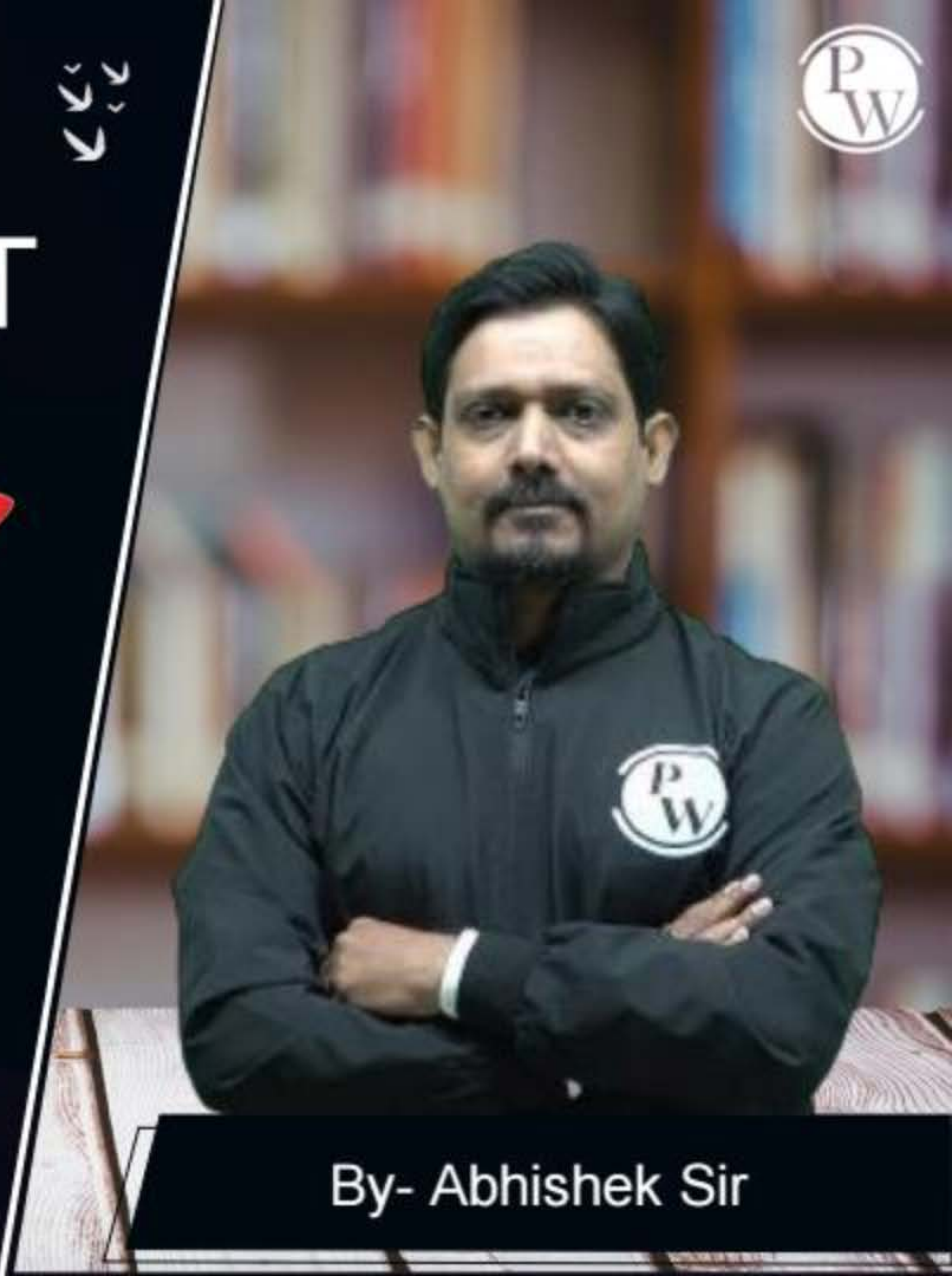
Computer Science & IT

C programming

Function & Storage Class

Lecture No. 04



By- Abhishek Sir

Recap of Previous Lecture

Topic — Recursion

Topic — Recursion Tree (Management)

