Revision and Activity

Master Theorem

Revision and Activity:

- Master Theorem

Solve the following recurrence equation using Master Theorem (if possible)

- $T(n) = 3T(^n/_2) + n^2$
- $T(n) = 4T\left(\frac{n}{2}\right) + n^2$
- $T(n) = 16T\left(\frac{n}{4}\right) + n$
- $T(n) = 2T\left(\frac{n}{2}\right) + n \lg n$
- $T(n) = 0.5T\left(\frac{n}{2}\right) + \frac{1}{n}$
- $T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$

$$T(n) = 3T\left({}^{n}/_{2}\right) + n^2$$

- $a = 3, b = 2, c = 2, f(n) = n^2$
- $\left(\frac{a}{b^c}\right) = \left(\frac{3}{2^2}\right) < 1, \mathit{hence\ may\ be\ able\ to\ test\ using\ case}\ 3.$

Test using case 3:

$$n^2 \in \Omega\left(n^{\log_2 3+\varepsilon}\right) \text{ for some } \varepsilon > 0.$$

$$n^2 \in \Omega(n^{1.6+\varepsilon}) \text{ true for } \varepsilon > 0.4.$$

Test if $af\left(\frac{n}{b}\right) \leq kf(n)$ for some constant k < 1.

$$3\left(\frac{n}{2}\right)^2 \leq kn^2$$

$$\frac{3}{4}n^2 \leq kn^2, \text{and this is true for } k = \frac{3}{4} < 1.$$

$$\therefore\ \Theta(f(n)) = \boldsymbol{\Theta}\left(\boldsymbol{n^2}\right)$$

$$T(n) = 4T\left(\frac{n}{2}\right) + n^2$$

- $a = 4, b = 2, c = 2, f(n) = n^2$
- $\left(\frac{a}{b^c}\right) = \left(\frac{4}{2^2}\right) = 1, \textit{hence may be able to test using case } 2.$

Test using case 2:

$n^2 \in \Theta\left(n^{\log_2 4}\right)$

$n^2 \in \Theta(n^2)$ this is true.

$$\therefore\ \Theta\left(n^{\log_b a} \lg n\right) = \Theta\left(n^{\log_2 4} \lg n\right) = \boldsymbol{\Theta\left(n^2 lg\, n\right)}$$

$$T(n) = 16T\left(\frac{n}{4}\right) + n$$

- $a = 16, b = 4, c = 1, f(n) = n$
- $\left(\frac{a}{b^c}\right) = \left(\frac{16}{4^1}\right) > 1, hence\ may\ be\ able\ to\ test\ using\ case\ 1.$

Test using case 1:

$n \in O\left(n^{\log_4 16-\varepsilon}\right)$ for some $\varepsilon > 0$.

$n \in O(n^{2-1})$ true for $\varepsilon = 1$.

$$\therefore\ \Theta\left(n^{\log_b a}\right) = \Theta\left(n^{\log_4 16}\right) = \boldsymbol{\Theta}\left(\boldsymbol{n^2}\right)$$

$$T(n) = 2T\left(\frac{n}{2}\right) + n\lg n$$

- $a = 2, b = 2, c = 1, f(n) = n\lg n$
- $\left(\frac{a}{b^c}\right) = \left(\frac{2}{2^1}\right) = 1, \mathit{hence\ may\ be\ able\ to\ test\ using\ case}\ 2.$

Test using case 2:

$$n\lg n \in \Theta\left(n^{\log_2 2}\right)$$

$n\lg n \in \Theta(n)$ this is NOT true.

