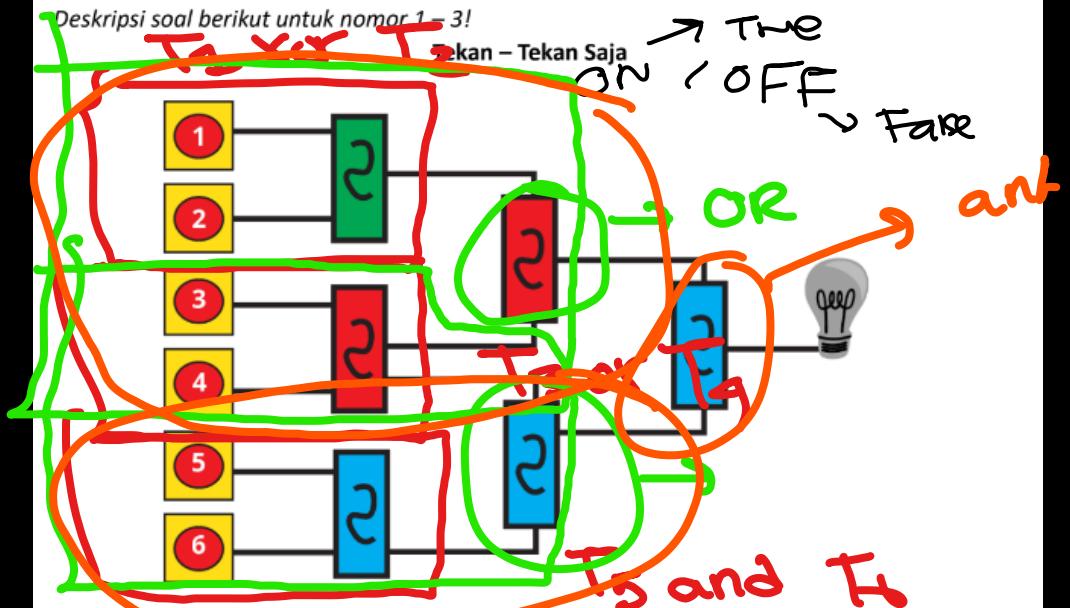


Deskripsi soal berikut untuk nomor 1 – 3!



Pak Dengklek mempunyai rangkaian listrik yang terdiri dari enam buah tombol dan jika ditekan akan mengalirkan arus listrik pada kabel-kabel terhubung. Namun arus yang mengalir akan melewati gerbang controller dengan 3 jenis yaitu : Gerbang yang bisa dilewati satu buah arus terhubung saja , Gerbang yang bisa dilewati minimal satu arus terhubung , dan gerbang yang hanya bisa dilewati jika arus yang terhubung adalah dua arus sekaligus 

1. Dari permasalahan di atas penekanan tombol 1 – 2 – 3 – 4 – 5 – 6 secara berurutan yang benar sehingga lampu dapat menyala adalah ...

- a. ON – OFF – OFF – ON – ON – ON
- b. ON – OFF – ~~OFF~~ – ON – ON
- c. ON – ON – OFF – ON – OFF – ON
- d. ON – ON – ON – OFF – OFF – OFF
- e. OFF – ON – OFF – OFF – ON – OFF

JAWABAN A

minimal salah = OR

hanus salah = XOR

gaboleh False

$$\left(\left(T_1 \text{ xor } T_2 \right) \text{ or } \left(T_3 \text{ or } T_4 \right) \right) \text{ and } \left(\left(T_3 \text{ or } T_4 \right) \text{ and } \left(T_5 \text{ and } T_6 \right) \right)$$

gaboleh $T_5 = F$
atau $T_6 = F$
 T_5 dan $T_6 = \text{True}$

Mage Legendaris Hobi Berpetualang



4. Berdasarkan gambar di atas jika Frieren berjalan pada hari $5^{2023} + 24^{25^{26}}$ Frieren akan berada di ...
- Rumah Fern
 - Rumah Stark
 - Rumahnya Sendiri
 - Bertemu Demon Jahat
 - Tidak dapat ditentukan

