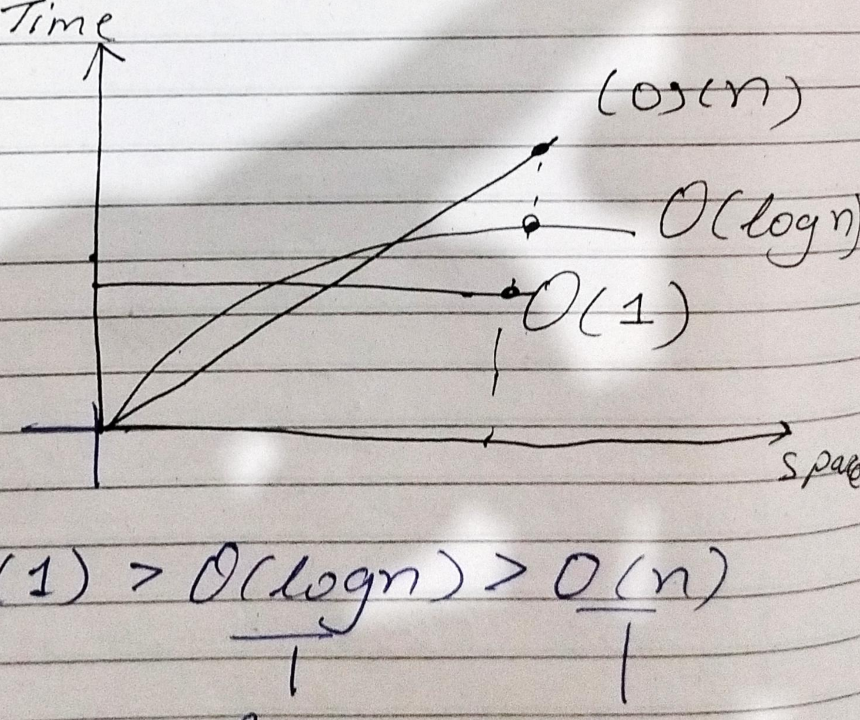
Time and Space Complexity

☆ Time complexity :- Function that give the relationship about, How the time will grow as the input grows.



$O(1) > O(\log n) > O(n)$

Eg → Binary Search > Linear Search

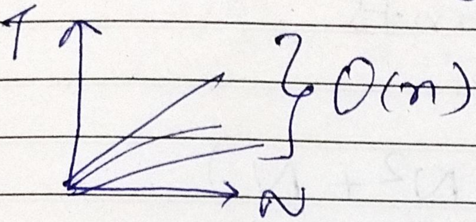
Better

What do we consider when taking about complexity.

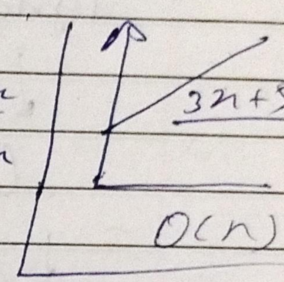
(1) Always look for worst case complexity

(2) Alway look at complexity for large /∞ data

(3)



$y = x$
$y = 2x$
$y = 4x$



(*) Even tho value of actual time ~~value of~~ is different they are showing linearity.

(*) We don't care about actual time

(*) This is why, we ignore constants.

(4) $O(N^3 + \log N)$

(*) => from point (2)

1mil → $O((1mil)^3 + \log(1mil))$

$= (1mil)^3 + 6$

(*) very small

That is why we ignore less dominating terms.

Ex → $O(3N^3 + 4N^2 + 5N + 6)$

↓

ignore constants

↓

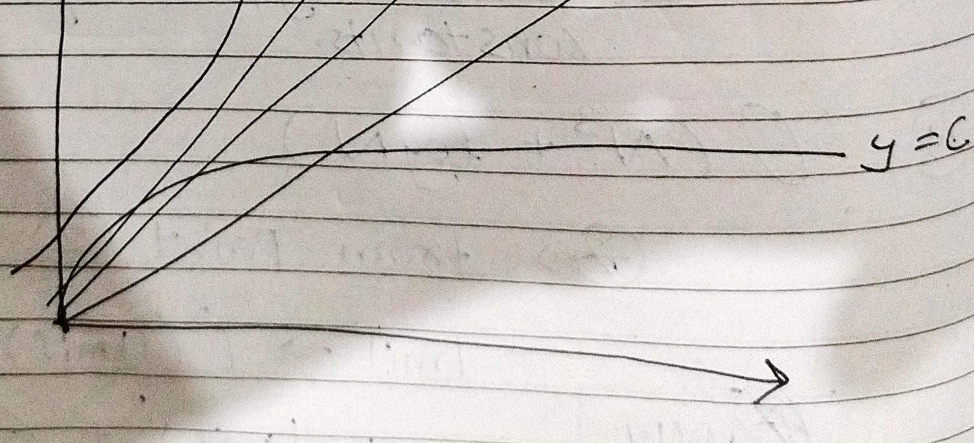
$O(N^3 + N^2 + N)$

↓

ignore less dominating terms

↓

$\boxed{O(N^3)}$ ← final complexity



Notations

① Big-O notation

└ upper bound
(worst case)

$$\boxed{\lim_{n \to \infty} \frac{f(n)}{g(n)} < \infty}$$ for

$f(n) \leq O(g(n))$

② Big-Omega notation

└ lower bound
(Best case)

$$\boxed{\lim_{n \to \infty} \frac{f(n)}{g(n)} > 0}$$

Big
③ -Theta-notation :-

⇒ Both upper bound & lower bound.