Hence, master theorem cannot be applied. Expansion and substitution will be used instead.

$$T(n) = 2T\left(\frac{n}{2}\right) + n\lg n$$

$$T(n) = 2T\left(\frac{n}{2}\right) + n\lg n$$

$$T(n) = 2\left[2T\left(\frac{n}{2^2}\right) + \frac{n}{2}\lg\left(\frac{n}{2}\right)\right] + n\lg n$$

$$T(n) = 2^2T\left(\frac{n}{2^2}\right) + n(\lg n\ - 1) + n\lg n$$

Note: $\lg\left(\frac{n}{2}\right) = \lg n\ - \lg 2 = \lg n - 1$

$$T(n) = 2^2T\left(\frac{n}{2^2}\right) + n\lg n\ - n + n\lg n$$

$$T(n) = 2^2T\left(\frac{n}{2^2}\right) + 2n\lg n\ - n$$

Continue to next slide....

$$T(n) = 2T\left(\frac{n}{2}\right) + n \lg n$$

Continue from previous slide….

$$T(n) = 2^2 T\left(\frac{n}{2^2}\right) + 2n \lg n \ - n$$

$$T(n) = 2^2 \left[2T\left(\frac{n}{2^3}\right) + \frac{n}{2^2} \lg \frac{n}{2^2}\right] + 2n \lg n \ - n$$

$$T(n) = 2^3 T\left(\frac{n}{2^3}\right) + n(\lg n - 2) + 2n \lg n \ - n$$

Note: $\lg\left(\frac{n}{2^2}\right) = \lg n \ - 2$

$$T(n) = 2^3 T\left(\frac{n}{2^3}\right) + n \lg n - 2n + 2n \lg n \ - n$$

$$T(n) = 2^3 T\left(\frac{n}{2^3}\right) + 3n \lg n - 3n$$

Continue to next slide….

$$T(n) = 2T\left(\frac{n}{2}\right) + n\lg n$$

Continue from previous slide....

$$T(n) = 2^3 T\left(\frac{n}{2^3}\right) + 3n\lg n - 3n$$

$$T(n) = 2^3\left[2T\left(\frac{n}{2^4}\right) + \frac{n}{2^3}\lg\frac{n}{2^3}\right] + 3n\lg n - 3n$$

$$T(n) = 2^4 T\left(\frac{n}{2^4}\right) + n(\lg n - 3) + 3n\lg n - 3n$$ Note: $\lg\left(\frac{n}{2^3}\right) = \lg n - 3$

$$T(n) = 2^4 T\left(\frac{n}{2^4}\right) + n\lg n - 3n + 3n\lg n - 3n$$

$$T(n) = 2^4 T\left(\frac{n}{2^4}\right) + 4n\lg n - 6n$$

Continue to next slide....

The recurrence relation can be generalized as

$$T(n) = 2^k T\left(\frac{n}{2^k}\right) + k\, n \lg n - \left(\frac{k(k-1)}{2}\right) n \qquad \cdots\cdots\cdots eq(1)$$

The recursive call will continue and stop when $\left(\frac{n}{2^k}\right) = 1$. Solving the equality, we have $n = 2^k$ and hence $k = \lg n$.

Substituting k into $eq(1)$, we have

$$T(n) = 2^{\lg n} T\left(\frac{n}{2^{\lg n}}\right) + \lg n\, (n \lg n) - \left(\frac{\lg^2 n\ - \lg n}{2}\right) n$$

Continue from previous slide....

$$T(n) = 2^{\lg n} T\left(\frac{n}{2^{\lg n}}\right) + \lg n\,(n \lg n) - \left(\frac{\lg^2 n\ - \lg n}{2}\right) n$$

$$= nT\left(\frac{n}{n}\right) + n \lg^2 n - \frac{n}{2} \lg^2 n + \frac{n}{2} \lg n$$

$$= nc + \left(\frac{2n - n}{2}\right) \lg^2 n + \frac{n}{2} \lg n$$

$$= nc + \frac{1}{2} n \lg^2 n + \frac{1}{2} n \lg n$$

$\therefore$ *the running time complexity is* $\boldsymbol{\Theta\left(n \lg^2 n\right)}$.

$$T(n) = 0.5T\left(\frac{n}{2}\right) + \frac{1}{n}$$

- $a = 0.5, b = 2, c = -1, f(n) = \frac{1}{n}$
- a is less than 1, hence master theorem cannot be applied to determine the running time complexity of this recurrence relation. Expansion and substitution will be used instead.