Topic

Topic

Topic

Slide

Topics to be Covered

Topic practice Recursion

Topic Indirect Recursion

Topic Nested Recursion

Topic Tower of Hanoi

Topic

Slide

Recursion

- * Base Condition
- * Recursion Tree

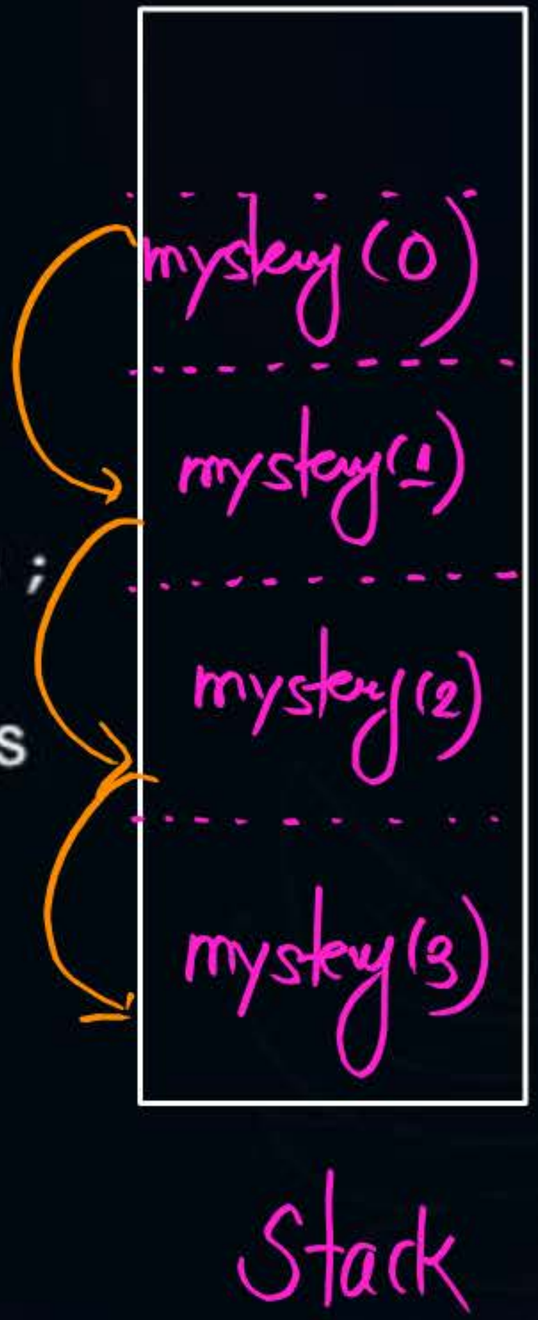
Slide

Question

#Q. Consider the following program

```
int mystery(int n) {
        if (n <= 0)
                return 1;
        else
                return 3 + mystery(n - 1);
}
```

