Set 2 Homework, Analysis of Algorithms

Jay R Bolton

May 6, 2012

- P 52: 3.1-1, 3.1-2
- P 60: 3.2-1, 3.2-2 and Problems: 3-1, 3-3, 3-4
- P 107 Problems: 4-1, 4-2, 4-4

Chapter 3

3.1-1 Prove: $max(f(n), g(n)) = \Theta(f(n) + g(n))$

By theorem 3.1, in order for a func to be big-Theta, it should be both big-O and big-Omega.

$$\begin{aligned}
&\mathcal{O}:\\
&max(f(n), g(n)) \le f(n) + g(n)\\
&max(f(n), g(n)) = O(f(n) + g(n))\\
&\Omega:\\
&2 * max(f(n), g(n)) >= f(n) + g(n)\\
&max(f(n), g(n)) >= f(n) + g(n) * 1/2\\
&max(f(n), g(n)) = \Omega(f(n) + g(n))
\end{aligned}$$

3.1-2 Prove: $(n + a)^b = Theta(n^b)$

Similarly to 3.1-1, we need to prove that the RHS is both big-O and big-Omega of the LHS.

$$\begin{aligned}
&\mathcal{O}:\\
&Show: (n + a)^b \le n^b * c && \text{for some constant c}\\
&Where: b > 0\\
&Cases:\\
&a \le 0:\\
&(n - a))^b < n^b\\
&(n - a))^b = O(n^b)\\
&a > 0:\\
&n + a \le n * a\\
&(n + a)^b \le (n * a)^b = n^b * a^b\\
&(n + a)^b \le n^b * a^b\\
&(n + a)^b = O(n^b) && \text{with constant } a^b \text{ for } a > 0
\end{aligned}$$

$\Omega:$
$a \geq 0:$
$(n+a) \geq n$
$(n+a)^b \geq n^b$
$(n+a)^b = \Omega(n^b)$
$a < 0:$
$(n-a) \geq n \cdot -a$
$(n-a)^b \geq (n \cdot -a)^b$
$(n-a)^b \geq n^b \cdot -a^b$
$(n-a)^b = \mathcal{O}(n^b)$ with constant a^b for $a < 0$

3.2-1

$Show:$
If f(n) and g(n) are monotonically increasing, then so are:
$f(n) + g(n):$
$f(n) \leq f(m)$
$g(n) \leq g(m)$
$f(n) + g(n) \leq f(m) + g(m)$
$f(g(n)):$
$f(n) \leq f(m)$
$g(n) \leq g(m)$
$f(g(n)) \leq f(g(m))$
$*$ Let: $g(n) = p$ and $g(m) = q$
$*$ We know that $p \leq q$ because it was stated that $g(n) \leq g(m)$
$*$ We already said $f(n) \leq f(m)$ for all $n \leq m$, and that $p \leq q$
$*$ Thus $f(p) \leq f(q)$, that is $f(g(n)) \leq f(g(m))$

$Show:$
If $f(n)$ and $g(n)$ are nonnegative, then:
$f(n) \cdot g(n)$ is monotonically increasing
$Definitions:$
$*\, f(n) \leq f(m)$ forall $n \leq m$
$*\, g(n) \leq g(m)$ forall $n \leq m$
$*\, f(n) > 0$ forall n
$*\, g(n) > 0$ forall n
$Conclusions:$
$*$ Since $f(n)$ and $g(n)$ are monotonically increasing and only positive, then they will only be positively increasing.
$*\, f(n) \cdot g(n) \leq f(m) \cdot g(m)$ forall $n \leq m$
$*$ This holds true because increasing positive integers multiplied will still be increasing.

3.2-2

$Prove:$

$a^{log(b,c)} = c^{log(b,a)}$

I assume we can use the equations above this one.

$Definition: q = b^y <=> log(b,q) = y$

$a^{log(b,c)} = c^{log(b,a)}$

$= log(c, a^{log(b,c)}) = log(b,a)$

$= log(b,c) * log(c,a) = log(b,a)$

$= log(c,a) = log(b,a)/log(b,c)$

$= log(c,a) = log(c,a)$

This used equations on p56 above the equation we proved.

3-1 The following is a lemma that I'll use for this problem:

a. Prove: $k \geq d \to p(n) = \mathcal{O}(n^k)$

$$Show: \sum_{i=0}^{d} a_i n^i \leq c \cdot n^k \text{ for some constant c}$$

Let $a_m = max(a_i)$

$$\sum_{i=0}^{d} a_i n^i \leq (a_m d) \cdot n^d \leq (a_m d) \cdot n^k$$

$$\sum_{i=0}^{d} a_i n^i = \mathcal{O}(n^k) \qquad \text{with constant } (a_m \cdot d)$$

b. Prove: $k \leq d \to p(n) = \Omega(n^k)$

$$Show: \sum_{i=0}^{d} a_i n^i \geq c \cdot n^k \text{ with some constant c}$$

$$\sum_{i=0}^{d} a_i n^i \geq n^d \geq n^k$$

$$\sum_{i=0}^{d} a_i n^i = \Omega(n^k) \qquad \text{with constant 1}$$

c. Prove: $k = d \to p(n) = \Theta(n^k)$

See proof in (a) and (b); by Theorem 3.1, n^d is also Θ.

