

CSC505 HW1

Liam Adams

January 27 2019

Problem 1

The sorted order where $g_i \in o(g_{i+1})$ or $g_i \in \Theta(g_{i+1})$ is

$$lg^2n, n^{.0001}, n^2, n^{lgn}, 2^n, (n-1)!, n!, n^n$$

I will prove each relationship from left to right. The transitivity of little-o and big-theta allow me to only prove the adjacent relationships.

$$\lim_{n \rightarrow \infty} \frac{\frac{d}{dn} n^{.0001}}{\frac{d}{dn} lg^2n} = \frac{.0001n^{.0001}ln2}{2lgn} \text{ and } \frac{\frac{d}{dn} .0001n^{.0001}ln2}{\frac{d}{dn} 2lgn} = \frac{.0001^2 n^{.0001} ln^2 2}{2} = \infty$$

Therefore $lg^2n = o(n^{.0001})$

$$\lim_{n \rightarrow \infty} \frac{lg^2n}{n^2} = \frac{\frac{d}{dn} (lgn \times lgn)}{\frac{d}{dn} n^2} = \frac{lgn}{n^2 ln 2} \text{ and } \frac{\frac{d}{dn} lgn}{\frac{d}{dn} n^2 ln 2} = \frac{1}{n^2 ln^2 2} = 0, \text{ therefore } lg^2n = o(n^2)$$

$$\lim_{n \rightarrow \infty} \frac{n^2}{n^{lgn}} = 0 \text{ for } n > 4, \text{ therefore } n^2 = o(n^{lgn})$$

$\lim_{n \rightarrow \infty} \frac{n^{lgn}}{2^n} = 0$ for all $n > 0$, the power of the denominator is polynomially greater than the power of the numerator, so $n^{lgn} = o(2^n)$

For any positive constant $c > 0$ there exists a constant $n_0 > 0$ such that $0 \leq 2^n < c(n-1)!$ for all $n \geq n_0$. For $n_0 > 4$, $2^n < c(n-1)!$ with $c = 1$. For $n_0 < 4$, a sufficiently large c can satisfy the preceding inequality. For $n = 1$ we have $c > 2$, for $n = 2$ we have $c > 4$, for $n = 3$ we have $c > 4$, for $n = 4$ we have $c > \frac{8}{3}$. Therefore $2^n = o((n-1)!)$

$$\lim_{n \rightarrow \infty} \frac{(n-1)!}{n!} = \frac{1}{n} = 0, \text{ therefore } (n-1)! = o(n!)$$

For any positive constant $c > 0$ there exists a constant $n_0 > 0$ such that $0 \leq n! < cn^n$ for all $n \geq n_0$. For $n_0 > 1$, $n! < cn^n$ with $c = 1$. For $n_0 \leq 1$, a sufficiently large c can satisfy the preceding inequality. For $n = 0$ we have $c > 1$, for $n = 1$ we have $c > 1$. Therefore $n! = o(n^n)$

Problem 2

From page 1151 of the book, for a series $\sum_{i=1}^n a_i$ if we let $a_{max} = \max_{1 \leq i \leq n} a_i$, then $\sum_{i=1}^n a_i \leq n \cdot a_{max}$

So we have $f(n) = O(n^{d+1}lg^m n)$ for $c \geq 1$ and $n_0 = 1$, and $d > 0, m \geq 1$

To find the lower bound we have $\sum_{i=1}^n i^d lg^m i \geq \sum_{i=\frac{n}{2}}^n \frac{n}{2}^d lg^m \frac{n}{2} = \frac{n}{2}^{d+1} lg^m \frac{n}{2} = \frac{n}{2}^{d+1} (lgn - 1)^m$

Now we have $(\frac{1}{2})^{d+1} n^{d+1} (lgn - 1)^m \geq \frac{1}{2}^{d+1} n^{d+1} (\frac{lgn}{2})^m = (\frac{1}{2})^{d+m+1} n^{d+1} lg^m n$ for $n_0 \geq 8$

So we have $f(n) = \Omega(n^{d+1}lg^m n)$ for $d > 0, m \geq 1, c \leq 4, n_0 \geq 8$

We have $c \leq 4$ because the biggest possible value for $(\frac{1}{2})^{d+m+1}$ is $\frac{1}{4}$

We have $n_0 \geq 8$ because $(lgn - 1)^m \geq (\frac{lgn}{2})^m$ for $n_0 \geq 8$

Because we have big-O and Ω , $f(n) = \Theta(n^{d+1}lg^m n)$

Problem 3

a) Case 1 of the Master Theorem applies here. $f(n) = n^2lgn$ and $n^{\log_b a} = n^{4.03}$. $f(n) = O(n^{4.03-\epsilon})$ for $\epsilon = 1$, therefore $T(n) = \Theta(n^{\log_4 17})$

b) The Master Theorem does not apply here because $nlg^2(n) \in o(n^{1+\epsilon})$

level	instances	size/each	cost/each	total cost
0	1	n	$nlg^2 n$	$nlg^2 n$
1	2	$\frac{n}{2}$	$\frac{n}{2}lg^2 \frac{n}{2}$	$nlg^2 \frac{n}{2}$
i	2^i	$\frac{n}{2^i}$	$\frac{n}{2^i}lg^2 \frac{n}{2^i}$	$nlg^2 \frac{n}{2^i}$
k	2^k	$\frac{n}{2^k}$	c_0	$c_0 2^k$

$$c_0 n + n \sum_{i=0}^{k-1} lg^2 \frac{n}{2^i} = c_0 n + n \sum_{i=0}^{k-1} lg^2 n - 2ilgn + i^2 = c_0 n + knlg^2 n + n \sum_{i=0}^{k-1} i^2 - 2ilgn = c_0 n + knlg^2 n + \frac{n(k-1)(k)(2k-1)}{6} - nlg n(k)(k-1)$$