$$(5 \bmod 8) + (5^{2023} \bmod 8) \bmod 8$$

$$(5^2)^{10^{11}} \bmod 8 \\ 25 \bmod 8 \equiv (25 \bmod 8)^{10^{11}} \equiv 1$$

$$5^{2023} + 24^{25^{26}} \bmod 8$$

$$(5^{2023} \bmod 8) + (24^{25^{26}} \bmod 8)$$

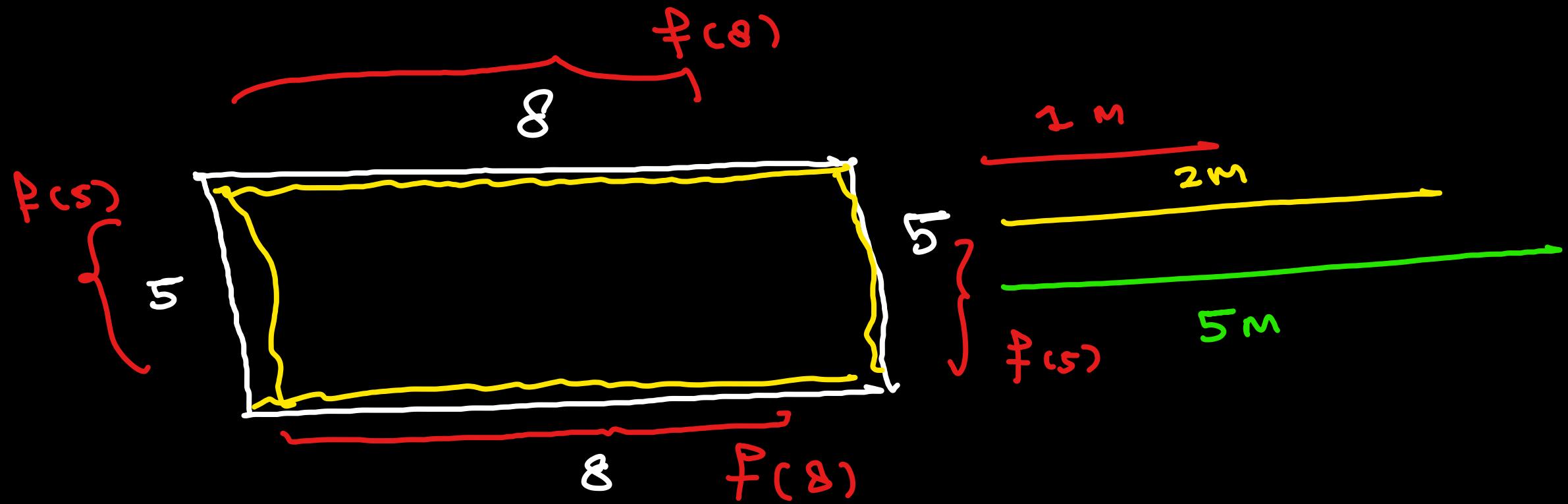
$$\bmod 8 = 5^{2023} \bmod 8$$

$$(5 \cdot 5^{2022} \bmod 8) = 5$$

$$a \stackrel{x}{\pm} b \bmod n \equiv ((a \bmod n) \stackrel{x}{\pm} (b \bmod n)) \bmod n$$

$$\begin{aligned} 99999 \bmod 5 &= ((\cancel{9990} \bmod 5) + \\ &\quad (9 \bmod 5)) \bmod 5 \\ &= 9 \bmod 5 \\ &= \underline{\underline{4}} \end{aligned}$$

