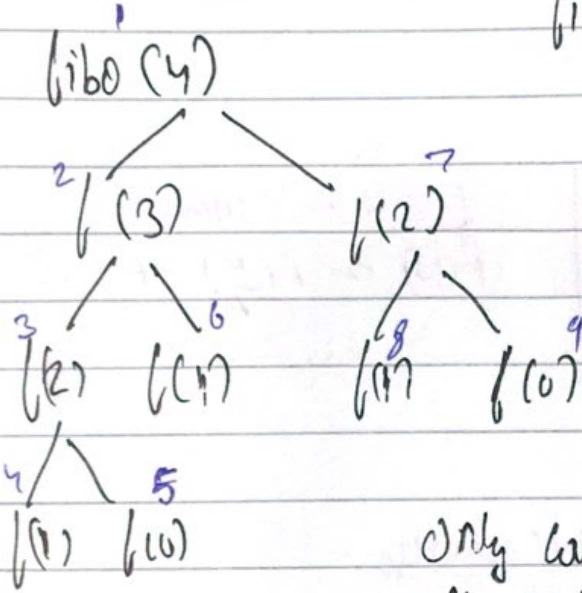
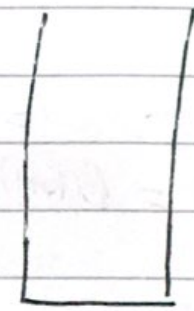
Recurssive Algorithms

fibonacci example





Only calls that are interlinked with each other are in the stack at the same time

Space complexity = ~~high~~ height of recursion tree

=> O(N)

Space complexity of recursive program (here nth fibonacci no.) is O(N)

2 types of recursion
↳ Linear
↳ Divide & conquer

Linear	Divide & Conquer
$F(N) = F(N-1) + F(N-2)$	$F(N) = F(\frac{N}{2}) + O(1)$

① Divide & Conquer recurrences

Form: $T(n) = a_1 T(b_1 n + \varepsilon_1(n))$
$+ a_2 T(b_2 n + \varepsilon_2(n))$
$+ \dots a_k T(b_k(n) + \varepsilon_k(n))$
$+ g(n)$
for some $n \geq n_0$
↓
Some constant

eg: $T(N) = T(\frac{N}{2}) + C$
↳
$a_1 = 1$ $g(n) = C$
$b_1 = \frac{N}{2}$
$\varepsilon(n) = 0$

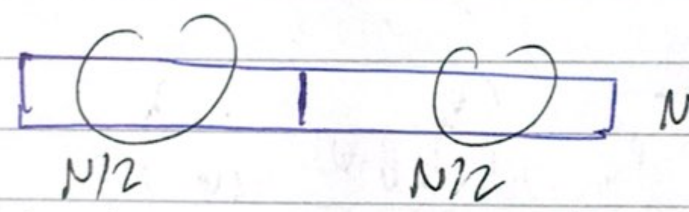
eg $T(N) = 9\,T\left(\frac{N}{3}\right) + \frac{4}{3}\,\cancel{T}\left(\frac{5}{6}N\right) + 4N^3$

a_1 b_1 a_2 b_2 $(g N)$

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

a_1 b_1 $g(N)$

→ When you get answers from this + what you are doing with that answer takes how much time



Sort N/2 Sort N/2 ⇒ via recursion

then add it up / merge them

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

merging ⇒ takes ~~…~~ N-1 comparisons

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

How to actually solve to get complexity?

(1) Plug & Chug

$$\cancel{F(N) + F(N}$$

$$F(N) = F\left(\frac{N}{2}\right) + C$$

(2) Masters theorem

(3) Akra Bazzi (1996)

Akra Bazzi

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

What is p?

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

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

$T(N) = \cancel{\angle} T(\frac{N}{2}) + C$ | $O(1)$
$O(C)$
$O(\cancel{2\ldots}) = O(1)$

Ex: $T(N) = 2T(\frac{N}{2}) + (N-1)$

$a = 2$ $g(n) = n-1$
$b = \frac{1}{2}$

$$2 \times \left(\frac{1}{2}\right)^P = 1$$

$$\Rightarrow P = 1$$

put P in formula

$$T(x) = \Theta\left(x^P + x^P \int_1^x \frac{g(u)}{u^{P+1}} du\right)$$

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

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

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

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

$\cancel{Q \quad q} \; ($

$\Theta(n\log n + 1)$

$\approx \Theta(n\log n)$ // Time Complexity

For array of Size N M.S. Complexity
$= O(N\log N)$

Qn

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

$a_1 = 2$ $a_2 = 8/9$ $g(N) = N^2$
$b_1 = 1/2$ $b_2 = 3/4$

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

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

Let $p = 2$

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

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

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

$\Theta\left(x^2 + x^2 \log x\right)$

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

If you can't find value of P:

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

Let $P = 1$

$\left(3 \times \frac{1}{3}\right) + \left(4 \times \frac{1}{4}\right) \neq 1$

↳ increase the denominator

$P > 1$

Let $P = 2$

$3 \times \frac{1}{9} + 4 \times \frac{1}{16} \Rightarrow \frac{7}{12} < 1$

$\Rightarrow$ ↳ $P < 2$

$\Rightarrow 1 < p < 2$

Note: When $p <$ power of $g(x)$
then ans $= g(x)$

Here $g(x) = x^2$
& $p < 2$
$\Rightarrow$ ans $= \Theta(g(x))$

Why?

for above example

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

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

$$= \Theta(x^p + x^2)$$

and we know that $p < 2$

so

$\approx \Theta(x^2)$ ie $\Theta(g(x))$