$$\boxed{0 < \lim_{n \to \infty} \frac{f(n)}{g(n)} < \infty}$$

Date
Page

(4) little O notation

∟ loose upper bound.

Big O	little O (stronger statement)
$f = O(g)$	$f = o(g)$
$f \leq g$	$f < g$

(5) little - Omega notation

∟ Loose lower bound

Space Complexity

⇒ $\boxed{\text{input space}}$ + $\boxed{\text{Auxillary Space}}$

Extra Space or a temporarily same taken by the algorithm

Date ____
Page ____

Ques) Find time complexity :-

$$\text{for } (i = 1;\ i \le N;\ \cancel{i{+}{+}})$$
{
for (j = 1 ; j ≤ K ; j++)
{
// Statement → time (t)
}
i = i + k
}

K × t time

Soln

$\cancel{N(N) = O(N^2)}$

O (K+ time * no of times the outer loop is running

?

$\cancel{O(Kt}$

$i = 1, 1+k, 1+2k, 1+3k, \ldots, 1+xk$

$1 + xk \le N$

$xk \le N - 1$

$$\boxed{x \le \frac{N-1}{k}}$$

=

$i = 1, 1+k, 1+2k, 1+3k, 1+4k, N$

$i = \cancel{N + k[1+2+3+ \ldots N]}$

$= \cancel{N + k\left[\frac{N(N+1)}{2}\right]}$

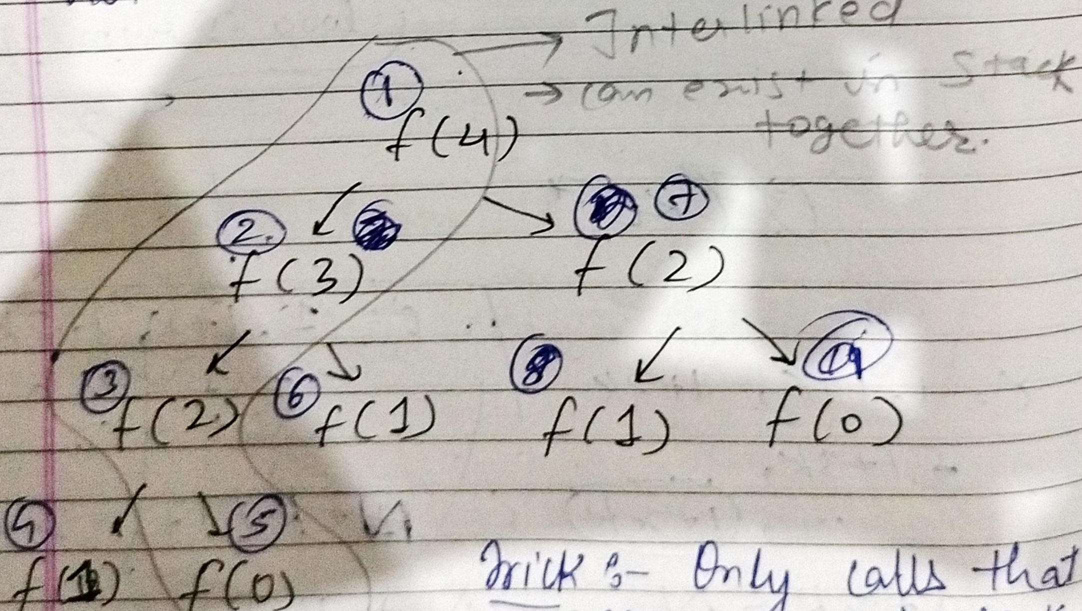
=

$$O\left(Kt * \frac{(N-1)}{K}\right)$$

$$\boxed{O(N)}$$

Recursive Algorithms.

* Recursive Algorithms take some memory in the stack. Hence, ~~there~~ their space complexity is not constant.



Trick :- Only calls that are interlink will be at in the stack at same

(1) Divide & conquer Recurrences

Form :-

$$T(x) = a_1 T(b_1 x + \varepsilon_1(n)) + a_2 T(b_2 x + \varepsilon_2(n)) + \cdots + a_k T(b_k n + \varepsilon_k(n)) + g(x)$$

Same

for $x \geq x_0$

$\uparrow$ Some constant.

