INTRO TO ALGORITHM, CSE 241

PROFESSOR ZHANG

Homework 1

Peiyun(Seed) Zeng
EN:437801

September 22, 2014

PROBLEM 1

Decide whether the following statement is True or False. You must justify your answer to receive full credit (if you think it is true, argue it; if you think it is false, give a counter example).

(a) $\mathbf{f}(\mathbf{n}) = \mathbf{O}(\mathbf{g}(\mathbf{n})) \wedge \mathbf{f}(\mathbf{n}) = \Omega(\mathbf{h}(\mathbf{n})) \longrightarrow \mathbf{g}(\mathbf{n}) = \theta(\mathbf{h}(\mathbf{n}))$

Solution:
This is False. **Counter example:** $f(n) = n^2; g(n) = n^3; h(n) = n$. which apparently satisfies our constrain ($\mathbf{f}(\mathbf{n}) = \mathbf{O}(\mathbf{g}(\mathbf{n})) \wedge \mathbf{f}(\mathbf{n}) = \Omega(\mathbf{h}(\mathbf{n})) \longrightarrow \mathbf{g}(\mathbf{n}) = \theta(\mathbf{h}(\mathbf{n}))$.
Yet, it ***does not*** follow that $g(n) = n^3 = \theta(h(n)) = \theta(n)$

INTRO TO ALGORITHM, CSE 241

PROFESSOR ZHANG

Homework 1

Peiyun(Seed) Zeng
EN:437801

September 22, 2014

PROBLEM 2

(a) $T(n) = 7T(n/2) + n^3 * \log n + 10n * \log n$

Solution:
Let's apply Master Method

1. $a = 7, b = 2 \longrightarrow x = \log_b a = \log_2 7 \approx 2.8$
2. $n^3 * \log n$ grows much faster $\rightarrow f(n) = n^3 * \log n + 10n * \log n = \Theta(n^3 * \log n) \rightarrow y = 3, k = 1$
3. $y > x \rightarrow$ This Corresponds to Case 3 – The root Dominates since f(n) grows faster than x
4. **So we can conclude:** $T(n) = \Theta(f(n)) = \Theta(n^3 * \log n)$

Final Result:$T(n) = \Theta(f(n)) = \Theta(n^3 * \log n)$

(b) $T(n) = 2T(n/8) + \sqrt[3]{n} + 10\lg n$

Solution:
Let's apply Master Method for this one

1. $a = 2, b = 8 \rightarrow x = \log_b a = \log_8 2 = (1/3)$
2. $\sqrt[3]{n}$ grows significantly faster than$10\lg n \rightarrow f(n) = \sqrt[3]{n} + 10\lg n = \Theta(\sqrt[3]{n}) \rightarrow y = 1/3, k = 0$
3. This corresponds to Case 2– Roots has equal wight to the leaf since x grows roughly the rate as the leaf
4. **So we can conclude:** $T(n) = \Theta(n^x(\log n)^{(k+1)}) = n^{(1/3)} * \log n$

Final Result: $T(n) = \Theta(n^x(\log n)^{(k+1)}) = n^{(1/3)} * \log n$

(c) $T(n) = 2T(\sqrt{n}) + \lg\lg n$

Solution:

Unfortunately, this cannot be directly solved by Master Method. Let's start from substitution r

1. Let $m = \lg n \equiv n = 2^m \rightarrow$ now substitute into the original equation
2. $T(2^m) = 2 * T(2^{m/2}) + \lg m$ (equation [1])
3. If we look at the equation, we realize m is our new variable
4. So now let $S(m) = 2 * S(m/2) + \lg m$ (equation [2], from equation[1])
5. equation [2] can be solved using **Master theorem**
6. $a = 2, b = 2, x = \log_2 2 = 1$
7. $f(n) = \lg m = \Theta(n^0 * (\log m)^1 \rightarrow y = 0, k = 1$
   (Since Log function grows really slow,we ignore the differences between bases)
8. $x > y \rightarrow$ This corresponds to Case 1
9. So we can conclude from master theorem :$S(m) = \Theta(m^1) = \Theta(m) = \Theta(\lg n)$
   (Substitute $m = \log n$ back)

Final Result: T(n) = $\lg n$

INTRO TO ALGORITHM, CSE 241

PROFESSOR ZHANG

Homework 1

Peiyun(Seed) Zeng
EN:437801

September 22, 2014

Since I realized there is no extra credit for using latex. and i donot want to draw a tree using latex. Solution will be presented in handwritten (Next Page)
Welcome to the messy part of the story :) Good Luck