$$a^b \bmod n \equiv (a \bmod n)^b \bmod n$$



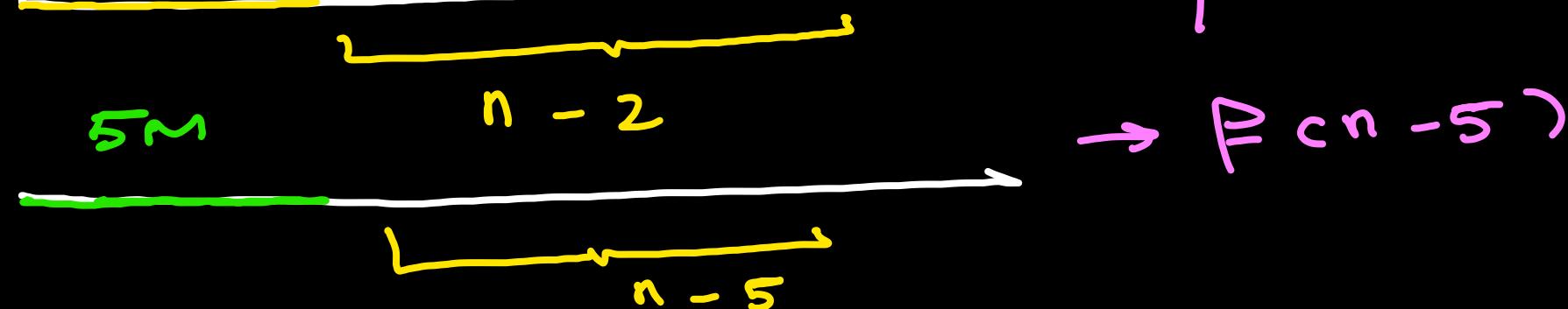
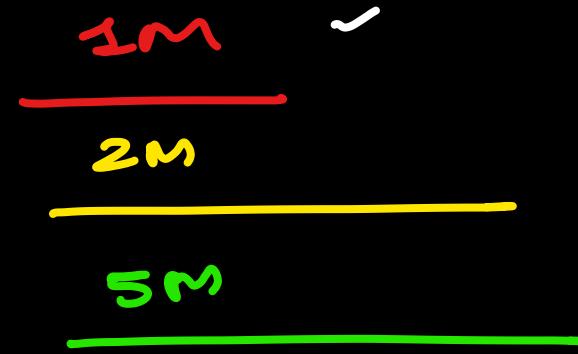
* Dynamic Programming \Rightarrow Recursive

$$F(8) = F(8) + F(5)$$

$$= F(8)^2 + F(5)^2$$



$f(n)$ = banyak cara memasang pagar
pada perimter berukuran n



$$F^{(n)} = F_{\underline{n-1}} + F_{\underline{n-2}} + F^{(n-5)}$$

$$F^{(0)} = 1 \quad , \quad F_{\underline{2}} = 2$$

$$F^{(3)} = 3 \quad , \quad F^{(4)} = 5$$

int base_case[5] = {1,1,2,3,5}

```
if(n<5){
    return base_case[n];
else{
    return ...
```

