GT - Part IV

Complete Course on Algorithm for GATE - CS & IT

Subbarao Lingamgunta • Lesson 16 • Dec 20, 2022

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

RR- No. of multiplication

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

RR- value

$$V(n) = \begin{cases} 1 & \text{if } n \le 1 \\ n * V(n-1) & \text{if } n > 1 \end{cases}$$

RR- Time

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

Recurrence Relation solving

1. Substitution method
2. Recursive Tree method
3. Master theorem

Substitution Method

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

const

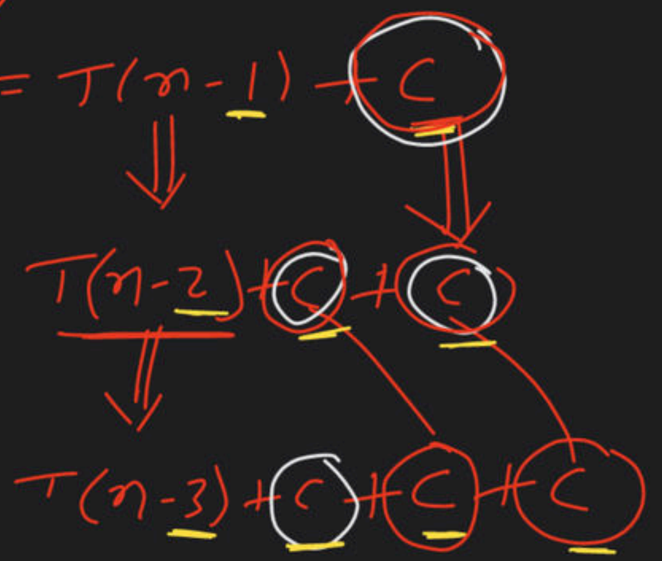
$T(50) = T(49) + c$

$T(200) = T(199) + c$

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

$n - k = 1$

$n - 1 = k$



k times

$n-1$

$= T(n-k) + k \cdot c$

$= T[n-(n-1)] + (n-1) \cdot c$

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

$= O(1) + (n-1) \cdot c = O(n)$

$\Omega(n)$

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n-1) * n & \text{if } n > 1 \end{cases}$$

$T(50) = T(49) * 50$

$T(200) = T(199) * 200$

$T(2) = T(1) * 2$

$T(1) = 1$

$n - K = 1$

$K = n - 1$

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

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

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

100

$= T(n-100) * (n-99) * (n-98) * \cdots * (n-0)$

K times

$n-1$

$= T(n-K) * (n-(K-1)) * (n-(K-2)) * \cdots * (n-0)$

$= T(n-(n-1)) * (n-[(n-1)-1]) * (n-(n-1-2)) * \cdots * n$

$= T(1) * 2 * 3 * 4 * 5 * 6 * \cdots * n-1 * n$

$= 1 * 2 * 3 * 4 * \cdots * n = n! = O(n^n)$

$= \Omega(2^n)$

Fibinocci Series

n	0	1	2	3	4	5	6	7	8	9	10
fib(n)	0	1	1	2	3	5	8	13	21	34	55

fib(soup) = ٢٠٥ + ١٠٣٠٢٠٥

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

Value

Time

Additi

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