$[f(N) = f(\frac{N}{2}) + C]$ - binary Search.

$a_1 = 1$ $\quad$ $g(n) = C$

$b_1 = 1/2$

$\varepsilon_1(n) = 0$

So,

Space Complexity = Heigh of tree.
(Path)

So for $f(4) \rightarrow O(4)$

→ finally for fibonacci num-ber

$$\boxed{\text{Space Complexity} = O(n)}$$

④ Types of recursions:

① linear	② Divide & conquer.
$f(N) = f(N-1) + f(N-2)$	$f(N) = O(1) + f(N/2)$

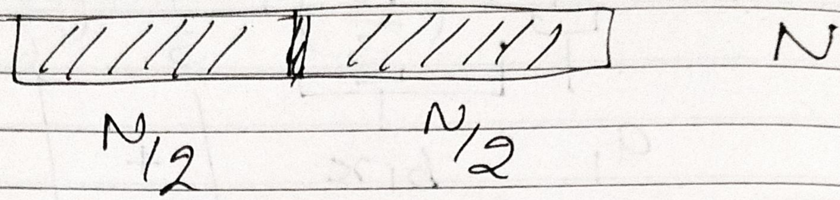
$$T(N) = \underbrace{9}_{a_1} \, T\Big(\underbrace{\tfrac{N}{3}}_{b_1 x}\Big) + \underbrace{\tfrac{4}{3}}_{a_2} \, T\Big(\underbrace{\tfrac{5N}{6}}_{b_2 x}\Big) + \underbrace{4N^3}_{g(N)}$$

$$T(N) = \underbrace{2}_{a_1} \, T\Big(\underbrace{\tfrac{N}{2}}_{b_1}\Big) + \underbrace{(N-1)}_{g(N)}$$

when you get ans from this

+

what you are doing (after) with that answer eg ⇒ Merge Sort



$$T(N) = T\left(\frac{N}{2}\right) + T\left(\frac{N}{2}\right) + \underbrace{(N-1)}_{\text{merge them}}$$

$$T(N) = \underbrace{2T\left(\frac{N}{2}\right)}_{a,\ b,\ N} + \underbrace{(N-1)}_{g(N)}$$

How to actually solve to get complexity....

① Plug & chug

$$F(N) = F(N/2) + C$$

② Master's Theorm

③ Akra Bazzi (1996)

(1) Akra Bazzi - for divide & conquer

Akra Bazzi :-

$$T(x) = \theta\left(x^p + x^p \int \frac{g(u)}{u^{p+1}} du\right)$$

$$\boxed{T(x) = \theta\left(x^p + x^p \int_1^x \frac{g(u)}{u^{p+1}} \cdot du\right)}$$

(1) $\theta \rightarrow$ lower bound

(2) $g(u) \longrightarrow$ our $\rightarrow g(N)$ in recurrence.

(3) What is P?

$$a_1 b_1 + a_2 b_2 + \cdots = 1$$

$$\boxed{\sum_{i=1}^{K} a_i b_i^p = 1}$$

Ex → $T(N) = T\left(\frac{N}{2}\right) + C$

Ex → $T(N) = 2T\left(\frac{N}{2}\right) + (N-1)$

$a_1 = 2$ $\quad$ $g(n) = n-1$

$b_1 = 1/2$

So

$$\sum_{i=1}^{k} a_i (b_i)^p = 1$$

$$2 * \left(\frac{1}{2}\right)^p = 1 \quad \left(\because \sum_{i=1}^{k} a_i (b_i)^p = 1\right)$$

from this $\boxed{p = 1}$ (Hit and trial)

* Now put p in the Akra-Bazzi formula.

$$= \theta\left(x + x\int_1^x \left(\frac{u-1}{u^2}\right) \cdot du\right)$$

u^{-2}

$\frac{u^{-2+1}}{-2+1}$

$\frac{u^{-1}}{-1}$

$$= \theta\left(x + x\int_1^x \left(\frac{u}{u^2} - \frac{1}{u^2}\right) \cdot du\right)$$

$$= \theta\left(x + x\left[\log u + \frac{1}{u}\right]_1^x\right)$$

$$= \theta\left(x + x\left[\log x + \frac{1}{x}\right] - (\log(1) + 1)\right)$$

0

$$= \theta\left(x + x\log x + 1 - x\right)$$

$1 - \cancel{x}$

$O(\cancel{x} + x\log x + \cancel{1})$

$O(\cancel{x} + x\log x)$ $+1$ (∵ Ignore constants)

$O(x\log x + 1)$ (∵ Ignore constants)

