ECS 122A Homework 2

Hardy Jones
999397426
Professor Bai
Spring 2014

1. (a) We can expand some of the polynomial

$$(n+2)^{10} = n^{10} + 20n^9 + \cdots + 1024$$

Looking at everything except the first term, we see that the greatest power is n^9. We can set a *little-oh* bound of n^{10}, because $\forall c > 0, \exists n_0 > 0$ such that $0 \leq 20n^9 + \cdots + 1024 < cn^{10}, \forall n \geq n_0$. That is:

$$(n+2)^{10} = n^{10} + o(n^{10})$$

Now, this is $\Theta(n^{10})$ because
$\exists c_1 > 0, c_2 > 0, n_0 > 0$ such that $0 \leq c_1 n^{10} \leq n^{10} + o(n^{10}) \leq c_2 n^{10},\ \forall n \geq n_0$
This simplifies to $0 \leq c_1 \leq 1 + \frac{o(n^{10})}{n^{10}} \leq c_2$.
Now, since $o(n^{10})$ is an upper bound we know the largest the term $\frac{o(n^{10})}{n^{10}}$ can be is 1. So, this further siplifies to $0 \leq c_1 \leq 2 \leq c_2$.
So this is $\Theta(n^{10})$ if we choose appropriate constants. For instance we can choose $c_1 = c_2 = 2$.
Thus, $(n+2)^{10} = \Theta(n^{10})$

(b) We can apply reasoning similar to the previous problem.
We can expand some of the polynomial

$$(n+a)^b = n^b + \binom{b}{1} n^{b-1} a + \cdots + a^b$$

Looking at everything except the first term, we see that the greatest power is n^{b-1}. We can set a *little-oh* bound of n^b, because $\forall c > 0, \exists n_0 > 0$ such that $0 \leq \binom{b}{1} n^{b-1} a + \cdots + a^b < cn^b, \forall n \geq n_0$. That is:

$$(n+a)^b = n^b + o(n^b)$$

Now, this is $\Theta(n^b)$ because
$\exists c_1 > 0, c_2 > 0, n_0 > 0$ such that $0 \leq c_1 n^b \leq n^b + o(n^b) \leq c_2 n^b,\ \forall n \geq n_0$
This simplifies to $0 \leq c_1 \leq 1 + \frac{o(n^b)}{n^b} \leq c_2$.
Now, since $o(n^b)$ is an upper bound we know the largest the term $\frac{o(n^b)}{n^b}$ can be is 1. So, this further siplifies to $0 \leq c_1 \leq 2 \leq c_2$.
So this is $\Theta(n^b)$ if we choose appropriate constants. For instance we can choose $c_1 = c_2 = 2$.
Thus, $(n+a)^b = \Theta(n^b)$

2. "The running time of algorithm A is at least $O(n^2)$," states that the running time of "A" can be greater than $O(n^2)$. This has no meaning because *big-oh* notation states an asymptotic upper bound on a function. Asserting that the running time can be greater than this value negates the reason for using a bound in the first place.

3. (a) Yes. We need to find some positive constants c, n_0 such that $0 \leq 2^{n+1} \leq c2^n \forall n \geq n_0$.

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

. So we have our $c = 2, n = 1$.
Thus $2^{n+1} = O(2^n)$.

   (b) No. We need to show that no constants exist that satisfy the definition.

$$2^{2n} = 2^n \cdot 2^n \leq c2^n$$

.
It's easy to see that no constant will make this equation true for all values of n.
Thus $2^{2n} \neq O(2^n)$.

4. (a) From smallest to largest:

Function	Notes
lg$n$	
$n$lg$n$	
n	
n^2, n^2+lg$n$	These are the same asymptotically because the lg$n$ term will "drop-out"
n^3	
$n - n^3 + 7n^5$	
2^n	

   (b) From smallest to largest:

Function	Notes
1	
lg lg$n$	
lg$n$, ln$n$	The change of base merely changes the value by a constant factor
(lg$n$)^2	
$n$lg$n$	
$\sqrt{n}$, $\sqrt{2}^{\lg n}$, $n^{1+\epsilon}$	$\sqrt{n} = \sqrt{2}^{\lg n}$, $n^{1+\epsilon}$ differs by constant factor
n	
n^2, n^2+lg$n$	These are the same asymptotically because the lg$n$ term will "drop-out"
n^3	
$n - n^3 + 7n^5$	
2^n, 2^{n-1}	See Problem 3.a
e^n	
$n!$	

5. (a) $a = 2, b = 4, f(n) = 1, n^{log_4 2} = n^{\frac{1}{2}} = \sqrt{n}$

   We can see this fits case 1 of the master theorem, where we let $\epsilon = \frac{1}{2}$.

   So we have $f(n) = O(n^{log_4 2 - \frac{1}{2}}) = O(n^{\frac{1}{2} - \frac{1}{2}}) = O(n^0) = O(1)$.

   Thus, $T(n) = \Theta(n^{\frac{1}{2}}) = \Theta(\sqrt{n})$

   (b) $a = 2, b = 4, f(n) = \sqrt{n}, n^{log_4 2} = n^{\frac{1}{2}} = \sqrt{n}$

   This is exactly case 2 of the master theorem.

   Thus, $T(n) = \Theta(\sqrt{n} \lg n)$

   (c) $a = 2, b = 4, f(n) = n, n^{log_4 2} = n^{\frac{1}{2}} = \sqrt{n}$

   This might fit case 3 of the master theorem (with $\epsilon = \frac{1}{2}$) if we can find an appropriate constant c.

$$
\begin{aligned}
af\left(\frac{n}{b}\right) &= 2f\left(\frac{n}{4}\right) \\
&= 2\left(\frac{n}{4}\right) \\
&= \frac{1}{2}n
\end{aligned}
$$

   So, if we let $c = \frac{1}{2}$, then we satisfy the conditions of case 3 of the master theorem.

   Thus, $T(n) = \Theta(n)$

   (d) $a = 2, b = 4, f(n) = n^2, n^{log_4 2} = n^{\frac{1}{2}} = \sqrt{n}$

   This might fit case 3 of the master theorem (with $\epsilon = \frac{3}{2}$) if we can find an appropriate constant c.

$$
\begin{aligned}
af\left(\frac{n}{b}\right) &= 2f\left(\frac{n}{4}\right) \\
&= 2\left(\frac{n}{4}\right)^2 \\
&= 2\left(\frac{n}{16}\right) \\
&= \frac{1}{8}n
\end{aligned}
$$

   So, if we let $c = \frac{1}{8}$, then we satisfy the conditions of case 3 of the master theorem.

   Thus, $T(n) = \Theta(n^2)$