What is the output the program when mystery(3) is called

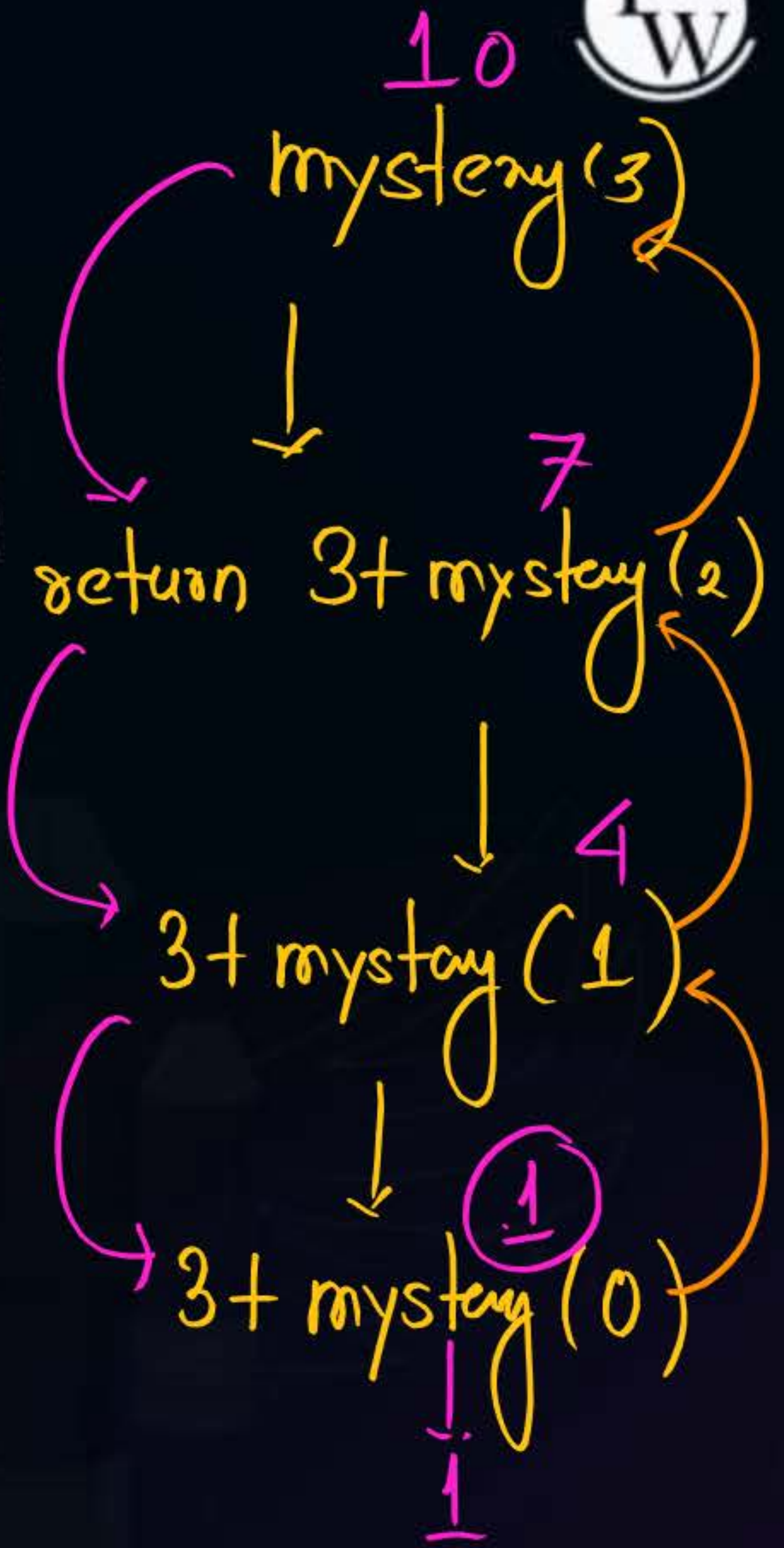


(A) 9

[✓ (B) 10]

(C) 11

(D) 12



Slide

Recursion

#Q. Consider the following program

```
void Dosomething(int n) {
      if (n > 1)
            Dosomething(n-1);
      for (int i = 0; i < n; i++)
            printf("*");
      printf("\n");
}
```

The number of stars will print if the Dosomething(5) is called ?