$O(x\log x)$

Q. $T(N) = \underset{a_1}{2}T\left(\frac{N}{\underset{b_1}{2}}\right) + \underset{a_2}{\frac{8}{9}}\, T\left(\underset{b_2}{\frac{3N}{4}}\right) + \frac{N^2}{9(4)}$

$$\sum_{i=1}^{k} a_i (b_i)^p = 1$$

$$2\left(\frac{1}{2}\right)^p + \frac{8}{9}\left(\frac{3}{4}\right)^p = 1$$

Put $p = 2$

$$2 \times \frac{1}{4} + \frac{8}{9} \times \frac{9}{16} = 1$$

$$\frac{1}{2} + \frac{1}{2} = 1$$

Date
Page

$\frac{2}{2} = 1$

$1 = 1$

So $\boxed{p = 2}$

Now, Applying Akra Bazzi formula.

$$= \theta\left(x^2 + x^2 \int_1^x \frac{u^2}{u^3} \cdot du\right)$$

$$= \theta\left(x^2 + x^2 [(\log u)]_1^x\right)$$

$$= \theta\left(x^2 + x^2 \log x - \cancel{\log(1)}^{0}\right)$$

$$= \theta\left(x^2 + x^2 \log x\right)$$

$$= \cancel{\theta(x^2(1+\log x))}$$

$$= \theta\left(x^2 + x^2 \log x\right)$$

$$= \theta\left(x^2 \log x\right)$$

* If you can't find (P) ??

$$T(N) = 3T\left(\frac{n}{3}\right) + 4T\left(\frac{n}{4}\right) + n^2$$

$$\cancel{T(N)} = \left\{ \sum_{i=1}^{K} a_i (b_i)^P = 1 \right\}$$

$$\Rightarrow \quad 3 \times \left(\frac{1}{3}\right)^P + 4\left(\frac{1}{4}\right)^P = 1$$

let $P = 1$

$$1 + 1 = 2$$

$$2 \neq 1$$

$P > 1$ $\quad$ $2 > 1$

$\Rightarrow$ Increase the denominator

let $P = 2$.

$$\frac{1}{3} + \frac{1}{4} = 1$$

$$\frac{7}{12} \neq 1$$

$P < 2$ $\quad$ $0.58 < 1$

Hence $p > 1$ && $p < 2$

Note:-

when :- $p <$ power of $(g(n))$

then ans $= g(n)$
$\to \Theta(n^2)$

So, $p < 2$

Complexity $= O(n^2) = g(n)$

How :- $\Rightarrow$

use Akra-Bazzi formula without p.

$2-p-1$
$1-p$

$$\Theta\left(x^p + x^p \int_1^x \frac{u^2}{u^{p+1}} \cdot du\right)$$

$$\Theta\left(x^p + x^p \int_1^x u^{1-p} \cdot du\right)$$

$$\Theta\left(x^p + x^p \left[\frac{u^{2-p}}{2-p}\right]_1^x\right)$$

$$x^p\left(1-\frac{1}{2-p}\right) = x^p\left(\frac{2-p-1}{2-p}\right)$$

$$O\left(x^p + \frac{x^p}{2-p}\left[x^{2-p} - 1\right]\right)$$

$$O\left(x^p + \frac{x^{2-\cancel{p}+\cancel{p}}}{2-p} - \frac{x^p}{2-p}\right)$$

$$O\left(\frac{\cancel{x^p}}{\cancel{2-p}}\left(\frac{1-p}{2-p}\right)x^p + \frac{x^2}{2-p}\right)$$

Ignore constents.

$$O(x^p + x^2)$$

$O(x^2)$

Ans

We know that

P is less than 2

So

x^2 is more dominating

linear recurrences.

→ Solving linear Recurrence

Ex → $F(N) = F(N-1) + F(N-2)$

Form :→ $f(x) = a_1 f(x-1) +$

$a_2 f(x-2) + a_3 f(x-3) +$

$\ldots\ldots + a_n f(x-n)$

$$f(n) = \sum_{i=1}^{n} a_i (n-i)$$

for a_i and n is fixed

n = order of reccurence.

Soln for fibonacci

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

$f(n-2)$ —— ①

Steps

① Put $f(n) = \alpha^n$ for some constant α.

$\Rightarrow \alpha^n = \alpha^{n-1} + \alpha^{n-2}$

characteristic eqn of recurrence $\alpha^n - \alpha^{n-1} - \alpha^{n-2} = 0$

÷ by α^{n-2}

$\alpha^2 - \alpha - 1 = 0$

$\frac{\alpha^n}{\alpha^{n-2}}$

α^{n-n+2}

α^2

$\alpha^2 - \alpha - 1 = 0$

② finds roots

$x^2 - x - 1 = 0$

$\alpha = \frac{\cancel{-b} \pm \sqrt{1-4(-1)}}{2}$

$1 \times 1 = 1$

$- + = -1$

$\alpha = \frac{1 \pm \sqrt{1+4}}{2}$

$-b \pm$

$\sqrt{4} = 2$

$\sqrt{9} = 3$

$\alpha = \frac{1 \pm \sqrt{5}}{2}$

$\sqrt{5} = 2.25$

$\alpha = \frac{1+\sqrt{5}}{2}, \frac{1-\sqrt{5}}{2}$

$\alpha = \frac{1+2.25}{2}, \frac{1-2.25}{2}$ -1.15

$\alpha = \frac{3.25}{2} \Rightarrow 1.62, -0.57$

$\alpha_1 = 1.62$, ~~-0.57~~

$\alpha_2 = -0.57$

(2) $f(n) = C_1 \alpha_1^n + C_2 \alpha_2^n$

is a soln for fibonacci

$= C_1 \left(\frac{1+\sqrt{5}}{2}\right)^n + C_2 \left(\frac{1-\sqrt{5}}{2}\right)^n$ —— (2)