$$T(n) = \frac{1}{2}T\left(\frac{n}{2}\right) + \frac{1}{n}$$

$$T(n) = \frac{1}{2}\left[\frac{1}{2}T\left(\frac{n}{2^2}\right) + \frac{1}{\frac{n}{2}}\right] + \frac{1}{n}$$

Continue to next slide…

$$T(n) = 0.5T\left(\frac{n}{2}\right) + \frac{1}{n}$$

Continue from previous slide…

$$T(n) = \frac{1}{2}\left[\frac{1}{2}T\left(\frac{n}{2^2}\right) + \frac{1}{\frac{n}{2}}\right] + \frac{1}{n}$$

$$T(n) = \frac{1}{2^2}T\left(\frac{n}{2^2}\right) + \left(\frac{1}{2}\right)\frac{2}{n} + \frac{1}{n}$$

Continue to next slide…

$$T(n) = 0.5T\left(\frac{n}{2}\right) + \frac{1}{n}$$

Continue from previous slide…

$$T(n) = \frac{1}{2^2}T\left(\frac{n}{2^2}\right) + \left(\frac{1}{2}\right)\frac{2}{n} + \frac{1}{n}$$

$$T(n) = \frac{1}{2^2}\left[\frac{1}{2}T\left(\frac{n}{2^3}\right) + \frac{1}{\frac{n}{2}}\right] + \left(\frac{1}{2}\right)\frac{2}{n} + \frac{1}{n}$$

$$T(n) = \frac{1}{2^3}T\left(\frac{n}{2^3}\right) + \left(\frac{1}{2^2}\right)\frac{2}{n} + \left(\frac{1}{2}\right)\frac{2}{n} + \frac{1}{n}$$

The recurrence relation can be generalized as

$$T(n) = \frac{1}{2^k} T\left(\frac{n}{2^k}\right) + \left[\frac{1}{2^k} + \frac{1}{2^{k-1}} + \frac{1}{2^{k-2}} + \cdots + \frac{1}{2^2} + \frac{1}{2^1}\right]\left(\frac{2}{n}\right) + \frac{1}{n} \qquad \cdots eq(1)$$

The recursive call will continue and stop when $\left(\frac{n}{2^k}\right) = 1$. Solving the equality, we have $n = 2^k$ and hence $k = \lg n$.

Substituting k into $eq(1)$, we have

$$T(n) = \frac{1}{2^{\lg n}} T\left(\frac{n}{2^{\lg n}}\right) + \left[\frac{1}{2^{\lg n}} + \frac{1}{2^{(\lg n)-1}} + \cdots + \frac{1}{2}\right]\left(\frac{2}{n}\right) + \frac{1}{n}$$

Continue to next slide…

Continue from previous slide…

$$T(n) = \frac{1}{2^{\lg n}} T\left(\frac{n}{2^{\lg n}}\right) + \left[\frac{1}{2^{\lg n}} + \frac{1}{2^{(\lg n)-1}} + \cdots + \frac{1}{2}\right]\left(\frac{2}{n}\right) + \frac{1}{n}$$

$$T(n) = \frac{1}{n} T\left(\frac{n}{n}\right) + \left[\frac{1}{2^{\lg n}} + \frac{1}{2^{(\lg n)-1}} + \cdots + \frac{1}{2}\right]\left(\frac{2}{n}\right) + \frac{1}{n}$$

$$T(n) = \frac{1}{n} T(1) + \left[\frac{1}{2^{\lg n}} + \frac{1}{2^{(\lg n)-1}} + \cdots + \frac{1}{2}\right]\left(\frac{2}{n}\right) + \frac{1}{n}$$

Continue from previous slide…

$$T(n) = \frac{1}{n}T(1) + \left[\frac{1}{2^{\lg n}} + \frac{1}{2^{(\lg n)-1}} + \cdots + \frac{1}{2}\right]\left(\frac{2}{n}\right) + \frac{1}{n}$$

$\Theta\left(\frac{1}{2^{\lg n}}\right) = \Theta\left(\frac{1}{n}\right)$. Does not seem to be sensible algorithm complexity because as n increases, the complexity reduces (becomes less complex).

$$T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$$

- $a = 64, b = 8, c = 2, f(n) = -n^2 \lg n$
- f(n)is negative, hence master theorem cannot be applied to determine the running time complexity of this recurrence relation. Expansion and substitution will be used instead.

