NAME:- KRUNAL RANK

ADM. NO:- U18CO081

BTECH 3RD YEAR

Scanned with CamScanner

U18CO081

17/9/20

DAA Tutorial 5

Q1.

a) Given $T(n) = 2T(n/2) + n^4$

Here $a = 2, b = 2$

And $f(n) = n^4$

$f(n) = O(n^4) = \Theta(n^4) = \Omega(n^4)$

But $n^{\log_b a} = n < n^4$. $\varepsilon = 3$

Hence,

$$T(n) = \Theta(n^4)$$

b) Given $T(n) = T(n-2) + n^2$

The master theorem cannot be applied in such cases because the recurrence relation is not of the form,

$T(n) = aT(n/b) + n^2$ and $b > 1$ but $b = 1$

c) Given $T(n) = 5T(n/2) + \Theta(n)$

Here $a = 5, b = 2$

$f(n) = \Theta(n)$

Now,

$n^{\log_b a + \varepsilon} = n^{2.1 - \varepsilon}$

$\Theta(n) = \Theta(n^{\log_b - \varepsilon})$

Hence,

$$T(n) = \Theta(n^{\log_2 5})$$

Date / /
Page

~~Ans~~

d) Given

$$T(n) = 2T(n/4) + \sqrt{n}$$

Here,

$a = 2$ $b = 4$

$f(n) = \sqrt{n} = \Theta(\sqrt{n})$

Now, $n^{\log_b a} = \Theta\sqrt{n} = \Theta(\sqrt{n})$

Hence, $f(n) = \Theta(n^{\log_b a})$

Hence, $T(n) = \Theta(\sqrt{n}\log n)$

e) $T(n) = \frac{1}{2}T(n/2) + \frac{1}{n}$

Here $a = \frac{1}{2}$ $b = 2$, $f(n) = \frac{1}{n}$

Here $a < 1/2$ but a needs to be greater than or equal to 1. Hence, master theorem can not be applied.

f) $T(n) = \sqrt{n}\,T(\sqrt{n}) + n$

Here $a = \sqrt{n}$ $b = 1$ and $f(n) = n$.

Here, $b = 1$ and ~~b ≠ 1~~ b is not greater than n.

Hence, master theorem cannot be applied.

Scanned with CamScanner

g) Given,

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

Here, $a=3$ $b=2$

$f(n) = n/2 = \Theta(n)$

Now, $c = n^{\log_b a} = n^{\log_2 3} = \Theta(n^{\log_2 3}) > \Theta(n)$

Hence,

$$\underline{\underline{T(n) = \Theta(n^{\log_2 3})}}$$

h) $T(n) = 2T(n/2) + \frac{n}{\log n}$

Here, $a=2, b=2$

$f(n) = \frac{n}{\log n}$

Here $c = n^{\log_b a} = \underline{n}$

We cannot apply master theorem because $f(n)$ is non polynomial and its difference between with c is also non polynomial.

Scanned with CamScanner

i) Given,

$$T(n) = T(n/2) + T(n/4) + T(n/8) + n$$

Here, the ~~to~~ recurrence is not of the form

$$T(n) = a(T(n/b)) + f(n)$$

but it can be reduced by neglecting lower order terms.

Hence, $T(n) = T(n/2) + n$

Here, $a = 1, b = 2, f(n) = n = O(n)$

$c = n^{\log_b a} = n^0 < O(n)$

Hence, $T(n) = O(n)$

j) Given,

$$T(n) = 4T(n/3) + n\log n.$$

Here, $a = 4, b = 3$

$f(n) = n\log n = O(n\log n)$

$c = n^{\log_3 4} = n^{1.26185} > O(n\log n)$

Hence, $T(n) =$ ~~$O(n\log n)$~~ $O(n^{1.26185})$

k ~~a~~) Given,

$$T(n) = 64\,T(n/8) + n^2\log n.$$

Here, $a = 64, b = 8$

$f(n) = n^2\log n = O(n^2\log n)$

$c = n^{\log_b a} = n^{\log_8 64} = n^2 = O(n^2) < O(n^2\log n)$

Hence, $T(n) = O(n^2\log n)$

Scanned with CamScanner

l) Given,

$$T(n) = 2T(n-1) + O(1)$$

Here, $a=2$, $b=1$.

But for master theorem to be applicable, $b>1$.

Hence, master theorem cannot be applied.

m) Given, $T(n) = 8T(n/4) + n^2 \log n$.