(3) Fact :- No of roots = No of answer

Here α_1, α_2 are 2 roots
So, 2 answers already

$f(0) = 0$ & $f(1) = 1$

So,

$f(0) = c_1 + c_2 = 0$

$$\boxed{c_1 = -c_2} \quad \text{—— } (3)$$

$f(1) = c_1\left(\frac{1+\sqrt{5}}{2}\right) + c_2\left(\frac{1-\sqrt{5}}{2}\right) = 1$

$= c_1(1+\sqrt{5}) + c_2(1-\sqrt{5}) = 2$

$= c_1 + c_1\sqrt{5} + c_2 - c_2\sqrt{5} = 2$

$= c_1 + c_1\sqrt{5} + c_2 - c_2\sqrt{5} = 2$

from (3) $c_1 = -c_2$

$= \cancel{-c_2} - c_2\sqrt{5} + \cancel{c_2} - c_2\sqrt{5} = 2$

$= -\cancel{2}c_2\sqrt{5} = \cancel{2}$

$$\boxed{c_2 = \frac{-1}{\sqrt{5}}}$$

$$\boxed{c_1 = \frac{1}{\sqrt{5}}}$$

Putting it in eqn no (2)

$= \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$

$$\boxed{f(n) = \left(\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n\right)}$$

formula for n^{th} fibo no.

Now,

Complexity :-

$$f(n) = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \underbrace{\left(\frac{1-\sqrt{5}}{2}\right)^n}_{\text{as } n\to\infty \text{ this is } \to 0}\right]$$

less dominating

Ignore

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n\right]$$

$$= O\left(\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n\right)$$

$$= \boxed{O\left((1.6180)^n\right)}$$

$$= \boxed{O(1.6180)^n}$$

final formula

$$\boxed{f(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n}$$

Complexity = $O\left(\left(\frac{1+\sqrt{5}}{2}\right)^n\right)$

or

$$\boxed{O(1.6180)^n}$$

~~Comp~~ Let's summarize

Step ①:- Put $f(n) = \alpha^n$

Step ②:- make a characteris-tic ~~eqn~~ eqⁿ of recurrence.

Step ③:- Find the roots

Step ④:- Fact:- No of roots = No of answers

eg $f(0)$ & $f(1)$ = 0 & 1 resp.

Step ⑤:- put values in $F(n) = C_1\alpha_1 + C_2\alpha_2$

Step ⑥:- Find C_1 and C_2

Step ⑦ :- Put it back in the

$$f(n) = C_1\alpha_1 + C_2\alpha_2$$

$$\sum_{i=1}^{k} C_i\alpha_i \quad eq^n.$$

Step (8) - Simplify it

└ That is your Answer.

for $f(n)$

To find Complexity:-

Just put $\underline{\underline{O(f(n))}}$.

Special case

⇒ if roots are equal

$\alpha_1 = \alpha_2$.

eg:-

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

$$\alpha^n = 2\alpha^{n-1} + \alpha^{n-2}$$

Divide by α^{n-2}

$$\alpha^2 = 2\alpha + 1$$

$$\therefore \alpha^2 - 2\alpha + 1 = 0$$

$$\alpha^2 - \alpha - \alpha + 1$$

$$\alpha(\alpha - 1) - (\alpha - 1) = 0$$

$$(\alpha - 1)(\alpha - 1) = 0$$

$$\alpha_1 = 1, \quad \alpha_2 = 1$$

$$\alpha_1 = \alpha_2$$

if general case :-

α is repeated r times then.

$$\alpha^n, \; n\alpha^n, \; n^2\alpha^n, \; \ldots \; n^{r-1}\alpha^n$$

are solutns to the reccurrence.

Homogeneous Equations

⇒ Does not have any additional function like $g(n)$

Non Homogeneous linear recurrence :→

$$f(n) = a_1 f(n-1) + a_2 f(n-2) + a_2 f(n-3) + \dots + a_d f(n-d) + g(n)$$

when their is this extra function

non-homo ~~non~~ → linear recurrence relation.

(*) How to Solve?