- The running time complexity of the algorithm is $\boldsymbol{\Theta\left(n^2 \log_8{}^2 n\right)}$.

$$T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$$

$$T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$$

$$= 64\left[64T\left(\frac{n}{8^2}\right) - \left(\frac{n}{8}\right)^2 \lg\left(\frac{n}{8}\right)\right] - n^2 \lg n$$

$$= 64^2T\left(\frac{n}{8^2}\right) - n^2(\lg n - 3) - n^2 \lg n$$

$$= 64^2T\left(\frac{n}{8^2}\right) - 2n^2 \lg n + 3n^2$$

continue to next slide...

$$T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$$

Continue from previous slide…

$$= \ 64^2 T\left(\frac{n}{8^2}\right) - 2n^2 \lg n + 3n^2$$

$$= 64^2 \left[64T\left(\frac{n}{8^3}\right) - \left(\frac{n}{8^2}\right)^2 \lg\left(\frac{n}{8^2}\right)\right] - 2n^2 \lg n + 3n^2$$

$$= \ 64^3 T\left(\frac{n}{8^3}\right) - n^2(\lg n \ - 6) - 2n^2 \lg n + 3n^2$$

$$= \ 64^3 T\left(\frac{n}{8^3}\right) - n^2 \lg n + 6n^2 - 2n^2 \lg n + 3n^2$$

$$= \ 64^3 T\left(\frac{n}{8^3}\right) - 3n^2 \lg n + 9n^2$$

continue to next slide…

$$T(n) = 64T\left(\frac{n}{8}\right) - n^2 \lg n$$

Continue from previous slide…

$$= 64^3 T\left(\frac{n}{8^3}\right) - 3n^2 \lg n + 9n^2$$

$$= 64^3 \left[64T\left(\frac{n}{8^4}\right) - \left(\frac{n}{8^3}\right)^2 \lg\left(\frac{n}{8^3}\right)\right] - 3n^2 \lg n + 9n^2$$

$$= 64^4 T\left(\frac{n}{8^4}\right) - n^2(\lg n - 9) - 3n^2 \lg n + 9n^2$$

$$= 64^4 T\left(\frac{n}{8^4}\right) - n^2 \lg n + 9n^2 - 3n^2 \lg n + 9n^2$$

$$= 64^4 T\left(\frac{n}{8^4}\right) - 4n^2 \lg n + 18n^2$$

continue to next slide…

The recurrence relation can be generalized as

$$T(n) = 64^k T\left(\frac{n}{8^k}\right) - kn^2 \lg n + 3\left(\frac{k(k-1)}{2}\right)n^2 \qquad \cdots\cdots\cdots eq(1)$$

The recursive call will continue and stop when $\left(\frac{n}{8^k}\right) = 1$. Solving the equality, we have $n = 8^k$ and hence $k = \log_8 n$.

Substituting k into $eq(1)$, we have

$$T(n) = 64^{\log_8 n} T\left(\frac{n}{8^{\log_8 n}}\right) - (\log_8 n)n^2 \lg \mathrm{n} + 3\left(\frac{\log_8 n(\log_8 n-1)}{2}\right)n^2.$$

$$T(n) = (8^2)^{\log_8 n} T\left(\frac{n}{n}\right) - n^2 \log_n n \log_2 n + 3n^2\left(\frac{{\log_8}^2 n - \log_8 n}{2}\right)$$

$$T(n) = (8^2)^{\log_8 n} T\left(\frac{n}{n}\right) - n^2 \log_8 n \log_2 n + 3n^2 \left(\frac{\log_8{}^2 n - \log_8 n}{2}\right)$$

$$T(n) = \left(8^{\log_8 n}\right)^2 T(1) - n^2 \log_8 n \log_2 n + \frac{3n^2}{2} \log_8{}^2 n - \frac{3n^2}{2} \log_8 n$$

$\therefore$ *The running time complexity is* $\boldsymbol{\Theta\left(n^2 \log_8{}^2 n\right)}$