Discrete Mathematics #3

20192208 김형훈

8.3

Problem 29

$n = b^k$

$k = 0, n = 1$

$f(1) = f(1)\,\text{log}_b\, 1 = 0$

Inductive step: Assume the formula holds for some $k \geq 0$ For $n = b^{k+1}$:

$f(b^{k+1}) = af(\frac{b^{k+1}}{b}) + c(b^{k+1})^d$

$= b^d f(b^k) + cb^{(k+1)d}$ (since $a = b^d$)

$= b^d[f(1)(b^k)^d + c(b^k)^d \,\text{log}_b\, b^k] + cb^{(k+1)d}$ (by inductive hypothesis)

$= f(1)b^{(k+1)d} + cb^{(k+1)d}k + cb^{(k+1)d}$

$= f(1)b^{(k+1)d} + cb^{(k+1)d}(k+1)$

$= f(1)(b^{k+1})^d + c(b^{k+1})^d \,\text{log}_b\, b^{k+1}$

Therefore, the formula holds for all powers of b.

Problem 30

From Problem 29, we know that for powers of b: $f(n) = f(1)n^d + cn^d \,\text{log}_b\, n$

Since $f(1)$ and c are constants: $f(n) = O(n^d) + O(n^d \,\text{log}_b\, n) = O(n^d \,\text{log}\, n)$

For values of n that are not powers of b, since f is increasing: $f(n) \leq f(b^{\lceil \text{log}_b n \rceil}) = O((b^{\lceil \text{log}_b n \rceil})^d \,\text{log}\, b^{\lceil \text{log}_b n \rceil}) = O(n^d \,\text{log}\, n)$

Therefore, $f(n)$ is $O(n^d \,\text{log}\, n)$ for all positive integers n.

Problem 31

Let's verify this solution satisfies the recurrence relation. Substitute n/b for n:

$f(n) = af(\frac{n}{b}) + cn^d$

$= a[C_1(\frac{n}{b})^d + C_2(\frac{n}{b})^{log_b a}] + cn^d$

$= aC_1\frac{n^d}{b^d} + aC_2\frac{n^{log_b a}}{b^{log_b a}} + cn^d$

$= \frac{aC_1}{b^d}n^d + C_2 n^{log_b a} + cn^d$

For this to match our proposed solution: $C_1 = \frac{aC_1}{b^d} + c$

Solving for C_1: $C_1(1 - \frac{a}{b^d}) = c$ $C_1 = \frac{b^d c}{b^d - a}$

And C_2 can be determined from the initial condition $f(1) = C_1 + C_2$.

Problem 32

From Problem 31, we have: $f(n) = C_1 n^d + C_2 n^{log_b a}$

When $a < b^d$, we have $log_b a < d$

Therefore, $n^{log_b a}$ grows slower than n^d

Thus, $f(n) = C_1 n^d + C_2 n^{log_b a} = O(n^d)$

Problem 33

From Problem 31, we have: $f(n) = C_1 n^d + C_2 n^{log_b a}$

When $a > b^d$, we have $log_b a > d$

Therefore, $n^{log_b a}$ grows faster than n^d

Thus, $f(n) = C_1 n^d + C_2 n^{log_b a} = O(n^{log_b a})$