① ⇒ Replace $g(n)$ by 0 & solve usually

$$f(n) = 4f(n-1) + 3^n$$

Now to solve

$$\alpha^n = 4\alpha^{n-1} + 0$$

$$\alpha^n - 4\alpha^{n-1} = 0$$

divide by α^{n-1} to find a characteristic reccurrence relation

$$\alpha - 4 = 0$$

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

$$\boxed{\alpha = 4}$$

Now,

$$f(N) = C_1\alpha_1^n$$

$$f(N) = 4^n C_1$$

$$\boxed{f(N) = (4)^n C_1}$$

(2) ⇒ Take $g(n)$ on ~~one~~ side and find particular Solution.

$$f(n) - 4f(n-1) = 3^n$$

Have something that is Similar to $g(n)$

If $g(N) = n^2$, given polynomial of degree 2.

my guess :-

$$\boxed{f(n) = c \cdot 3^n}$$

$$c3^n - 4c3^{n-1} = 3^n$$

$\cancel{c3^n - 3^n = 4c3^{n-1} - 3^n c\,[1 - 3^{-1}] = 3^n}$

$\cancel{3^n(3(-1) = \frac{4c}{3}}$
$\cancel{4 \cdot 3^n \quad 3c = -3}$

$3(c-1) = 4c$

$3c - 3 = 4c$

$-3 = 4c - 3c$

$c3^n - 3^n = 4c3^{n-1}$

$3^n(c-1) = 4c3^{n-1}$

$(c-1) = \frac{4c}{3}$

$3c - 3 = 4c$

$-3 = 4c - 3c$

$$\Rightarrow \boxed{c = -3}$$

Particular Sol^n:-

$$f(n) = -3 \cdot 3^n$$

$$\boxed{f(n) = -3^{n+1}}$$

③ Add both solution together.

$$f(n) = c_1 \cdot 4^n + (-3^{n+1}))$$

$$f(n) = 4^n c_1 - \underline{3^{n+1}}$$

$$f(1) = 4c_1 - 9 = 1$$ $\frac{7}{4}$

$$c_1 = 10/4 = 5/2$$

$$c_1 = 5/2$$

$$\boxed{f(n) = \frac{5}{2} \cdot 4^n - 3^{n+1}}$$

Ans

How to guess the particular solution

④ If $g(n)$ is exponential, guess of same type

Ex:- $g(n) = 2^n + 3^n$

guess:→ $f(n) = a\,2^n +$
$b\,3^n$

If it is polynomial $(g(n))$,

⇒ guess of same degree → $\leq$ deg

Ex:- $g(n) = n^2 - 1$

⇒ guess of same degree

$f(n) = an^2 + bn + c$

Combination:-

Ex:- $2^n + n$

$f(n) = a2^n + bn + c$

let say you guessed,

$g(n) = a2^n$ and it

→ If fails then try $(an+b)2^n$,

→ if it also failes,

try $(an^2+bn+c)2^n$

Eq^n: $f(n) = 2f(n-1) + 2^n$,

$f(0) = 1$.