(A) 7
(B) 10
(C) 1~~6~~5 ←
(D) 17

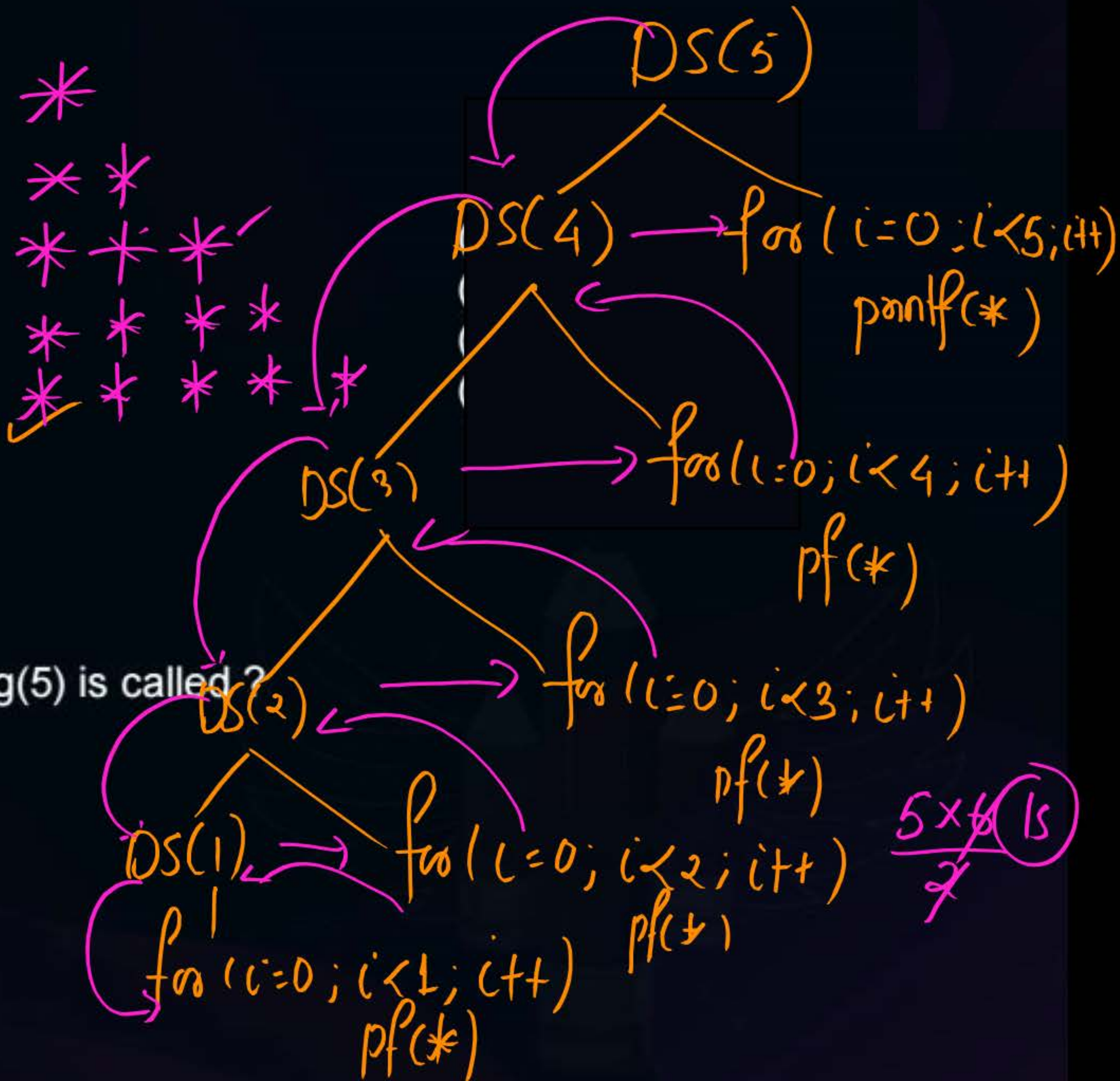
Options were wrong

Recursion

#Q. Consider the following program

```
void Dosomething(int n) {
      if (n > 1)
            Dosomething(n-1);
      for (int i = 0; i < n; i++)
            printf("*");
      printf("\n");
}
```

The number of stars will print if the Dosomething(5) is called ?