$$F^{(8)} = \dots$$

$$F^{(5)} = \dots$$

1 semua
 2 1 2
 2 1 1
 1 2 1
 2 2 } 5 cara

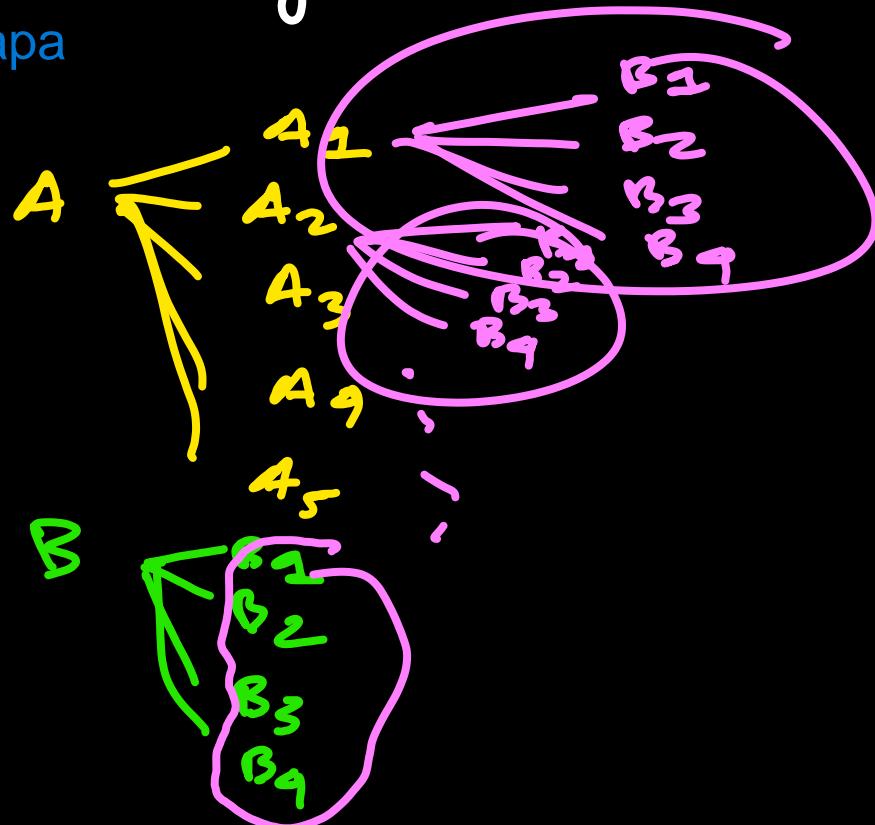
Kombinatorika

* Arusan Perkawian

Ada 5 jalan dari A ke B, 4 jalan dari B ke C, berapa banyak jalan dari A ke C tanpa putar balik

$$5 \times 4$$

$k_1 \leftrightarrow k_2$ (saling berkaitan)
Punya insan



Analisis Kompleksitas

Diberikan input bilangan bulat N
hitung berapa

$$1 + 2 + 3 + 4 + \dots + N$$

↳ gauss summation

$O(C \dots)$ =

↳ Kompleksitas

Big-O Notation

$O(C \dots)$ ↳ $10^8 \Rightarrow 1$ denik

Cara 1 :

```
for(int i = 1; i<=N; i++)  
    sum+=i
```

Cara 2 :

```
cout<<n*(n+1)/2<<endl;
```

* Konstan $\rightarrow \mathcal{O}(1)$

define variable, array, tipe data, dll,
operasi (TIDAK ADA LOOP, TIDAK ADA
RECURSIVE)

```
int n = ...  
if(...)  
else  
cout<< ... <<<
```

* Linear - Polynomial

```
for(int i=1; i<=10; i++){  
    cout<<"Roblox"<<endl;  
}
```

10 iterations

```
for(int i=1; i<=N; i++){  
    cout<<"Roblox"<<endl;  
}
```

} $O(N)$

→ N orang , setiap orang sebut 2x Roblox

```
for(int i = 1; i<=N; i++){
    for(int j = 1; j<=3;j++){
        cout<<"Roblox"<<endl;
    }
}

for(int i = 1; i<=N; i++){
    for(int j = 1; j<=N;j++){
        cout<<"Roblox"<<endl;
    }
}
```

$O(3N)$

$O(N^2)$

```

for(int i = 1; i<=N; i++){
    for(int j = 1; j<=M;j++){
        cout<<"Roblox"<<endl;
    }
}

```

worst $\leftarrow i_{\max} = N$

```

for(int i = 1; i<=N; i++){
    for(int j = 1; j<=i;j++){
        cout<<"Roblox"<<endl;
    }
}