Sol^n

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

$\alpha^n = 2\alpha^{n-1}$

$\alpha^n - 2\alpha^{n-1} = 0$

$\alpha - 2 = 0$

$\alpha = 2$

(2) Guess particular sol^n :-

P.S $\Rightarrow a2^n$

$f(n) = a_1 c_1^n$

$= 2 \; a 2^{n-1}$

Date

Page

$a 2^n = 2 a 2^{n-1} + 2^n$

$\cancel{2^n} a = a \cancel{2^n} + \cancel{2^n}$

$a = a + 1$ wrong

guess :-

$(an+b) 2^n$

$(an+b) \cancel{2^n} = \cancel{2} (a(n-1)+b) \cancel{2^{n-1}}$

$+ \cancel{2^n}$

$an + \cancel{b} = a(n-1) + \cancel{b} + 1$

$\cancel{an} = \cancel{an} - a + 1$

$a = 1$

$f(n) = (an+b) 2^n$

$f(n) = (n+b) 2^n$

~~$f(0) = 2 = 0 + b$~~

~~$b = 1$~~

$$\boxed{r^n = (n+1)2^n}$$

$$f(n) = n2^n + 2n$$

$$PS = f(n) = n2^n$$

③ General ans:-

$$f(n) = c_1 2^n + n2^n$$

$$f(0) = 1 = c_1$$

$$c_1 = 1$$

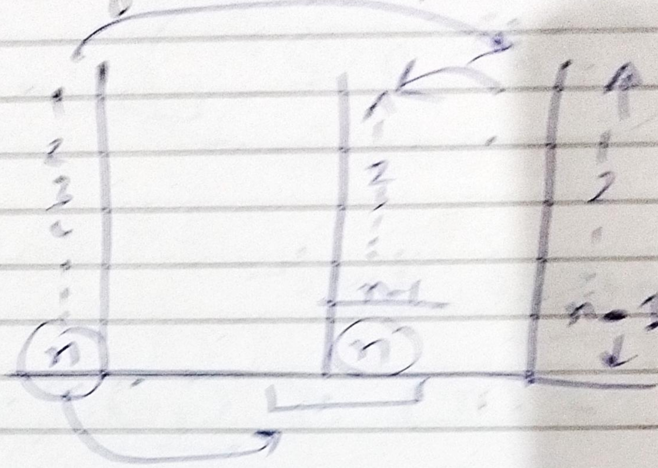
$$\boxed{f(n) = 2^n + n2^n}$$

$$\boxed{O(n2^n)}$$

② Revise Time Complexity

– mit OpenCourseware

⇒ Tower of hanoi problem



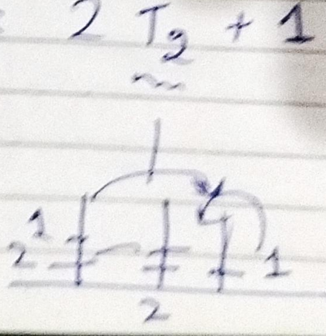
$$T_n = T_{n-1} + 1 + T_{n-1}$$

$$T_n = \cancel{2n-1}\ 2T_{n-1} + 1$$

Recursive solution

$$\boxed{T_n = 2T_{n-1} + 1}$$

for 3, $T_3 = 2T_2 + 1$



$T_2 = 3$ steps

$$T_3 = 2(3) + 1$$

$$T_3 = 6 + 1$$

$$T_3 = 7$$

T_3 = ~~would~~ take me 7 steps

⇒ from you can also calculate
$T_4 \rightarrow T_5 \rightarrow T_6$

→ But eventually we will need a formula for this recurrence relation.

$$\boxed{T_n = 2T_{n-1} + 1}$$

(*) Method - Guess / Subsitution method that we have discussed before.

⇒ $T_n = 2T_{n-1} + 1$

$f(n) = \underset{2}{a_1} f(n-1) + \underset{(g(n))}{1}$ → non-homogeneous Recurrence Relation

Date
Page

Guess & Verify

Guess $\boxed{T(n) = 2^n - 1}$

$$\begin{array}{ccccc} 0, & 1, & 3, & 7, & 15 \ldots \\ 0 & 1 & 2 & 3 & 4 \end{array}$$

$2^n - 1$

$(2^n \ldots$

⇒ **Verify by induction**

$$P(n) : T_n = 2^n - 1$$

Base case :-

$$T_1 = 2 - 1 = 1$$

Inductive Step :-

Assume to prove $T_n = 2^n - 1$ to prove $\boxed{T_{n+1} = 2^{n+1} - 1}$

$$T_{n+1} = 2 T_n + 1$$

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

$$T_{n+1} = 2^{n+1} - 2 + 1$$

$$\boxed{T_{n+1} = 2^{n+1} - 1}$$

#2 Method - Plug & Chug / exhaus-tion / expansion / iterations / Brute Force.

$\Rightarrow \quad T_n = 1 + 2T_{n-1}$

Plug $= 1 + 2(2T_{n-2} + 1)$

Chug. $= 1 + 2 + 4T_{n-2}$

Plug $= \cancel{1+2}\ \overset{1+2}{\cancel{3}} + 4(1 + 2T_{n-3})$

Chug $= \overset{1+2}{\cancel{3}} + 4 + 8T_{n-3}$

$= 1 + 2 + 4 + 8(T_{n-3})$

$= 1 + 2 + 4 + \dots + 2^{i-1}$
$+ 2^i(T_{n-i})$

$= 1 + 2 + 4 + \dots + 2^{i-1} + 2^i(T_{n-i})$

$i = n-1$

$= 1 + 2 + 4 + \dots + 2^{n-2} + 2^{n-1} \; T_1$

$= 2^n - 1$

Let's try for Merge Sort

$2T(\frac{n}{2})$

$\Rightarrow$ To sort $n > 1$,

$x_1, x_2, x_3, \dots \; x_n$

$\downarrow$ (n = power of 2)

1. Sort

$x_1, x_2 \dots, x_{n/2} \;\Big|\; x_{\frac{n}{2}+1}, x_{\frac{n}{2}+1}, \dots x_n$

2. Merge

Ex } Sort $\{10, 7, 23, 5, 2, 4, 3, 9\}$

Sort $\{10, 7, 23, 5\} \Rightarrow \{5, 7, 10, 23\}$

Sort $\{2, 4, 3, 9\}$

$\Rightarrow \{2, 3, 4, 9\}$

$\{5, 7, 10, 23\}$

(2) Merge $= \{2, 3, 4, 5, 7, 10, 23\}$

sorted :)

$= \{2, 3, 4, 5, 7, 10, 23\}$.

Now,

$T(n)$ = # Comparision used merge sort to sort n-numbers.

merging takes :- $(n-1)$ compars worst case.

$2T(n/2)$ comparisions for recursive sorting.