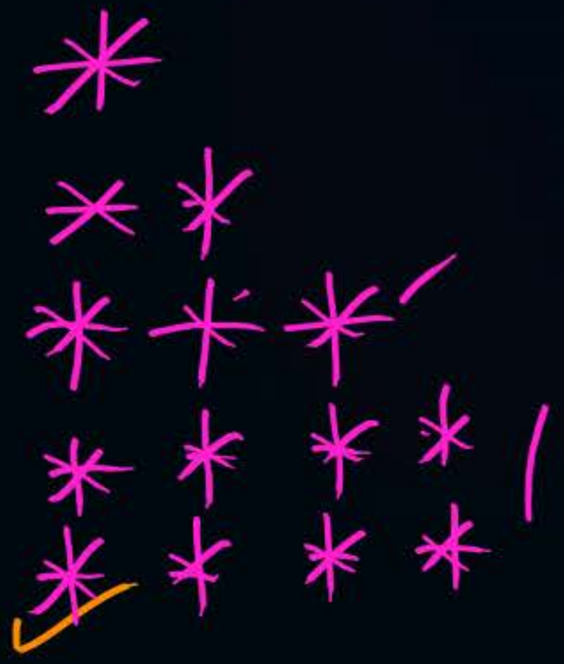
lide

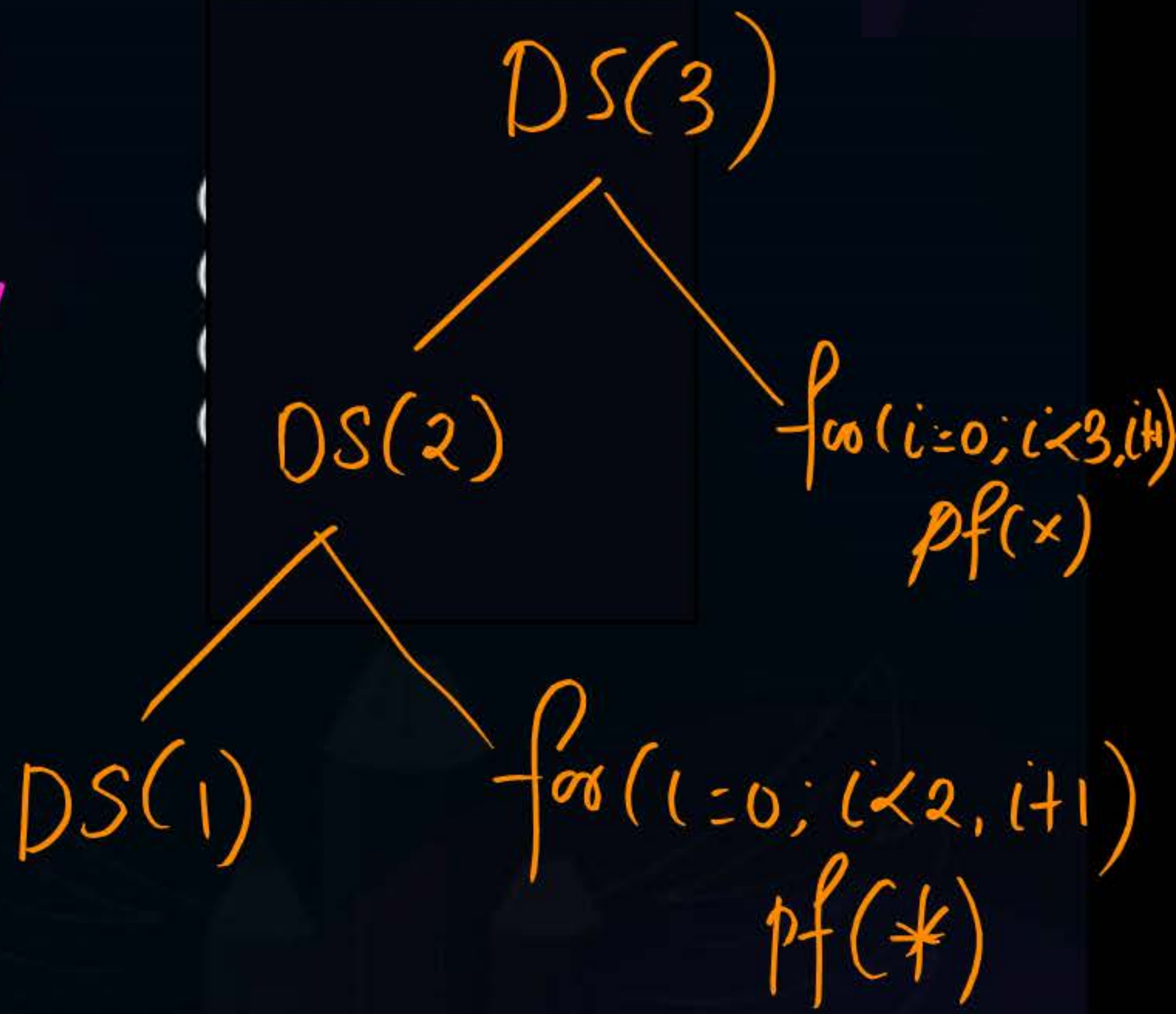
Recursion

#Q. Consider the following program

```
void Dosomething(int n) {
      if (n > 1)
            Dosomething(n-1);
      for (int i = 0; i < n; i++)
            printf("*");
      printf("\n");
}
```

The number of stars will print if the Dosomething(5) is called ?





Recursion

#Q. Consider the following program

```
int foo ( int x , int n){
int val=1;
if (n>0) {
     if (n%2 == 1) val = val *x;
     val = val * foo(x*x , n/2);
}
return val;
}
```

The value returned by foo as a function of x and n is?

(A) $\boxed{x^n}$ ✓

(B) x*n ✓

(C) x*n/2 ✓

(D) None of these

$4,3 - 4^3 = 64$

12

$4\times3/_2$

$= \textcircled{4}$

Slide

Recursion

#Q. Consider the following program

4 3

```
int foo ( int x , int n){
int val=1;
if (n>0) {
     if (n%2 == 1) val = val *x;
     val = val * foo(x*x , n/2);
}
return val;
}
```

odd

foo(4,3) ← 64

n

val: 4

val = 4 * foo(4^2, 1) 16

x n

val: 1 * 16

vall: 16 * foo(16*16, 0)

1

Slide

Recursion

#Q. Consider the following program

```
int foo ( int x , int n){
int val=1;
if (n>0)  {
        if (n%2 == 1) val = val *x;
        val = val * foo(x*x , n/2);
}
return val;
}
```

val=1

$foo(16, 1)$

val=1

$foo(\overset{\times 3}{4}, 3)$ ← 64

3/2

val = 4

val = $4 * foo(\underset{x}{4^2}, \underset{n}{1})$ ← 16

$4^3 = \underline{\underline{64}}$

val = $\underline{1 * 16}$

$\underline{val} = 16 * \underline{foo(16^2, 0)}$ ← 1

n>0

Slide

Recursion

#Q. Consider the following program

```
int foo ( int x , int n){
int val=1;
if (n>0) {
     if (n%2 == 1) val = val *x;
     val = val * foo(x*x , n/2);
}
return val;
}
```

The value returned by foo as a function of x and n is?

