Time Comp. Recursion → Recurrence

Ex. Factorial

```
long Fact (int n){
  if(n==0)
    return 1;

  return n * Fact(n-1);
}
```

$\underbrace{n *}_{} \underbrace{Fact(n-1)}_{T(n-1)}$

Rec. Relation $T(n) = \begin{cases} d, & n = 0 \\ T(n-1) + c, & n > 0 \end{cases}$

$T(n) = T(\cancel{n-1}) + c$

$T(\cancel{n-1}) = T(\cancel{n-2}) + c$

$T(\cancel{n-2}) = T(\cancel{n-3}) + c$

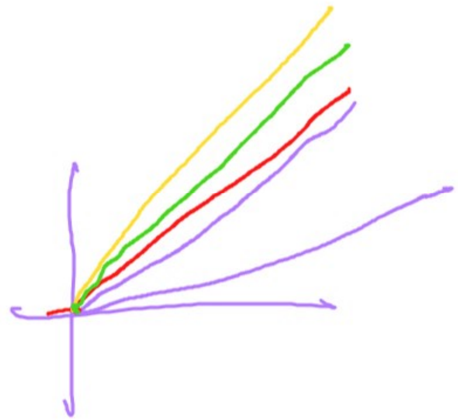
$\vdots$

$T(1) = T(\cancel{0}) + c$

$T(\cancel{0}) = d$

$T(n) = nc + d$

$= O(n)$



```
MergeSort(A, L, H){
   if(L < H)
      middle = (L+H)/2

   MergeSort(A, L, middle);
   MergeSort(A, middle+1, H);
   Merge(A, L, H);
}
```

R.R.

$$T(n) = \begin{cases} d, & n=1 \\ 2T\left(\frac{n}{2}\right) + n, & n>1 \end{cases}$$

A 0 ..... n-1

(A(0, n)

$T(n) = 2T(\frac{n}{2}) + n$ — (1)

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

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

$T(\frac{n}{2^3}) = 2T(\frac{n}{2^4}) + \frac{n}{2^3}$ — (4)

⋮

Now (2) in (1)

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

$= 2^2T(\frac{n}{2^2}) + 2n$ — (5)

Sub. (3) in (5)

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

$= 2^3T(\frac{n}{2^3}) + 3n$ — (6)

After k of steps.

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

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

$n = 2^k$

$k = \log n$

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

$= n.d + n\log n$

$= O(n\log n)$

$$T(n) = \begin{cases} d & , n=1 \\ 2T(\frac{n}{2}) + 10 & , n>1 \end{cases}$$

$T(n) = 2T(\frac{n}{2}) + 10$ — 1

$T(\frac{n}{2}) = 2T(\frac{n}{2^2}) + 10$ — 2

$T(\frac{n}{2^2}) = 2T(\frac{n}{2^3}) + 10$ — 3

Sub. 2 in 1

$T(n) = 2[2T(\frac{n}{2^2}) + 10] + 10$

$= 2^2 T(\frac{n}{2^2}) + 2^1 \cdot 10 + 2^0 10$ — 4

Sub 3 in 4

$T(n) = 2^2[2T(\frac{n}{2^3}) + 10] + 2^1 \cdot 10 + 2^0 \cdot 10$

$= 2^3 T(\frac{n}{2^3}) + 2^2 \cdot 10 + 2^1 \cdot 10 + 2^0 \cdot 10$

at K^{th} step

$T(n) = 2^K T(\frac{n}{2^K}) + \underbrace{2^{K-1} \cdot 10 + 2^{K-2} \cdot 10 + \dots + 2 \cdot 10 + 2^0 \cdot 10}_{10 \cdot (2^{K-1} + 2^{K-2} + \dots + 2^1 + 2^0)}$

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

$n = 2^K$

$\frac{2^K - 1}{2-1} = (2^K - 1) \cdot 10$

$\frac{r^n - 1}{r-1}$

$T(n) = n T(1) + (n-1) \cdot 10$

$= d \cdot n + 10n - 10$

$= O(n)$

$T(n) = 2T(n-1) + c$ — (1)

$T(n-1) = 2T(n-2) + c$ — (2)

$T(n-2) = 2T(n-3) + c$ — (3)

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

$= 2^2 T(n-2) + 2c + c$ — (4)

$T(n) = 2^2[2T(n-3) + c] + 2c + c$

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

after k^{th} step.

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

$n - k = 1$ will stop

$n = k + 1 \longrightarrow k = n - 1$

$T(n) = 2^{n-1} T(1) + 2^{n-2} \cdot c + 2^{n-3} \cdot c + \dots + c$

Assume n is very large

$n \approx n-1$

$T(n) = 2^n T(0) + \underbrace{2^{n-1} \cdot c + 2^{n-2} \cdot c + \dots + c}$

$= d \cdot 2^n + (n-1) \cdot c$

$= O(2^n)$