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

Base Case:-

$T(1) = 0$ { ∵ No comparison for 1 element }

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

$$\boxed{\begin{aligned} T(2) &= 1 \\ T(4) &= 5 \\ T(8) &= 17 \\ T(16) &= 49 \end{aligned}}$$

Plug & Chug

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

plug → $= n-1 + 2\left[2T\left(\frac{n}{4}\right) + \frac{n}{2} - 1\right]$

chug → $= n-1 + n-2 + 4T\left(\frac{n}{4}\right)$

plug → $= (n-1) + (n-2) + 4\left[2T\left(\frac{n}{8}\right) + \frac{n}{4} - 1\right]$

thus $\Rightarrow (n-1) + (n-2) + (n-4) + 8T\left(\frac{n}{8}\right)$

Now, we can guess pattern is :-

$$\sum_{i=0}^{k} (n-2^i)\, 2^i T\left(\frac{n}{2^i}\right) \quad 2^n$$

$\Rightarrow n-1 + n-2 + n-4 + n-8$

$+ \dots + (n - 2^{i-1})\, 2^{i} \left(\frac{n}{2^{i}}\right)$

$= n-1 + n-2 + n-4 + \dots + (n-2^{i-1}).$

$2^i T\left(\frac{n}{2^i}\right)$

$\Rightarrow n-1 + n-2 + n-4 + \dots + n - 2^{\log n - 1}$

$+ 2^{\log n} T(1)$

$$\Rightarrow \sum_{i=0}^{\log n - 1} (n-2^i) = \sum_{i=0}^{\log n-1} n - \sum_{i=0}^{\log n - 1} 2^i$$

$= n\log n - (2^{\log n} - 1)$

$= \boxed{n\log n - n + 1}$

=

Linear Reccurence

Def :- Reccurence is said to be linear if it is of the form :-

$$a_1 f(n-1) + a_2 f(n-2) + a_3 f(n-3) + \dots + a_d f(n-d)$$

$$= \sum_{i=1} a_i f(n-i), \text{ for fixed } a_i, d = \text{order}$$

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

Solⁿ :- ① Try $f(n) = \alpha^n$ for constant α

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

$$\alpha^n = \alpha^{n-1} + \alpha^{n-2}$$

divide by α^{n-2}

$$\alpha^2 = \alpha + 1$$

$$\alpha^2 - \alpha - 1 = 0$$

$$\alpha = \frac{1 \pm \sqrt{5}}{2}$$

$$\boxed{\alpha_1 = \frac{1+\sqrt{5}}{2}} \quad \boxed{\alpha_2 = \frac{1-\sqrt{5}}{2}}$$

Golden Ratio :)

Fact :- If $f(n) = \alpha_1^n$ and

$f(n) = \alpha_2^n$ are solutions

to a linear recurrence (without dummy's constant)

then :-

$$f(n) = c_1 \alpha_1^n + c_2 \alpha_2^n$$

is also a solution for any constant c_1 & c_2.

Now,

$$f(n) = C_1\left(\frac{1+\sqrt{5}}{2}\right)^n + C_2\left(\frac{1-\sqrt{5}}{2}\right)^n$$

Now, using Base conditions :-

$f(0) = 0$
$f(1) = 1$

$$f(0) = C_1\left(\frac{1+\sqrt{5}}{2}\right)^0 + C_2\left(\frac{1-\sqrt{5}}{2}\right)^0 = 0$$

$$\Rightarrow C_1 + C_2 = 0$$

$$C_1 = -C_2$$

$$f(1) = 1 = C_1\left(\frac{1+\sqrt{5}}{2}\right)^1 + C_2\left(\frac{1-\sqrt{5}}{2}\right)^1$$

$$C_1\left(\frac{1+\sqrt{5}}{2}\right) - C_1\left(\frac{1-\sqrt{5}}{2}\right) = 1$$

$$C_1\left[\frac{1+\sqrt{5}}{2} - \frac{(1-\sqrt{5})}{2}\right] = 1$$

$$C_1\left[\frac{\cancel{1}+\sqrt{5}-\cancel{1}+\sqrt{5}}{2}\right] = 1$$

$$C_1\left[\frac{\cancel{2}\sqrt{5}}{\cancel{2}}\right) = 1$$

$$\boxed{C_1 = \frac{1}{\sqrt{5}}, \quad C_2 = -\frac{1}{\sqrt{5}}}$$

Now:-

$$f(n) = \frac{1}{\sqrt{5}}\left[\frac{1+\sqrt{5}}{2}\right]^n - \frac{1}{\sqrt{5}}\left[\frac{1-\sqrt{5}}{2}\right]^n$$

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$$

Now,

$$f(n) = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right)$$

-ve

as $n \to \infty$

$$\underbrace{\frac{1-\sqrt{5}}{2}}_{-ve} \to 0$$

so, it is less domin-
-ating

$$f(n) = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n\right)$$