

GRADED ASSIGNMENT ALGORITHM

MASTER THEOREM SOLUTION

a) $T(n) = 3T(n/2) + n$

b) $T(n) = 64T(n/8) - n^2(\log n)$

c) $T(n) = 2nT(n/2) + n^n$

d) $T(n) = 3T(n/3) + n/2$

e) $T(n) = 7T(n/3) + n^2$

A) $T(n) = 3T(n/2) + n$

Solution: $\Theta(n^{\log_a b})$ $a = 3$ $b = 2$ $f(n) = \Theta(n) \Rightarrow d = 1$ $b^d = 2 = 2 \Rightarrow a > b^d$

So, Case 3. Thus, $T(n) = \Theta(n^{\log_a b})$

b) $T(n) = 64T(n/8) - n^2(\log n)$

Solution: NA $f(n) = n^2(\log n)$

Here $f(n)$ is not in the form of $\Theta(n^d)$.

So we cannot apply Master Theorem.

c) $T(n) = 2nT(n/2) + n^n$

Solution: NA $f(n) = n^n$ is not in the form of $\Theta(n^d)$.

So we cannot apply Master Theorem.

d) $T(n) = 3T(n/3) + n/2$

Solution: $\Theta(n^3 \log n)$ $a = 3$ $b = 3$ $f(n) = \Theta(n) \Rightarrow d = 1$ $b^d = 3 = a \Rightarrow a = b^d$

So, Case 2. Thus, $T(n) = \Theta(n^d \log n) = \Theta(n^3 \log n)$

e) $T(n) = 7T(n/3) + n^2$

Solution: $\Theta(n^2)$ $a = 7$ $b = 3$ $f(n) = \Theta(n^2) \Rightarrow d = 2$ $b^d = 3^2 = 9 > a \Rightarrow a < b^d$

So, Case 1. Thus, $T(n) = \Theta(n^d) = \Theta(n^2)$