```

$$1 + 2 + 3 + \dots + N$$

$$O(CNM)$$

$$\begin{aligned}j_{\max} &= i_{\max} \\j_{\max} &= N\end{aligned}$$

$$O(CN^2)$$

$$\begin{aligned}&= \frac{n * (n+1)}{2} \\&= \frac{n^2 + n}{2}\end{aligned}$$

$1 \leq N \leq 10^8$ $\mathcal{O}(N^2)$

1 denk $\mathcal{O}(C \dots)$ $\xrightarrow{\leq 10^8}$

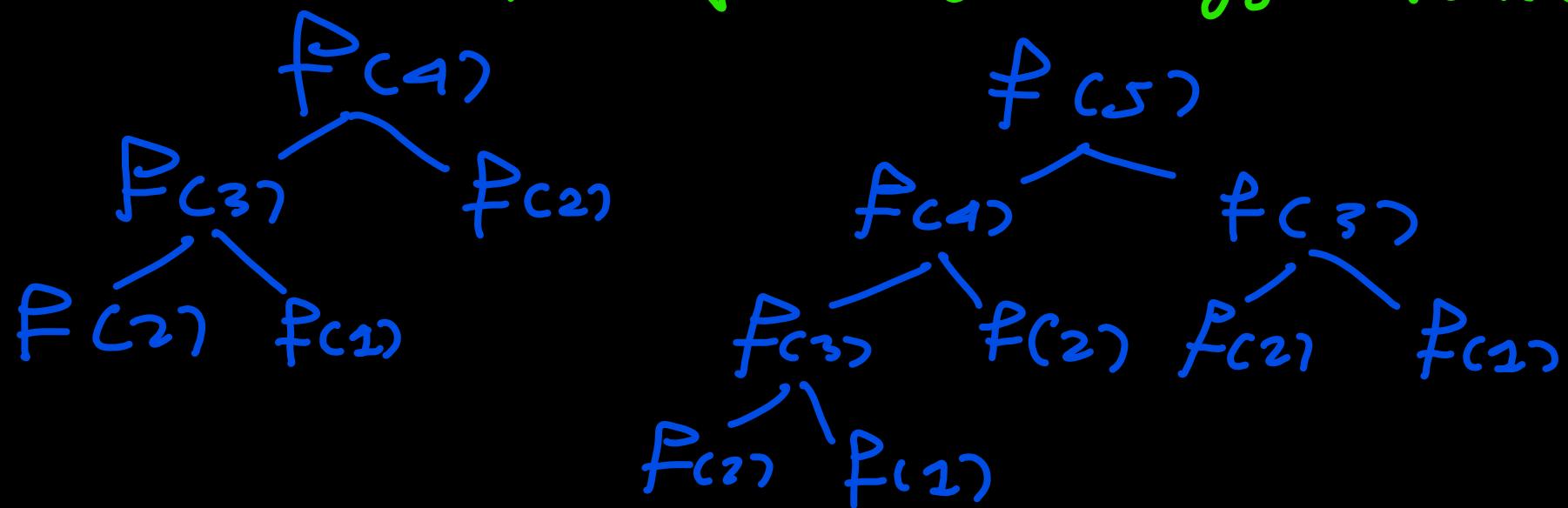
$\mathcal{O}(10^8)^2 = \mathcal{O}(10^{16})$ TLE
 $\xrightarrow{> 1 \text{ denk}}$

