



MATHEMATICS

GULIANA SILVA



* Recursive solution.

→ what is Recursion?

pseudocode.

→

function fac (n).

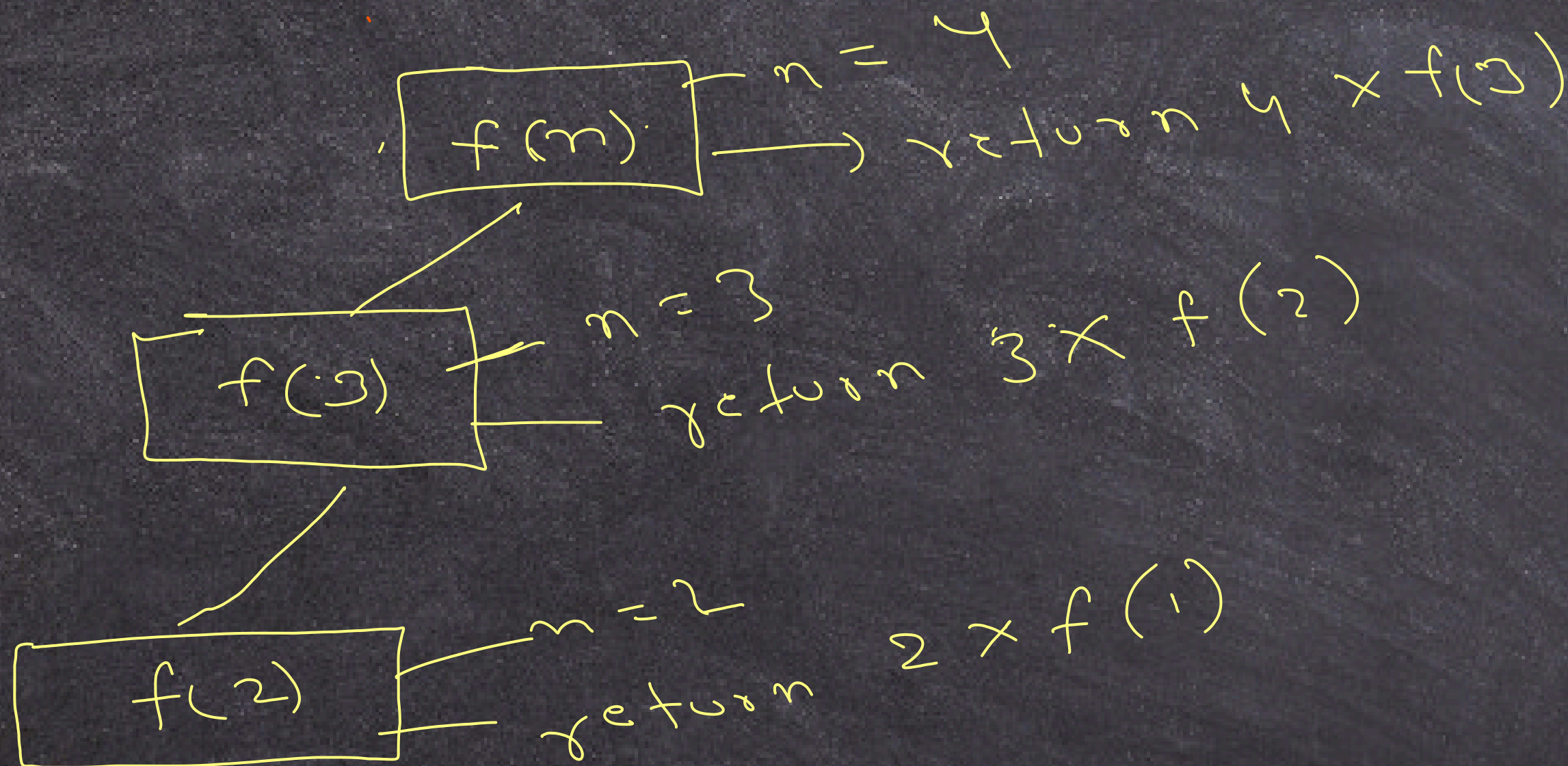
{

if (n == 1) return 1;

return n * f(n-1);

}

~~ex 4~~ $n = 4$



$\frac{27}{2}$ 43200 45 000
 $\frac{27}{2}$ 43200 45 000
 $\frac{27}{2}$ 43200 45 000

* Sol - 61,

→ Pseudo code

```
func Tγ (n){
    int n
    int res = 0 ;
```



```
while(n > 0) {
```

```
    if (n % 10 == 0 & n != 0)
```

```
    {
```

```
        res++;
```

```
        n = n / 10;
```

```
    }
```

```
}
```

Ex 4500

Initially

$$res = 0$$

$$n = 4500$$

Iteration - 01

$$res = 1$$

$$n = 450$$

Iteration 2

$$res = 2$$

$$n = 45$$

Iteration - 3
Break;

$$TC = O(\log n)$$

mathematical proof

$$\frac{n}{10 \times 10 \times 10 \dots} = \frac{3}{10^k}$$

$$\log n \geq \frac{1}{10^k}$$

$$\log n \geq k$$

Ques → find the trailing zeroes of $n!$

→ eg 25
 $1 \times 2 \times 3 \times 4 \times 25$

→ logic

fn $Tr(n)$ {
 $int\ n;$

$int\ res \neq 0;$

 while ($n >= 5$) {

$n = n/5$

$res = res + n;$

}

Output res:

}

$$T_C = O(\log n);$$

mathematical

$$\begin{aligned} \frac{n}{5 \times 5 \times 5 \dots} &= \frac{n}{5^k} \geq 1 \\ &= n \geq 5^k \\ &= \log n \geq k \log 5 \\ &= \boxed{\log n \geq 1} \end{aligned}$$