(A) x^n ✓

(B) x*n ✓

(C) x*n/2

(D) None of these

foo(4,3)

Slide

Recursion

#Q. Consider the following program

```
int I;
int main(){

    if (i==10)
        return 0;
    i++;
    printf("%i",i);
    main();


}
```

Output

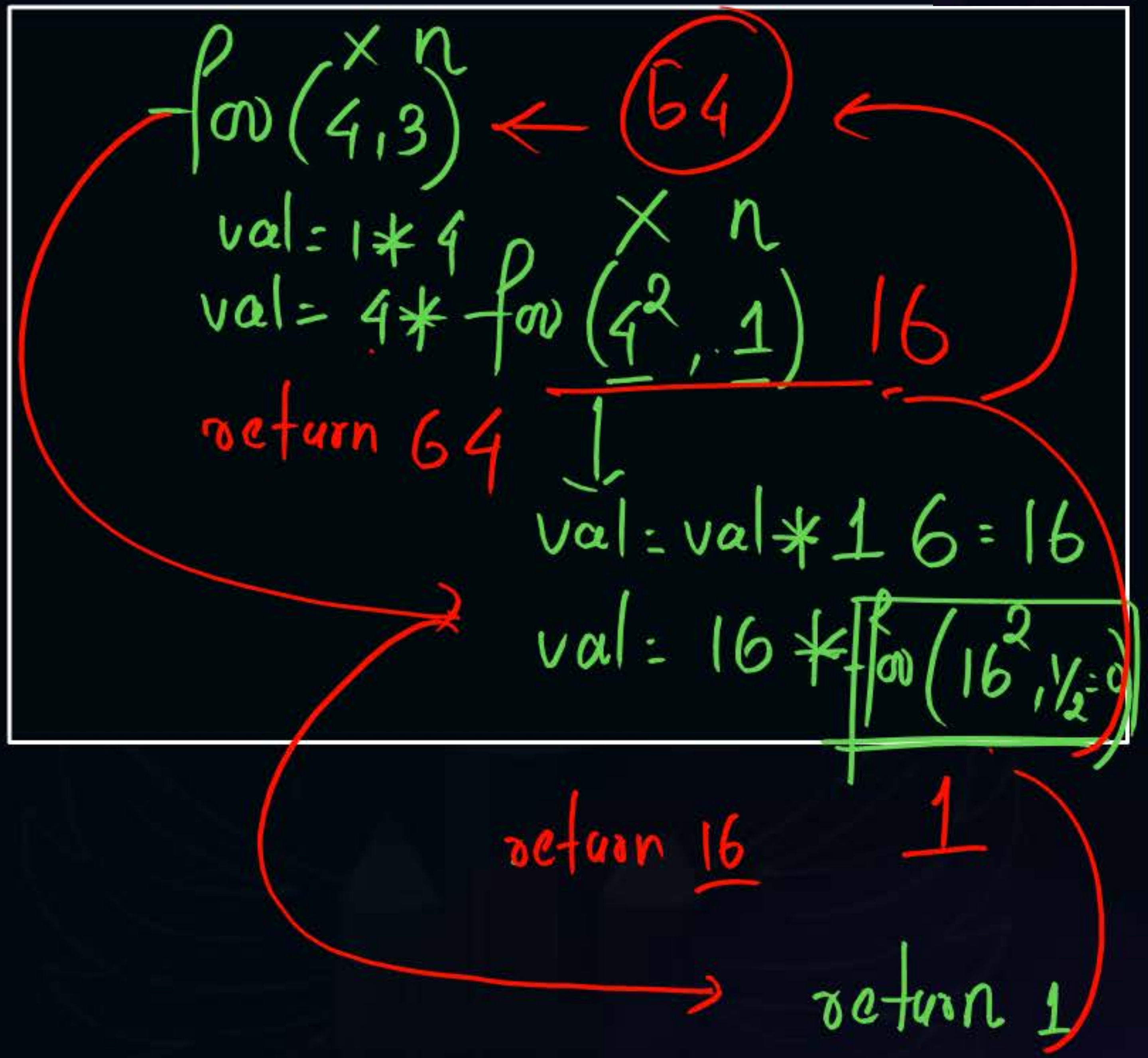
Slide

Question

#Q. Consider the following program

4 3

```
int foo ( int x , int n){
int val=1;
if (n>0) {
        if (n%2 == 1) val = val *x;
        val = val * foo(x*x , n/2);
}
return val;
}
```



$foo(4,3)$ ← 64

$val = 1 * 4$

$val = 4 * foo(4^2, 1)$ 16

return 64

$val = val * 16 = 16$

$val = 16 * foo(16^2, 1/2 = 0)$

return 16

1

return 1

Slide

Recursion

#Q. What value would the following function return for the input x = 97?