Here, $a=8$, $b=4$.

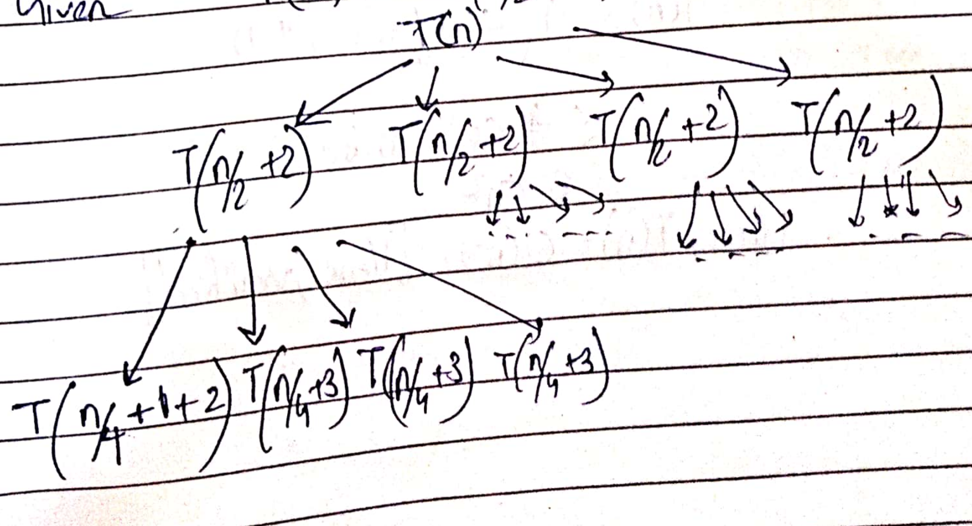
$f(n) = n^2 \log n$, $f(n) = O(n^2 \log n)$

$c = n^{\log_b a} = n^{\log_4 8} < O(n^2 \log n)$

Hence, $\underline{T(n) = \Theta(n^2 \log n)}$

Ans2.

Given $T(n) = 4T(n/2 + 2) + n$



At ith level, $T\left(\frac{n+2^{i+2}-4}{2^i}\right)$

no. of level $\approx \log n$

work at ith level.

$$\left(\frac{n}{2^i}+\frac{2^{i+2}}{2^i}-\frac{4}{2^i}\right)\times 4^i$$

Total ~~lo~~ Iterations $= \sum_{i=0}^{\log n}\left(n(2^i)+4^{i+1}-4(2)^i\right)$

$$\approx n^2+\frac{4n^2}{3}-4n$$

$$\simeq \underline{\underline{O(n^2)}}$$

Checking with substitution method,

$$T(n) = 4T(n/2+2)+n$$

let $T(n) = O(n^2)$

Then, $T\left(\frac{n}{2}+2\right) \leq c\left(n/2+2\right)^2$

$$T(n) \leq 4\left(c\left(n/2+2\right)^2\right)+n$$

$$\leq \cancel{4}cn^2+n(8c+1)+16$$

$$\leq cn^2$$

Hence, $T(n) = O(n^2)$. [Hence, verified]

Ans 3) $T(n) = T(\alpha n) + T((1-\alpha)n) + O(n)$

$$T(n)$$

$$T(\alpha n) \qquad T((1-\alpha)n)$$

$$T(\alpha^2 n) \quad T(\alpha(1-\alpha)n) \quad T(\alpha(1-\alpha)n) \quad T((1-\alpha)^2 n)$$

Hence, number of levels,

$$\alpha^i n = O(1)$$

$$i = \log_\alpha\left(\frac{1}{n}\right)$$

Depth is $\log(n)$ at each level is $\sqrt{n}$. ~~The~~

Hence,

$$T(n) = \sum_{i=0}^{\log_\alpha 1/n} (n + O(n)) = n\log_\alpha n + O(n)$$

$$= O(n\log n)$$

Using substitution,

$$T(n) \leq dn\log n$$

$$T(n) \leq d\alpha n\log(\alpha n) + d\beta n\log(\beta n) + cn$$

$$\leq d\alpha n\log n + d\beta n\log(n) + d\alpha n\log\alpha + d\beta\log\beta + cn$$

$\alpha + \beta = 1$

$$T(n) \leq dn\log(\alpha+\beta) + d(\alpha\log\alpha + \beta\log\beta) + cn$$

$$\leq n\log n$$

$$T(n) = O(n\log n).$$

Hence, verified.