Example - Recursion

$T(n)$ — Test (int n)

{ if (n>0)

{

1 — cout << n;

T(n-1) Test (n-1);

T(n-1) Test (n-1);

}

}

1

$$T(n) = \begin{cases} 1 & n=0 \\ 2T(n-1)+1 & n>0 \end{cases}$$

$$T(n) = 2T(n-1)+1 \quad \text{— (i)}$$

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

$$= 2^2 T(n-2)+2+1 \quad \text{— (ii)}$$

$$= 2^2 T(2(n-3)+1)+2+1$$

$$= 2^3 (T(n-3)+2^2+2+1 \quad \text{— (iii)}$$

Simply

$$= 2^k T(n-k) + \boxed{2^{k-1}+2^{k-2} \ldots +2^2+2+1}$$

Assume $n-k=0 \Rightarrow n=k.$

$$= 2^n T(n-n) + 2^k - 1 \quad =$$

$$= 2^n + 2^k - 1 = 2(2^n)-1$$

$$= 2^{n+1} - 1 = O(2^n)$$

⇒ **Master Therom for decreasing function**

$$T(n) = T(n-1) + 1 = O(n)$$
$$= T(n-1) + n = O(n^2)$$
$$= T(n-1) + \log n = O(n \log n)$$
$$= 2T(n-1) + 1 = O(2^n)$$
$$= 3T(n-1) + 1 = O(3^n)$$
$$= 2T(n-1) + n = O(n2^n)$$

$$\boxed{T(n) = aT(n-b) + f(n)}$$

$a > 0$, $b > 0$ & $f(n) = O(n^k)$

where $k \geq 0$

Cases

1) if $a = 1$ $\quad O(n^{k+1})$ or

$O(n * f(n))$

e.g.

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

Case 2

(2)

if $a > 1$

$$O(\overbrace{n^k}^{f(n)}\, a^{n/b}) \Rightarrow (f(n)\, a^{n/b})$$

e.g.

$$\overset{(a)}{2}T(n - \overset{b}{2}) + \overset{f(n)}{1}$$

$$O(n^0\, 2^{n/2})$$

$$O(2^{n/2})$$

if $a < 1$

$$O(n^k) \quad \text{or} \quad O(f(n))$$