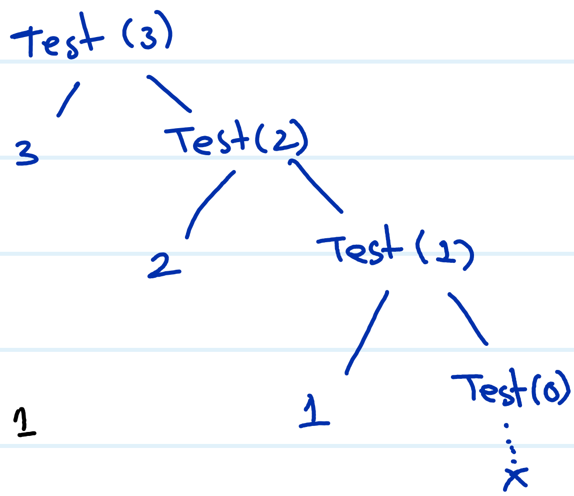
* Recurrence Relation

① Decreasing fn (Recursion Tree)

```
void Test (int n) {
    if (n>0) {                 → 1
        printf ("%d", n);      → 1
        Test (n-1);            → T (n-1)
    }
}
```

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

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



no. of cases $= 3 + 1$

no. of cases of $n = n+1$

$f(n) = O(n)$

$T(n) = T(n-1) + 1$ —① (Backward substitution Method)

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

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

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

Subs. $T(n-1)$ in eqn ①

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

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

$T(n) = T(n-3) + 3$ .... $T(n) = T(n-k) + k$
(continue for k times)

$T(n) = T(n-k) + k$

Assume $n-k=0$ (base condⁿ)

$\therefore k = n$

Subs k with n

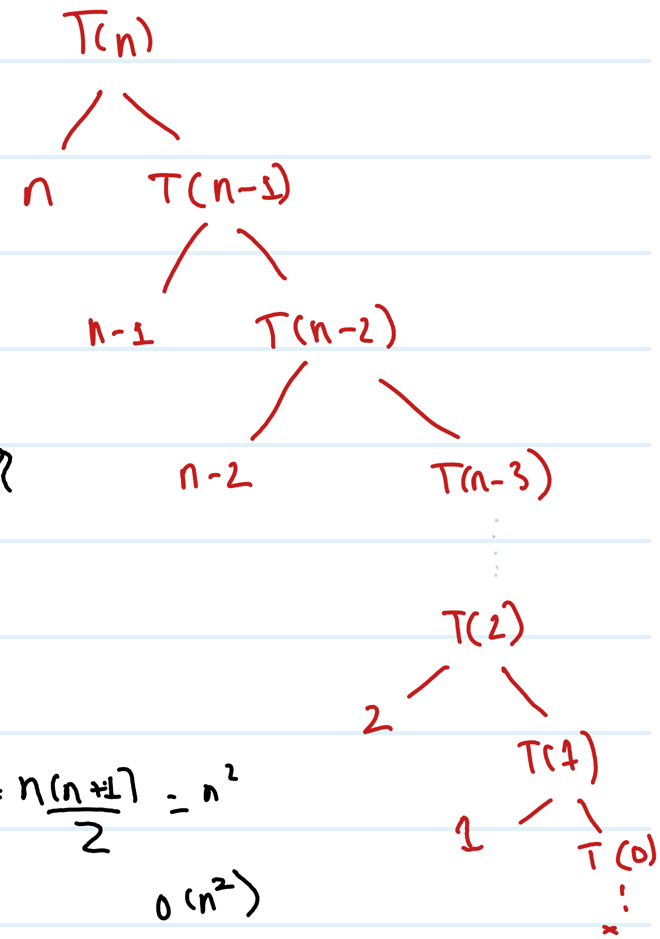
$T(n) = T(n-n) + n$

$T(n) = T(0) + n$

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

3)
```
Void Test(int n) {                  → T(n)
    if (n > 0) {                    → 1
      for (int i=0; i<n; ++i) {     → n+1
          printf("%d", n);          → n
      }
    Test(n-1);                      → T(n-1)
    }
}
```

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



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

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

$0+1+2+\dots n-1+n = \frac{n(n+1)}{2} = n^2$

$O(n^2)$

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

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

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

$\therefore$ Subs $T(n-1)$ in eq. 1

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

$\therefore$ Subs $T(n-2)$ in eq

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

. . . . .