Nested Recursion

(a) 89
(b) 90
(c) 91
(d) 92

```
int fun (int x){
    if (x > 100)
        return x - 10;
    else
        return  fun(fun (x + 11));
}
```

Recursive call

Recursion

#Q. What value would the following function return for the input x = 97?

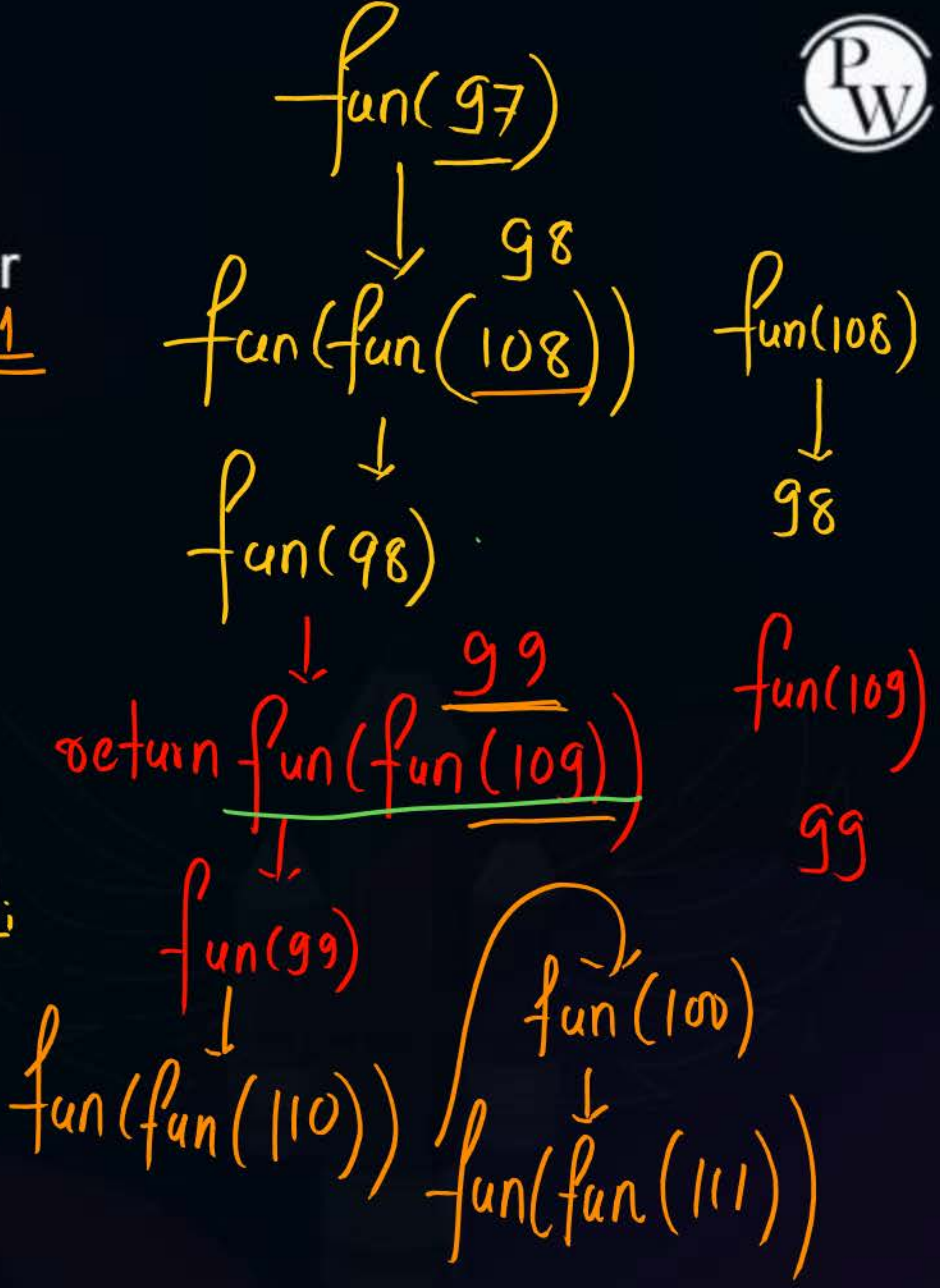
fun(fun(111))

fun(101)

91

```
int fun (int x){
    if (x > 100)
        return x - 10;
    else
        return fun(fun (x + 11));
}
```

91



Slide

Recursion

1996

#Q. What value would the following function return for the input x = 97?