With $k = lg(n)$ we have $T(n) = \Theta(nlg^3 n)$

c) The Master Theorem does not apply here because $\frac{n^2}{lg(n)} \in o(n^{2+\epsilon})$

level	instances	size/each	cost/each	total cost
0	1	n	$\frac{n^2}{lg(n)}$	$\frac{n^2}{lg(n)}$
1	4	$\frac{n}{2}$	$\frac{n^2}{4lg \frac{n}{2}}$	$\frac{n^2}{lg \frac{n}{2}}$
i	4^i	$\frac{n}{2^i}$	$\frac{n^2}{4^i lg \frac{n}{2^i}}$	$\frac{n^2}{lg \frac{n}{2^i}}$
k	4^k	$\frac{n}{2^k}$	c_0	$c_0 4^k$

$$c_0 n^2 + n^2 \sum_{i=0}^{k-1} \frac{1}{lg n - i}$$

The terms in the summation are $1 + \frac{1}{2} + \dots + \frac{1}{lg n}$ therefore it can be bounded by the harmonic series which gives $c_0 n^2 + n^2 \sum_{i=0}^{k-1} \frac{1}{lg n - i} \leq c_0 n^2 + n^2 ln(n)$

Taking the integral gives a tighter bound $\int_0^{k-1} \frac{1}{lg n - i} di = ln(lgn)$

Therefore we have $T(n) = \Theta(n^2 ln(lgn))$

d) Case 3 of the Master Theorem applies $\frac{n^{\frac{5}{4}}}{lg^2 n} = \Omega(n^{\log_4 5 + \epsilon})$

$$\lim_{n \rightarrow \infty} \frac{n^{1.16+\epsilon}}{\frac{n^{1.25}}{lg^2 n}} = \frac{n^{1.16+\epsilon}lg^2 n}{n^{1.25}} = \frac{lg^2 n}{n^{.09-\epsilon}}$$

$$\frac{\frac{d}{dn}lg^2 n}{\frac{d}{dn}n^{.09-\epsilon}} = \frac{2lgn}{n \cdot .09n^{-.91-\epsilon}ln2} = \frac{2lgn}{.09n^{.09-\epsilon}ln2}$$

$$\frac{\frac{d}{dn}2lgn}{\frac{d}{dn}.09n^{.09-\epsilon}ln2} = \frac{2}{.09^2 n^{.09-\epsilon}2ln2}$$

$\lim_{n \rightarrow \infty} \frac{2}{.09^2 n^{.09-\epsilon}2ln2} = 0$ for $\epsilon < .09$, therefore $\frac{n^{\frac{5}{4}}}{lg^2 n} = \omega(n^{\log_4 5 + \epsilon})$ and $T(n) = \Theta(\frac{n^{\frac{5}{4}}}{lg^2 n})$

Problem 4

a) The induction hypothesis for UNUSUAL is that if each half of the array is sorted when UNUSUAL is called, the entire array will be sorted when it returns.

In the base case where $n = 2$, each half of the array is sorted trivially and the elements will be swapped if they are out of order.

For $n > 2$, the for loop swaps the second quarter of the array with the third quarter of the array. This is done in order to sort the entire array, as the halves may not be sorted relative to one another. This swap can potentially make the halves unsorted, so recursive calls are made on each half. After each half is sorted again, another recursive call is made on the portion consisting of the quarters that have just been swapped, as they have not been sorted relative to one another. The recursive calls maintain the invariant that each half is sorted, as the quarters are already sorted and are now halves for the recursive calls. Since every call maintains the invariant, when UNUSUAL returns, the array is sorted.

The induction hypothesis for CRUEL is that the array will be sorted when it returns.

In the base case where $n = 1$ the array is sorted trivially.

For $n \geq 2$, CRUEL calls UNUSUAL on the entire array, each half is sorted so UNUSUAL returns a sorted array. It then unwinds to $n = 4$, where before calling UNUSUAL on the entire array, CRUEL calls itself recursively on each half of the array. This ensures each half will be sorted when it calls UNUSUAL, therefore UNUSUAL returns a sorted array. This repeats as the stack unwinds further, at which point the original array is sorted.

b) The array [3 4 1 2] would not sort correctly if the for loop were removed. Cruel would return [3 1 4 2]

c) The array [3 4 1 2] would not sort correctly if the last 2 lines were swapped. Cruel would return [1 3 2 4]

d) $T(n) = 3T(\frac{n}{2}) + \frac{n}{4} + 1$ and $T(2) = 3$
 $f(n) = \frac{n}{4} + 1$ and $n^{\log_b a} = n^{\lg 3} = n^{1.58}$
 Case 1 of the Master Theorem applies $\frac{n}{4} = O(n^{1.58-\epsilon})$
 $\lim_{n \rightarrow \infty} \frac{4n^{1.58}}{nn^\epsilon} = 4n^{.58-\epsilon} = \infty$ for $\epsilon < .58$
 Therefore $T(n) = \Theta(n^{\lg 3})$

e) $T(n) = 2T(\frac{n}{2}) + \Theta(n^{\lg 3})$ and $T(1) = 1$
 $f(n) = n^{\lg 3} = n^{1.58}$ and $n^{\log_b a} = n^{\lg 2} = n$
 Case 3 of the Master Theorem applies $n^{1.58} = \Omega(n^{1+\epsilon})$
 $\lim_{n \rightarrow \infty} \frac{n^{1.58}}{n^{1+\epsilon}} = n^{.58-\epsilon} = \infty$ for $\epsilon < .58$
 Therefore $T(n) = \Theta(n^{\lg 3})$