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

Assume $n = k = 0$

$T(n) = T(n-n) + (n(n-1) \dots$

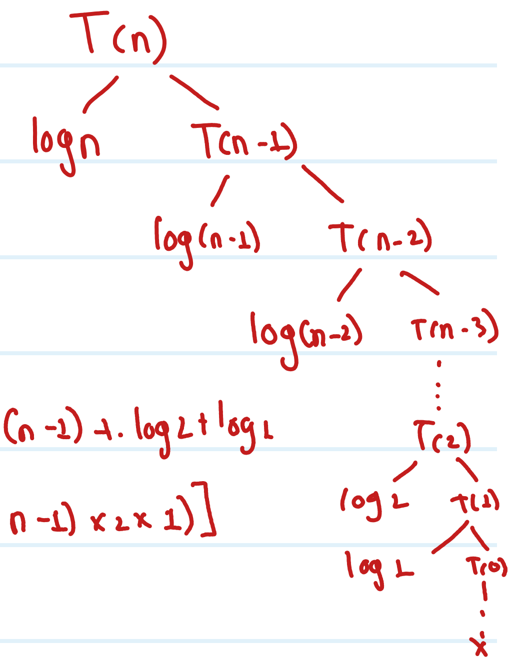
$T(n) = T(0) + 1 + 2 + \dots (n-1)+n$

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

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

```
void Test (int n) {            → T(n)
    if (n>0) {
        for (i=0; i<n; i=i*2) {
            printf("%d", i);   → log n
        }
        Test (n-1);            → T(n-1)
    }
}
```

$T(n) = T(n-1) + \log n$



$T(n) = \log(n-1) + . \log 2 + \log 1$

$= \log[n \times (n-1) \times 2 \times 1]$

$= \log n!$

$= O(n \log n)$

$T(n) = T(n-1) + \log n \quad -①$

$T(n) = T(n-1) + \log n$

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

$T(n-2) = T(n-3) + \log(n-2)$

Subs $T(n-1)$ in Eq ①

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

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

Subs $T(n-2)$ in above eq

$T(n) = T(n-3) + \log(n-2)] + \log(n-1) + \log n$

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

. - - - - . . - - - - - . - - .

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

Assume $n-k=0$, $k=n$

$\therefore T(n) = T(n-n) + \log(n(n-1) \ldots -$

$\therefore T(0) + \log[1 * 2 * 3 \ldots (n-1) * n]$

$\therefore T(n) = 1 + \log n! \quad [O(n!) = O(n^n)]$

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

* Imp

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

$T(n) = T(n-1) + n \longrightarrow O(n^2)$

$T(n) = T(n-1) + \log n \longrightarrow O(n\log n)$

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

$T(n) = T(n-2) + 1 \longrightarrow \frac{n}{2} \quad O(n)$

$T(n) = T(n-100) + n \longrightarrow O(n^2)$

Recurrence Relation for Decreasing Function(Observations for with coefficients)

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

* Master's Theorem

$$T(n) = \begin{cases} c & ; n \le 1 \\ aT(n-b) + f(n) & ; n > 1 \end{cases}$$

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

$T(n) = O(n^k)$, if $a<1$

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

$= O\left(n^k \cdot a^{\frac{n}{b}}\right)$ if $a>1$

② Recurrence Relation (Dividing fn)

```
Test (int n) {          → T(n)
  if (n>1) {
     printf("%d", n);   → 1
    Test (n/2)          → T(n/2)
  }
}
```

Recursion Tree

$T(n/2)$ → 1, $T(n/2)$

$T(n/2)$ → 1, $T(n/2^2)$

⋮

$T(n/2^k)$

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

$\therefore\ n = 2^k$

$\therefore\ \log_2 n = k$

$\therefore\ O(\log n)$

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

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

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

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

Subs $T(n/2)$ in eq ①

$T(n) = [T(n/2^2) + 1] + 1$

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

Subs $T(n/2^2)$ in

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

. . . — — — — -

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

$n/2^k = 1$ (base condition)

