master theorem :-

MT is applicable only to recurrence relations of form

$$T(n) = aT(n/b) + f(n), \quad a \geq 1, \quad b > 1, \quad a, b \rightarrow \text{constants.}$$

$f(n) \rightarrow$ function of 'n'

Ex:

$$\left.\begin{aligned} T(n) &= 10T(n/2) + n^2 \\ T(n) &= 8T(n/4) + n\log n. \end{aligned}\right\}$$

masters theorem :- Let $T(n) = a\,T(n/b) + f(n)$, a, b are constants

$a \geq 1$, $b > 1$

power

Case(i) If $f(n) = O(n^{\log_b a - \epsilon})$ $\epsilon > 0$ then $T(n) = \Theta(n^{\log_b a})$

power

Case(ii): If $f(n) = \Omega(n^{\log_b a + \epsilon})$ $\epsilon > 0$, then $T(n) = \Theta(f(n))$

Case(iii):- If $f(n) = \Theta(n^{\log_b a} \cdot (\log n)^k)$ where k is constants, $k \geq 0$, then $T(n) = (n^{\log_b a} (\log n)^{k+1})$

Dont follow these rules

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

$$\underbrace{f(n) \qquad\qquad n^{\log_b a}}$$

1) which one is greater.

2) Is it greater by polynomial time

$\Theta($greater term$)$

$f(n)$ $\qquad$ $n^{\log_b a}$

n^3 $\quad$ n $\quad$ n^2

$\Theta(n^3)$

$n^{2.81}$ $\quad$ $2^{.81}$ $\quad$ n^2

$\Theta(2^{2.81})$

n^{50} $\quad$ n^{100} $\quad$ n^{50}

$\Theta(n^{100})$.

$\underline{f(n)}$ $\qquad$ $\underline{n^{\log_b a}}$

$n \log n \quad > \quad n$

But it is not polynomial greater $(\log n)$

make them equal

$n \log n$ $\qquad$ $\underline{\underline{n * (\log n)^1}}$ $1 = k$ $= \Theta(n (\log n)^2)$

$= \underline{\underline{\Theta\left(n (\log n)^{k+1}\right)}}$

$f(n)$ $g(n)$

$n(\log n)^{10} > n$

But not polynomial greater.

$n(n \log n)^{10}$

$n(\log n)^{10} = k$

$\Theta(n(\log n)^{11}) \rightarrow k+1$

n^2 $\quad$ n.

n

$n^{2.81}$ $\quad$ n^2

$2^{.81}$

n^3 $\quad$ n^4

n

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

$\log^2 n = \underline{\log \log n}.$

$f(n)$ $\qquad$ $n^{\log_b a}$

$n(\log n)^5$ $\qquad$ n

not polynomial greater.

$n(\log n)^5$ $\qquad$ $n(\log n)^5 = k.$

$\Theta\left(n(\log n)^6\right)$

$f(n)$ $\quad n^{\log_b a}$

$n^2 \log n > n$

$n \log n \rightarrow$ greater by polynomial time.

$\Theta(n^2 \log n)$

$f(n)$	$n^{\log_b a}$
n^3	n^2

n^3 is ~~greater~~ polynomial greater ~~th~~ than n^2.

$$\Theta(n^3)$$

$f(n)$	$g(n)$
n^4	n^5

$\Theta(n^5)$

n^5 is polynomial time greater.

$f(n)$	$g(n)$
$n \log n$	n

$$\Theta\left(n(\log n)(\log n)\right) = \Theta\left(n(\log n)^2\right)$$

multiply with $\log n$

$f(n)$ $n^{\log_b a}$

$n(\log n)^{10} > n$

But it is not polynomial greater.

$\Theta\left(n(\log n)^{10} * \log n\right) = \Theta\left(n(\log n)^{11}\right)$

multiply

$\log n$

$f(n)$ $\quad g(n)$

$n^2(\log n)^0 \quad n^2$

$\left(n^2(\log n)^{0+1}\right)$

$= n^2 \log n$

$f(n) \quad n^{\log_b a}$

$n^3 \quad n^3$

$\Theta(n^3 \log n)$

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

$a = 8$, $b = 2$

$f(n)$ $\qquad n^{\log_b a}$

n^2 $\qquad n^{\log_2 8} = n^3$ → greater by polynomial time

$\Theta(n^3)$.

