Theta(n)$$

$$= \Theta(n\log n)$$

3-3 *Ordering by asymptotic growth rates*

a. Rank the following functions by order of growth; that is, find an arrangement $g_1, g_2, \ldots, g_{30}$ of the functions satisfying $g_1 = \Omega(g_2)$, $g_2 = \Omega(g_3)$, ..., $g_{29} = \Omega(g_{30})$. Partition your list into equivalence classes such that functions $f(n)$ and $g(n)$ are in the same class if and only if $f(n) = \Theta(g(n))$.

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	n^2	$n!$	$(\lg n)!$
$(\frac{3}{2})^n$	n^3	$\lg^2 n$	$\lg(n!)$	2^{2^n}	$n^{1/\lg n}$
$\ln\ln n$	$\lg^* n$	$n \cdot 2^n$	$n^{\lg\lg n}$	$\ln n$	1
$2^{\lg n}$	$(\lg n)^{\lg n}$	e^n	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$\lg^*(\lg n)$	$2^{\sqrt{2\lg n}}$	n	2^n	$n\lg n$	$2^{2^{n+1}}$

a) $2^{2^{n+1}}$

2^{2^n}

$(n+1)!$

$n!$

e^n

$n \cdot 2^n$

2^n

$(3/2)^n$

$n^{\lg \lg n} = (2^{\lg n})^{\lg \lg n} = (2^{\lg \lg n})^{\lg n} = (\lg n)^{\lg n}$

$(\lg n)!$

n^3

$4^{\lg n} = 2^{2 \lg n} = 2^{\lg n^2} = n^2$

$\lg(n!) \simeq n \lg n$

$2^{\lg n} \simeq n$

$(\sqrt{2})^{\lg n} \simeq 2^{\frac{1}{2} \lg n} = n^{1/2} = \sqrt{n}$

$2^{\sqrt{2 \lg n}}$

$\lg^2 n$

$\ln n$

$\sqrt{\lg n}$

$\ln \ln n$

$n^{1/\lg n} = n^{\lg 2 / \lg n} = n^{\log_n 2} = 2$

b. Give an example of a single nonnegative function $f(n)$ such that for all functions $g_i(n)$ in part (a), $f(n)$ is neither $O(g_i(n))$ nor $\Omega(g_i(n))$.

b) $$f(n) = \begin{cases} 2^{2^{n+2}} & \text{if } n \text{ is even} \\ 1/n & \text{if } n \text{ is odd} \end{cases}$$

$2^{2^{n+2}}$ is asymptotically larger than all the above functions and $1/n$ is asymptotically smaller than all of the above functions. Hence, it is neither $O(g_i(n))$ nor $\Omega(g_i(n))$

4.2-4

What is the largest k such that if you can multiply 3×3 matrices using k multiplications (not assuming commutativity of multiplication), then you can multiply $n \times n$ matrices in time $o(n^{\lg 7})$? What would the running time of this algorithm be?

From Strassen's algorithm, for a sub-problem size $n/3$ and 'k' matrix multiplications in each recursive step, we can write,

$$T(n) = kT(n/3) + \Theta(n^2)$$

Using case 1 of Master's theorem,

$$T(n) = \Theta(n^{\log_3 k})$$

For $T(n)$ to be $O(n^{\lg 7})$, $n^{\log_3 k} < n^{\lg 7}$

$$\log_3 k < \lg 7$$

$$k < 3^{\lg 7} \simeq 21.85$$

$\therefore$ Largest possible value of k is 21

Running time of this algorithm would be

$T(n) = \Theta(n^{\log_3 21}) = \Theta(n^{2.77})$