$$Show: \sum_{i=0}^{d} a_i n^i \geq c \cdot n^d \text{ with some constant c}$$

$$Also: \sum_{i=0}^{d} a_i n^i \leq e \cdot n^d \text{ with some constant e}$$

$$\sum_{i=0}^{d} a_i n^i \leq (a_m d) \cdot n^d$$

$$\sum_{i=0}^{d} a_i n^i \geq n^d$$

d. Prove: $k > d \rightarrow p(n) = o(n^k)$

$$Show: \sum_{i=0}^{d} a_i n^i < c \cdot n^k \text{ with some constant c}$$

$$\sum_{i=0}^{d} a_i n^i \leq (a_m d) \cdot n^d < (a_m d) \cdot n^k$$

e. Prove: $k < d \rightarrow p(n) = \omega(n^k)$

$$Show: \sum_{i=0}^{d} a_i n^i > c \cdot n^k \text{ with some constant c}$$

$$\sum_{i=0}^{d} a_i n^i \geq n^d > n^k$$

3-3 From largest to smallest:

2^{2^n}
$(n+1)!$
$n!$
e^n
$n \cdot 2^n$
2^n
$(3/2)^n$
$(lgn)^{lgn}$
$(lgn)!$
n^3
n^2
$nlgn, lg(n!)$
n
$2^{\sqrt{2lgn}}$
$(lgn)^2$
lgn
$\sqrt{lgn}$
$lglgn$
$2^{lg \cdot n}$
$(lgn)*$
$n^{1/lgn}$

Some more of them may be in equivalence classes...

3-4 **a.** False by counterexample: n and n^2

b. False by counterexample: n and n^2

c. True: $f(n) \leq g(n)$ and $lg(f(n)) \leq lg(g(n))$

d. True: $f(n) \leq g(n)$ and $2^{f(n)} \leq 2^{g(n)}$

e. True: $f(n) \leq f(n)^2$

f. True by transpose symmetry.

g. False: $n^2 > c * (n/2)^2$

h. True:

$$f(n) + o(n) \leq 2 \cdot f(n)$$
$$f(n) + o(n) \geq 1/2 \cdot f(n)$$

Chapter 4

4-1 I'll use master theorem on all of them (because it's quick/easy), except for the last.

a.

$$a = 2$$
$$b = 2$$
$$f(n) = n^4$$
$$f(n) = \Omega(n^{log_2^2+e})$$
$$2 \cdot (n^4)/(2^4) \leq c \cdot n^4$$
$$T(n) = \Theta(n^4)$$

b.

$$a = 1$$
$$b = 7/10$$
$$f(n) = n$$
$$f(n) = \Omega(n^{log_{7/10}^1+e})$$
$$e = 1$$
$$1 \cdot (n)/(7/10) \leq c \cdot n$$
$$T(n) = \Theta(n)$$

c.

$$a = 16$$
$$b = 4$$
$$f(n) = n^2$$
$$f(n) = \Theta(n^{log_4^{16}})$$
$$T(n) = \Theta(n^2 lgn)$$

d.

$$a = 7$$
$$b = 3$$
$$f(n) = n^2$$
$$f(n) = \Omega(n^{log_3^7+e})$$
$$7 \cdot (n^2)/(3^2) \leq c \cdot n^2$$
$$T(n) = \Theta(n^2)$$

e.

$$a = 7$$
$$b = 2$$
$$f(n) = n^2$$
$$f(n) = \mathcal{O}(n^{log_2^7-e})$$
$$T(n) = \Theta(n^{log_2^7})$$

f.

$$a = 2$$
$$b = 4$$
$$f(n) = \sqrt{n}$$
$$f(n) = \Theta(n^{log_4^2})$$
$$T(n) = \Theta(n^{1/2} lgn)$$

g. This recurrence can be represented by the sum:

$$
\begin{aligned}
& c \cdot \sum_{i=0}^{\lfloor n/2 \rfloor} n^2 \\
& = c \cdot \frac{(n^2/4+1)(n+1)}{6} \\
& = c \cdot \frac{n^3/4 + n^2/4 + n + 1}{6} \\
& = c \cdot (n^3 \cdot 1/24 + n^2 \cdot 1/24 + n/6 + 1/6) \\
& = \Theta(n^3)
\end{aligned}
$$

4-2 **a.** The recursive representation of the complexity of binary sort is: $T(n) = T(n/2) + 1$
For case 1, our complexity is $T(lgn)$ by the master theorem.
For case 2, our recurrence is $T(n') = T(n'/2) + n$ and our complexity for this is $T(nlgn)$.
For case 3, our complexity is still $T(nlgn)$.

b. For case 1, $T(n) = 2T(n/2) + n$ which is $\Theta(nlgn)$ by the master theorem.
For case 2, $T(n') = 2T(n'/2) + 4n$ which is $\Theta(nlgn)$ by the master theorem.
For case 3 $T(n) = 2T(n/2) + 4n$ which is $\Theta(nlgn)$ by the master theorem.

4-4 Show: $\mathcal{F}(z) = z + z\mathcal{F}(z) + z^2\mathcal{F}z$

$$
\begin{aligned}
\mathcal{F}(z) &= \sum_{i=0}^{\infty} \mathcal{F}_i z^i \\
&= z + \sum_{i=2}^{\infty} (\mathcal{F}_{i-1} + \mathcal{F}_{i-2}) z^i \\
&= z + \sum_{i=2}^{\infty} \mathcal{F}_{i-1} z^i + \sum_{i=2}^{\infty} \mathcal{F}_{i-2} z^i \\
&= z + z \sum_{i=0}^{\infty} \mathcal{F}_i z^i + z^2 \sum_{i=0}^{\infty} \mathcal{F}_i z^i \\
&= z + z\mathcal{F}(z) + z^2\mathcal{F}(z)
\end{aligned}
$$

Only had time to figure out the first part of this problem.