$f(n)$ $\qquad n^{\log_b a}$

n^2 $\qquad 2^n$ ✓

$\Theta(2^n)$

Gate:

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

$a = 2$

$b = 2$

$n^{\log_b a} = n$

$f(n)$

$\downarrow$

n^2

polynomial greater

n

$\Theta(n^2)$

Gate: $T(n) = 2T(n/2) + n$

$$\begin{array}{ccc} f(n) & & n^{\log_b a} \\ \downarrow & & \downarrow \\ n & = & n \end{array}$$

$$\Theta(n \log n)$$

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

$a = 7 \quad b = 2$

$f(n)$

$\downarrow$

$n^2 \Longleftrightarrow n^{\log_b a}$

$n^{\log_2 7}$

$n^{2.81} \rightarrow 2^{0.81}$ ✓

$\downarrow$

greater by polynomial time.

$$\Theta(n^{2.81})$$

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

$a = 1 \quad b = 2$

$n^{\log_b a}$

$f(n)$

$n^{\log_2 1}$

$\downarrow$

n

1

$$\underline{\underline{\Theta(n)}}$$

$T(n) = 2T(n/2) + n \log n$

$f(n)$

$\downarrow$

$n \log n$

But not greater by polynomial time

$n^{\log_b a}$

$n^{\log_2 2}$

$\downarrow$

n

$(n \log n * \log n)$

$= n(\log n)^2.$

$$T(n) = 8T(n/2) + n^3 \log n$$

$$f(n) \qquad n^{\log_b a}$$

$$\downarrow \qquad \downarrow$$

$$n^3 \log n \qquad n^3$$

$$\cancel{n^3 \log^2 n}$$

$$n^3 (\log n)^2$$

$$T(n) = 2T(n/2) + n.(\log n)^9$$

$$f(n) \qquad\qquad n^{\log_b a}$$

$$n(\log n)^9 \quad \text{———} \quad n$$

$$\left(n(\log n)^9 \log n\right)$$

Dont ask me why?

It is complicated to prove.

$T(n) = 2T(n/2) + n^2 \log n$

$n^2 \log n$ is polynomial greater than n.

$n^3 \log n$ ✓ $\quad n^3$

$n^2 \log n \quad n$

$f(n) \quad\quad n^{\log_b a}$

$\Downarrow \quad\quad \Downarrow$

$n^2 \log n$ ✓ $\quad n$

$\neq \Theta(\cancel{n^2 (\log n)^2})$

greater by polynomial.

$\Theta(n^2 \log n)$

✓ $n \log n$

$n \log n$

$n^3 \log n \quad n^3$

$n^2 \log n \quad n$

$$T(n) = 2^n T(n/2) + n^n \implies \text{Substitution} \implies \text{big answer.}$$

a should be a constant.

MT is not applicable

$$a = 2^n$$
$$b = 2$$
$$n^{\log_b a}$$
$$n^{\log_2 2^n}$$
$$= n^n$$

$$f(n)$$
$$\downarrow$$
$$(n^n)$$

$$\Theta(n^n \log n).$$

$$T(n) = 0.5\,T(n/2) + n$$

MT ✗ applicable.

$a \geq 1$ ✓

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

assume that $n = 2^k$

$T(2^k) = T(2^{k/2}) + c$

assume that $T(2^k) = S(k)$

$S(k) = S(k/2) + c \longrightarrow$ MT is applicable.

$f(k) \quad k^{\log_b a} = k^{\log_2 1}$

$\Downarrow$

$c = c = \Theta(c \log k)$

$$S(k) = \Theta(\log k)$$

$$T(2^k) = S(k)$$

$$S(k) = \Theta(\log k)$$

$$T(2^k) = \Theta(\log k)$$

$$n = 2^k \Rightarrow k = \log n$$

$$T\left(2^{\log_2 n}\right) = \Theta(\log_2 \log_2 n)$$

$$\Rightarrow T(n) = \Theta(\log \log n)$$

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

assume $n = 2^k$

$$T(2^k) = T(2^{k/2}) + c$$

assum $T(2^k) = S(k)$

$$T(2^k) = S(k)$$

$$S(k) = S(k/2) + c$$

$f(k)$ $\Downarrow$ c

$S^{\log_b a}$ $\Downarrow$ c

$\therefore$ $S(k) = \Theta(c \log k)$

$$S(k) = \Theta(\log k)$$

$$\Downarrow$$

$$T(2^k) = \Theta(\log k)$$

$$n = 2^k \Rightarrow k = \log n$$

$$T(2^k) = \Theta(\log k)$$

$$T(n) = \Theta(\log \log n)$$

** $T(n) = 2T(\sqrt{n}) + \log_2 n$

$\Rightarrow$ I have not given base conditions, But in exam base conditions will be given.

