Recursion – A f^n calling itself. $T(n)$

$A(n) \rightarrow T(n)$.

$A(n/2)$ $T(n/2)$

```
A(int n)
{ if(n>1) return A(n/2) + A(n/2);
  return 1;
}
```

1 $\quad T(n/2)$ $\quad T(n/2)$ $\quad \rightarrow 1$

$$T(n) = \begin{cases} 1 + 2T(n/2), & n>1 \\ 1, & n=1 \end{cases} \rightarrow \text{Recurrence Relation.}$$

```
A(int n) {
    if(n>1) return A(n-1);
    return 1;
}
```

$\rightarrow 1$

Let, $A \longrightarrow T(n)$ $\quad A(n-1) \rightarrow T(n-1)$

$1 + T(n-1)$

$$T(n) = \begin{cases} 1 + T(n-1), & n>1 \\ 1, & n=1 \end{cases} \quad \text{Recurrence Relat}^n.$$

Back Substitution Method –

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

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

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

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

$\vdots$

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

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

$T(1) = 1$

$T(n) + \cancel{T(n-1)} + \cancel{T(n-2)} + \cdots + \cancel{T(3)} + \cancel{T(2)} + \cancel{T(1)} =$

$1 + \cancel{T(n-1)} + 1 + \cancel{T(n-2)} + 1 + \cancel{T(n-3)} + \cdots + 1 + \cancel{T(2)} + 1 + \cancel{T(1)} + 1$

$\Rightarrow T(n) = n \times 1 = n = O(n).$

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

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

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

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

$\vdots$

$\vdots$

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

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

$T(1) = 1$

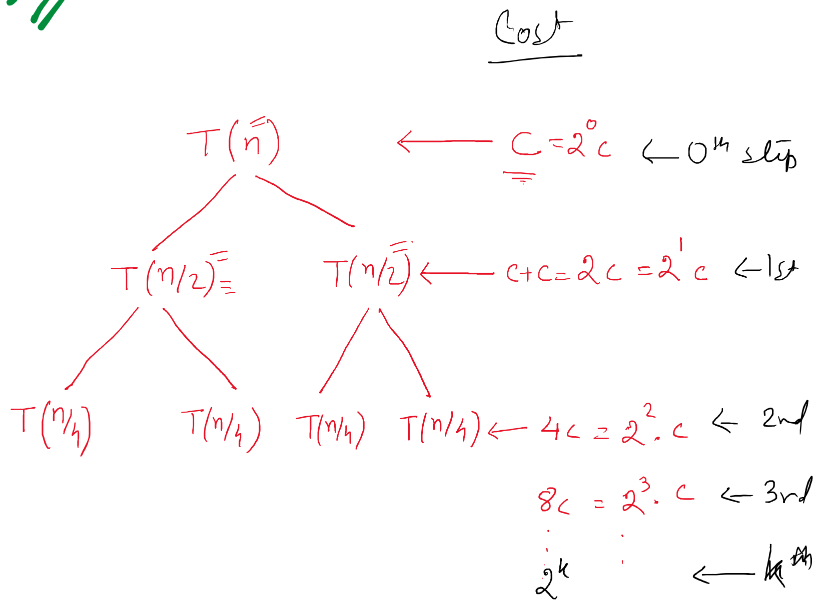
$$T(n) + \cancel{T(n-1)} + \cancel{T(n-2)} + \cdots + \cancel{T(3)} + \cancel{T(2)} + \cancel{T(1)} = n + \cancel{T(n-1)} + (n-1) + \cancel{T(n-2)} + (n-2) + \cancel{T(n-3)} + \cdots + 3 + \cancel{T(2)} + 2 + \cancel{T(1)} + 1 =$$

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

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

Recursion Tree Method

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



k

$$c\left(\underbrace{2^0 + 2^1 + 2^2 + \cdots + 2^k}_{\text{GP series sum}}\right) = \frac{a(r^n - 1)}{r-1} = \frac{1 \times (2^{k+1} - 1)}{2-1}$$

$$= (2^{k+1} - 1) \times c$$

$$= 2^{\log_2 n} \times c$$

$$= n^{\log_2 2} . c = nc = O(n)$$

$k+1 \approx k \approx \log_2 n$

$2^{\log_2 n} = n^{\log_2 2}$

$N = 2^k$

$N > 2^k$

$N < 2^k$

$2^k <= n \Rightarrow \log_2(2^k) <= \log_2 n$

$\Rightarrow k \times 1 <= \log_2 n$

$2^k . c$

$2^0 \quad 2^1 \quad 2^2 \quad \cdots \quad 2^k \quad 2^{k+1}$

$2^k = n$, n, $2^{k+1} \to n$

$$c^{\log_a b} = b^{\log_a c}$$