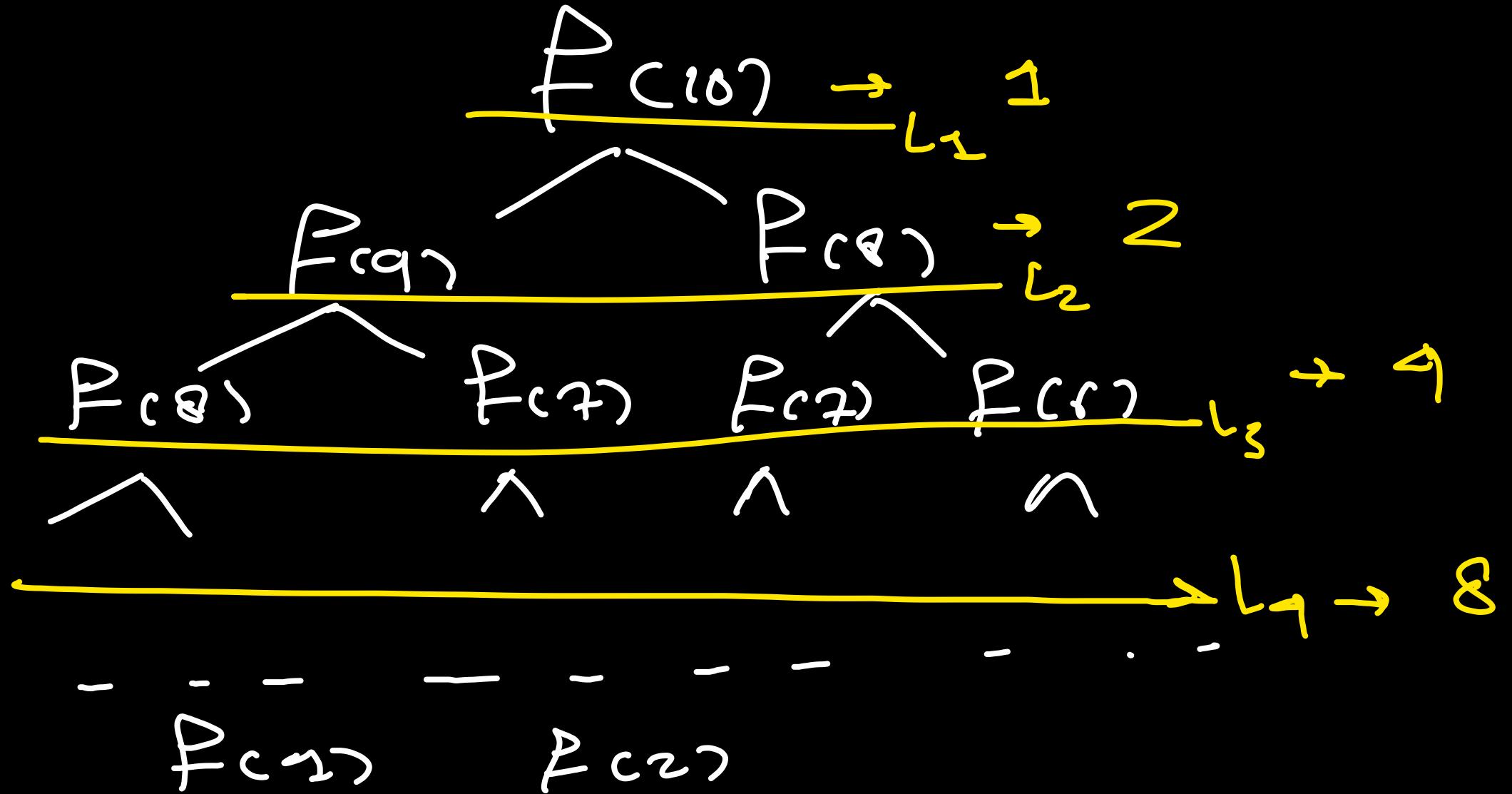
* Eksponensial

```
int f (int n){  
    if(n == 1 || n == 2) return 1;  
    return f(n - 1) + f(n - 2); → Fibonacci'
```

}

Berlalu kau manggil function





di level ke-n manggil function 2^n

$O(C 2^n)$

* Logaritmik

```
int N;  
cin>>N;  
while(N=2){  
    cout<<"Axel""<endl;  
}
```

While (...) {
 ↓
Stop ketika = False
 = 0
 ↑
 axel akan dicerak seus terakhir
 $N = 1$

Jalan ketika = true
!= 0

$$N * 2 * 2 * 2 * 2 * \dots * 2 = N \cdot 2^x$$

N iteration sebangak x

$\frac{N}{2^x} = \frac{N}{\cancel{2^x}}$

banyak iterasi dicoret

sebangak x

$x = \text{banyak iterasi}$

$$\frac{N}{2^x} = 1 \leftrightarrow$$

$$N = 2^x$$

$x = \log N \rightarrow O(\lg N)$

$O(\log N)$

$$a^x = N \Leftrightarrow x = \underbrace{a^{\log N}}$$

```
int T(int N){  
    if(N == 1){  
        return 1;  
    }else{  
        return T(N/2) + N;  
    }  
}
```

OC $\lg N$?

Dynamic Programming

$$f(n) = f(n-1) + f(n-2) \quad f(1)=1, f(2)=1$$

komplexität bis zu $\mathcal{O}(2^n)$

1, 1, 2, 3, 5, 8, 13, 21