assume $n = 2^k \Rightarrow \log n = k$

$T(2^k) = 2T(2^{k/2}) + k$

assume $T(2^k) = S(k)$

$S(k) = 2S(k/2) + k$

$f(k)$ $\Downarrow$ k

$k^{\log_b a}$ $\Downarrow$ k

$S(k) = k(\log k)$

$$S(k) = \Theta(k \log k)$$

$$T(2^k) = S(k)$$

$$T(2^k) = \Theta(k \log k)$$

$$2^k = n \Rightarrow \log n = k$$

$$T(n) = \Theta(\log n \log \log n)$$

$(k \log k)$

$k = \log n$

$(\log n \log \log n).$

$$T(n) = T(\sqrt{n}) + C$$

Substitution

$$= T(n^{1/2}) + C$$

$$= T(n^{1/4}) + C + C$$

$$= T(n^{1/8}) + C + C + C$$

$$= T(n^{1/2^3}) + 3C$$

k time

$$T(n^{1/2^k}) + kC$$

let $T(2) = 2$

given in exams.

$$n^{1/2^k} = 2$$

$$\frac{1}{2^k} \log n = 1$$

$$2^k = \log n$$

$$k = \log\log n$$

$$T(n) = 2 + \log\log n.$$

$$= \Theta(\log\log n)$$

Any problem

- Substitution ✓
- recursion tree ✓
- Master theorem (Some of them)

function:

$A(n) \Rightarrow T(n)$
$\Rightarrow m(n)$

```
{
  if (n <= 1)
    return 1;
  else
    return (n * A(n-1));
}
```

$n * A(n-1)$

$T(n-1)$

$m(n-1)$

$$m(n) = \begin{cases} 0 & \text{if } n \le 1 \\ m(n-1) + 1 & \text{if } n > 1 \end{cases}$$

Recurrence relation for TC

$$T(n) = \begin{cases} O(1) & \text{if } n \le 1 \\ T(n-1) + C; & n > 1 \end{cases}$$

o/p

~~$T(n) * T(n-1)$~~

~~$A(n) * A(n-1)$~~

~~$T(n) * T(n-1)$~~

Recurrence relation for the value, not TC.

$$A(n) = \begin{cases} 1, & n \le 1 \\ n * A(n-1) & n > 1 \end{cases}$$

A(n)
{
if (n ≤ 1) return (n)
else
return (A(n/3) + A(2n/3) + n)
}

RR for value = o/p.

$$v(n) = \begin{cases} n, & n \le 1 \\ v(n/3) + v(2n/3) + n \end{cases}$$

RRTC.

$$T(n) = \begin{cases} C & \text{if } n \le 1 \\ T(n/3) + T(2n/3) + C \end{cases}$$

RR for additions:-

$$add(n) = \begin{cases} 0 & \text{if } n \le 1 \\ add(n/3) + add(2n/3) + 2 \end{cases}$$

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

$T(n)$

$T(n/3)$ $\quad T(n/3/2)$

$T(n/3)$ $\quad T(n/3/2)$

$T(n/3)$ $\quad T(n/3/2)$

$T(n/3)$

$\log_3 n$

$\log_{3/2} n$

$f(n) \Rightarrow T(n)$

```
{
    if (n==0 || n==1)
        return (n)
    else
        return (f(n-1) + f(n-2))
}
```

$$\underbrace{f(n-1)}_{\Downarrow\ T(n-1)} + \underbrace{f(n-2)}_{\Downarrow\ T(n-2)}$$

RR for value -

$$V(n) = \begin{cases} n & ; \text{if } n == 0 \text{ or } n == 1 \\ V(n-1) + V(n-2) & \text{otherwise.} \end{cases}$$

TC :

$$T(n) = \begin{cases} O(1) & \text{if } n=0 \wedge n=1 \\ T(n-1) + T(n-2) + c ; \end{cases}$$

