Session 1: Introduction to Problem solving

☐ Easy: given n

calculate : $\underbrace{1 + 2 + \cdots + n}$

```
for loop i : 1 -> n :
    sum += i
```

10^{10}

$S_1 = 1 + 2 + 3 + \cdots + n$

$S_1 = n + (n-1) + (n-2) + \cdots + 1$

$2S_1 = (n+1) + (n+1) + \cdots + (n+1)$

$2S_1 = n(n+1)$

$\boxed{S_1 = \dfrac{n(n+1)}{2}}$ → Efficient.

$n = 5$

$1 + 2 + 3 + 4 + 5 = 15$

$\dfrac{5 \times \cancel{6}^{3}}{\cancel{2}} = 15$

☐ medium: n different numbers

$a = [5, -2, 0, 7, 25]$

$K = 10$, x_1, x_2, x_3 s.t. $\boxed{x_1 + x_2 + x_3 = K}$

```
i : 0 -> n-1 :
    j : 0 -> n-1 :
        l : 0 -> n-1 :
            a[i] + a[j] + a[l] == K
```

$100^3 = 10^6$

$= 10^9$

n^2

10^6

$x_1 + x_2 + x_3 = K$

$\Rightarrow x_2 + x_3 = \boxed{K - x_1}$

mp[K − x_1] += 1

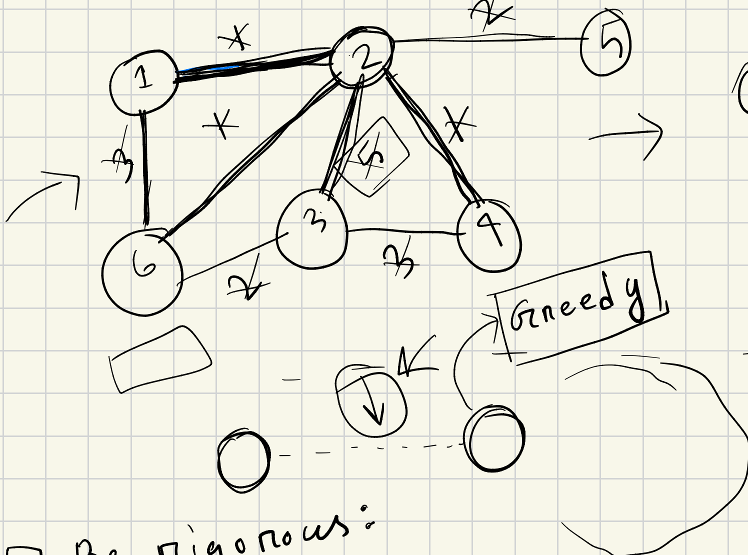
$x_1, x_2 \cdots$

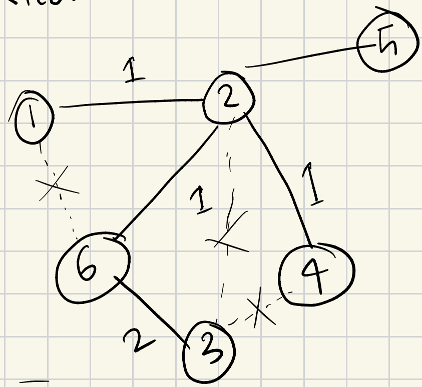
mp[$x_1 + x_2$]

Made with Goodnotes

fast fourier transform

Kruskal algo





- [ ] Be rigorous:

- [ ] Time and Space Complexity:

$\rightarrow O(n)\ ,\ O(n^2)\ ,\ O(n^2 + \boxed{25 + n}) \approx O(n^2)$

$f(n) = n + 5 \Rightarrow O(1) \Rightarrow O(1)$

$$f(n) = \begin{cases} 1 & : n == 1 \\ n + f(n-1) \end{cases}$$

$$f(4) = \boxed{4 + f(3)} = 4 + 3 + f(2) = 4 + 3 + 2 + \boxed{f(1)}^{1}$$

$$= 4 + 3 + 2 + 1$$

$O(n)$

$T(n)$ = number of iterations to find $f(n)$

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

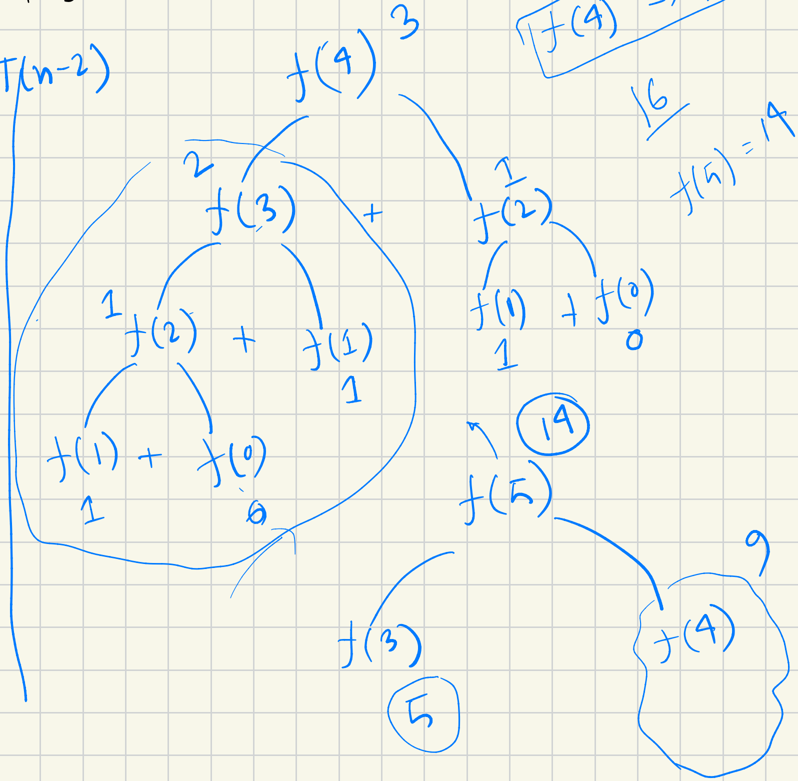
$$= 1 + 1 + T(n-2)$$

$$= 1 + 1 + \cdots\cdots + \overline{T(1)}^{\,1}$$

$$= n \Rightarrow O(n)$$

$$f(n) = \begin{cases} \boxed{n : n < 2} \\ f(n-1) + f(n-2) \end{cases} \Rightarrow \boxed{O(n)} \leftarrow$$

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



$f(4) \Rightarrow 9$

16

$f(5) = 14$ 32

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

$$\le 1 + T(n-1) + T(n-1)$$

$$\le 1 + 2T(n-1)$$

$$\approx 1 + 2\left[1 + 2T(n-2)\right]$$

$$= 1 + 2 + 4T(n-2)$$

$$\vdots$$

$$= 1 + 2 + 3 + \cdots n + 2^n$$

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

$$= \frac{n^2+n}{2} + 2^n = \frac{n^2}{2} + \frac{n}{2} + 2^n$$

$$O\left(\frac{n^2}{2} + \frac{n}{2} + 2^n\right)$$

$$O(\cancel{n^2} + n + 2^n)$$

$$\approx O(2^n)$$

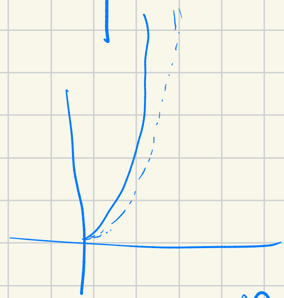
$2^{10} = 1024$

$10^2 = 100$

1000

$\frac{n}{2}$

$O(n^2 +$



$O(n)$ $O(n^2)$

$\boxed{2 \text{ secs.}}$

→ 1/2 core ....

→ $\boxed{\approx 2 \times 10^8 \text{ iterations}}$ — in 1s.

→ $\boxed{O(n^2)}$ $n = 10^4$ ⇒ $\approx 0.5s.$

$\boxed{n = 10^6}$

$$\frac{10^8}{2 \times \cancel{10^8}} \qquad \frac{1000\cancel{,10^{12}}}{\cancel{10}} = 5000s.$$