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

$T(n) = T(1) + \log n$

$T(n) = T(1) + \log n$

$O(\log n)$

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

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

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

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

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

$$\vdots$$

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

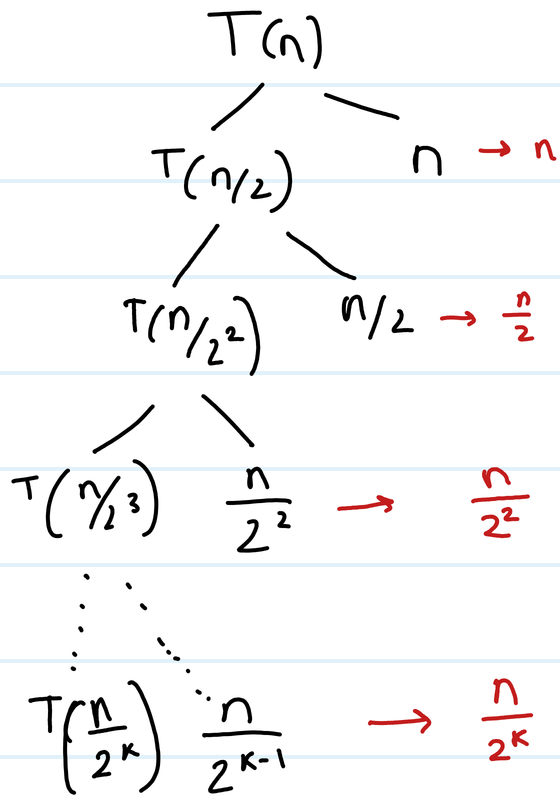
$$\frac{n}{2^k} = 1$$

$$\therefore \quad n = 2^k \;\&\; k = \log n$$

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

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

$$O(n)$$



$$T(n) = n + \frac{n}{2} + \frac{n}{2^2} + \dots \frac{n}{2^k}$$

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

$$= n \sum_{i=0}^{k} \frac{1}{2^i} = 1$$

$$= n \times 1$$

$$T(n) = n$$

$$O(n)$$

Master Theorem for Dividing fn

① $\log_b a$ $\quad T(n) = aT(n/b) + f(n)$

② k $\quad a \geq 1,\ b > 1 \quad f(n) = \Theta(n^k \log^p n)$

Case I: If $\log_b a > k$ then $\Theta(n^{\log_b a})$

Case II: If $\log_b a = k$

↳ if $p > -1$ $\quad \Theta(n^k \log^{p+1} n)$

if $p = -1$ $\quad \Theta(n^k \log\log n)$

if $p < -1$ $\quad \Theta(n^k)$

Case III: if $\log_b a < k$

↳ if $p \geq 0$ $\quad \Theta(n^k \log^p n)$

if $p < 0$ $\quad O(n^k)$

Examples:

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

$a = 2,\ b = 2,\ f(n) = \Theta(1)$

↳ $\Theta(n^0 \log^0 n)$

$k = 0,\ p = 0$

$\log_2 2 = 1,\ k = 0$

↳ Case I $\Rightarrow \overset{1}{\log_b a} > \overset{0}{k}$

↳ $\Theta(n^1)$

$T(n) = T(\sqrt{n}) + 1$

$\boxed{N = 2^m}$

$T(2^m) = T(2^{\frac{m}{2}}) + 1$

$S(m) = S(\frac{m}{2}) + 1$

$a = 1\ b = 2,\ f(n) = 1$

$\log_b a = 0,\ \underline{k = 0},\ p = 0$

↳ case II

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

$m = \log n$

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

② $T(n) = 4T(n/2) + n$

$a = 4$, $b = 2$, $f(n) = \Theta(n^1 \log^0 n)$

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

$k = 1$, $p = 0$

$\log_b a = \log_2 4 > k$ (2) (1)

Case I: $\Theta(n^2)$

③ $T(n) = 2T(\frac{n}{2}) + n$

$\log_2 2 = 1$, $k = 1$, $p = 0$

$\Theta(n \log n)$

④ $T(n) = 4T(\frac{n}{2}) + n^2$

$\log_2 4 = 2$, $k = 2$, $p = 0$

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

⑤ $T(n) = 4T(n/2) + n^2 \log n$

$\log_2 4 = 2$, $k = 2$

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

(6) $2T(n/2) + \frac{n}{\log n}$

$\log_2 2 = 1$, $k = 1$, $p = -1$

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

(7) $2T(n/2) + \frac{n}{\log^2 n}$

$\log_2 2 = 1$, $k = 1$, $p = -2$

$\Theta(n^k)$

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

$\log_2 1 = 0 < k = 2$, $p = 0$

$\Theta(n^2)$

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

$\log_2 1 = 0 < k = 2$, $p = 1$

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

(10) $T(n) = 4T(n/2) + \frac{n^3}{\log n}$

$\log_2 2 = 2 < k = 3$

$\Theta(n^3)$

Root Function

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

$$T(n) = T(\sqrt{n}) + 1$$

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

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

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

$$\vdots$$

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

Assume $n = 2^m$

$$T(2^m) = T(2^{\frac{m}{2^K}}) = T(2^1)$$

$$\therefore \quad \frac{m}{2^K} = 1$$

$$\therefore \quad m = 2^K \quad \& \quad K = \log_2 m$$

$$\therefore \quad n = 2^m \quad \& \quad m = \log_2 n$$

$$\therefore \quad K = \log \log_2 n$$

$$\therefore \quad \Theta(\log \log_2 n)$$

Forward Substitution Method

$T(N) = T(N-1) + N$

$T(1) = T(0) + 1 = 1$

$T(2) = T(1) + 2 = 1 + 2 = 3$

$T(3) = T(2) + 3 = 1+2+3$

$T(4) = T(3) + 4 = 1+2+3+4$

. — — — — .

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

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

(i) Let $T(1) = 1$

$$T(1) = \frac{1(2)}{2}$$

$T(1) = 1$

→ Check for 1, 2, 3, 4, 5