$$c + 2c + 4c + \dots 2^n c = \underline{\underline{O(2^n)}}$$

$$T(n) \quad — \quad c$$

$$T(n-1) \qquad T(n-2) \quad - \; 2c$$

$$T(n-2) \quad T(n-3) \qquad T(n-3) \quad T(n-4) \; - 4c$$

n

$n/2$

$$\underline{\underline{\Omega(2^{n/2})}}$$

$T(n-2)$

$T(n-4)$

$T(n-8)$

n

$n/2$

$\underbrace{2 \quad 4 \quad 8 \quad 6 \ldots n}_{n/2}$

$SC = O(n)$

$$\boxed{2^{n/2} \le T(n) \le 2^n}$$

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

$n == 1 \;||\; n == 0.$

$\cancel{c+2c+4c}$

$c + 2c + 2^2c + 2^3c + \dots 2^n c$

n steps

$c(1 + 2 + 2^2 + \dots 2^n)$

$= O(2^n)$

$T(n)$

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

$T(n-2) \quad T(n-3) \quad T(n-3) \quad T(n-4)$

$\vdots$

$T(1)$

— c

— $2c$

— $3c$

n

$n-2$

$n-4$

$\vdots$

0

$n/2$

$n/2.$

$\cancel{c+c+}$

$c + 2c + 2^2c + \dots 2^{n/2}c$

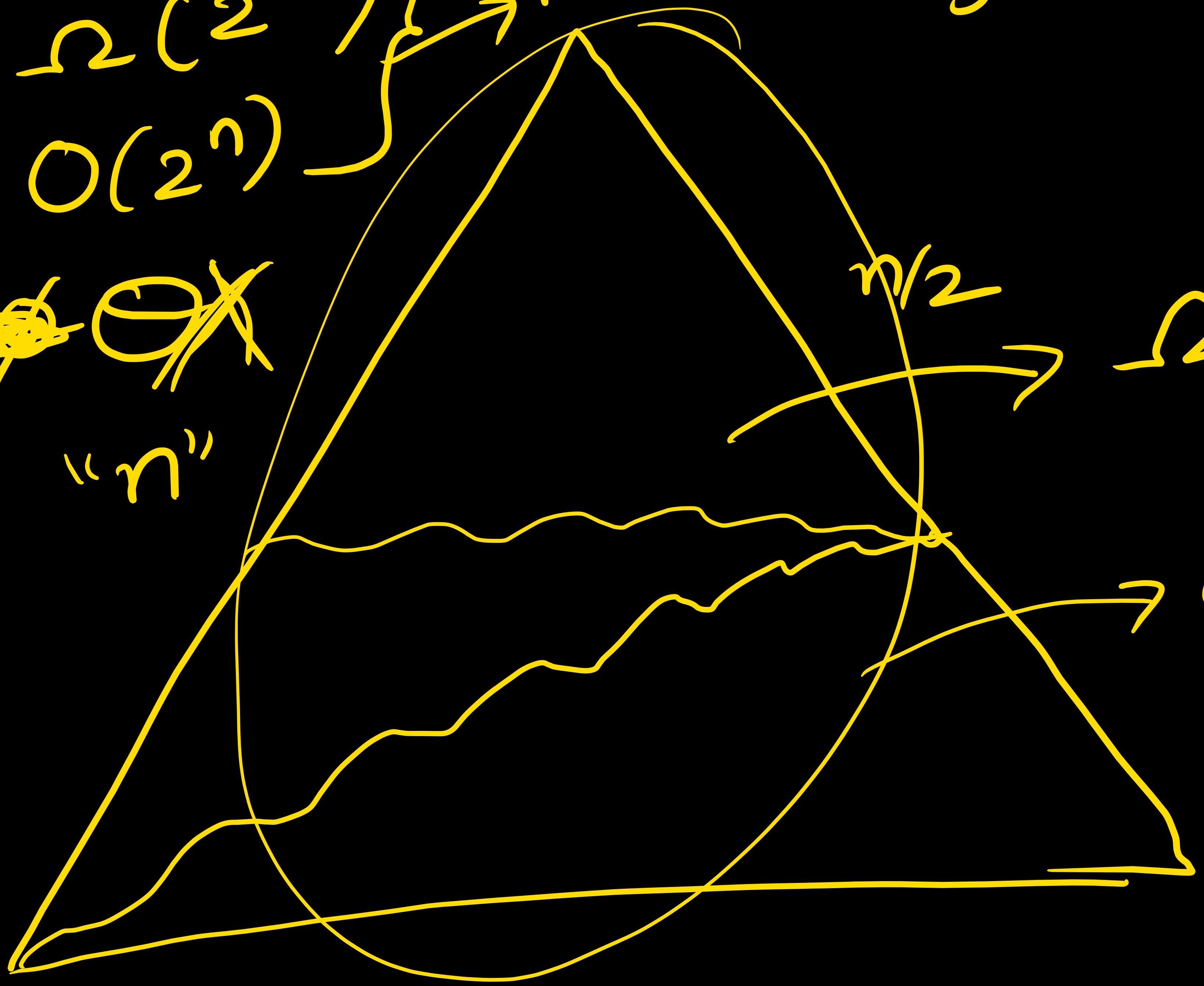
$c(1 + 2 + 2^2 + \dots 2^{n/2})$

$\Omega(2^{n/2})$

$$T(C) = \Omega(2^{n/2}) \Big\} \rightarrow \text{not different by constant.}$$
$$= O(2^n)$$

$\cancel{\neq \Theta}$

"n"



$\rightarrow \Omega(2^{n/2})$

$\rightarrow O(2^n).$