91

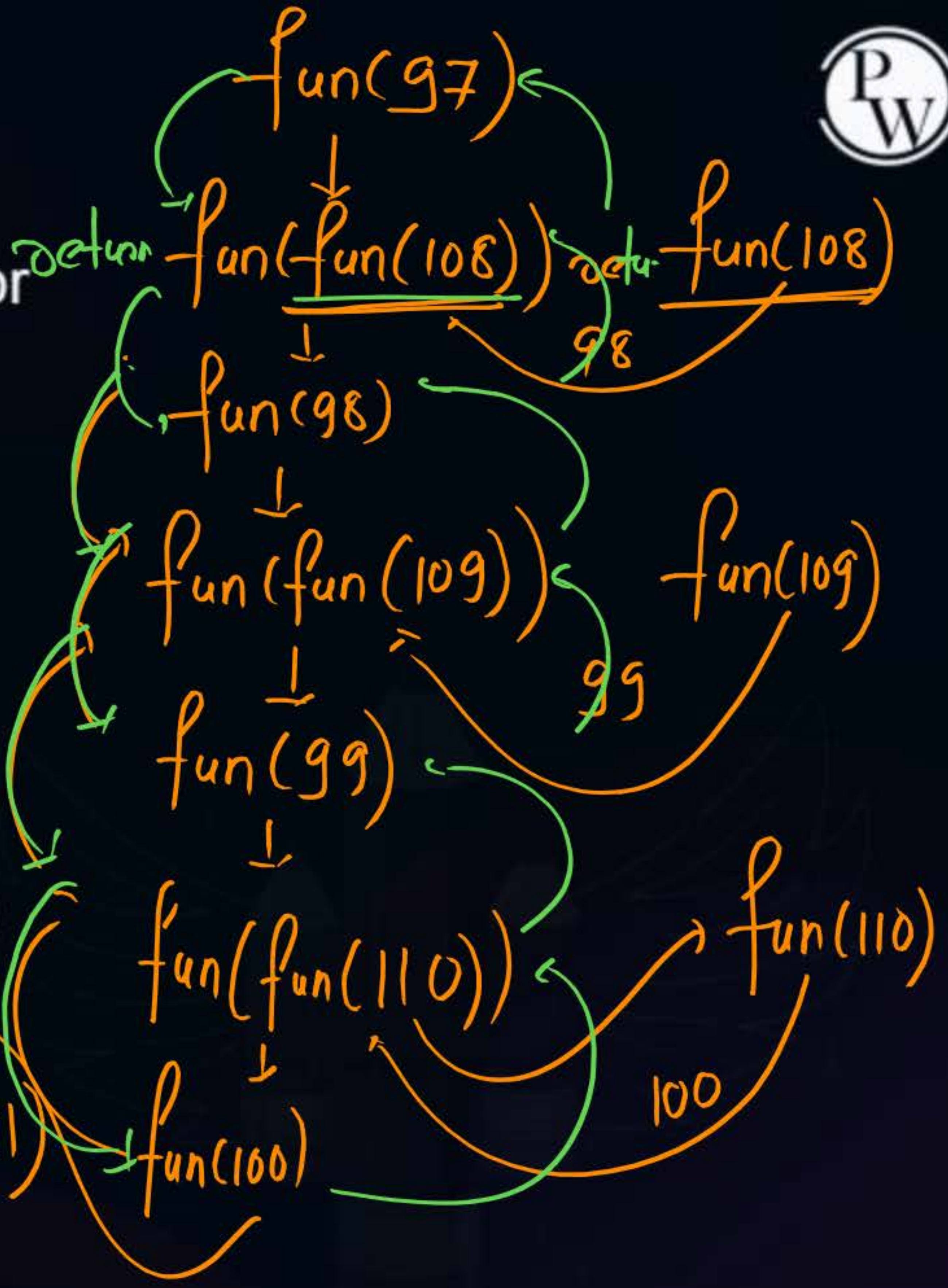
```
int fun (int x){
     if (x > 100)
          return x - 10;
     else
          return  fun(fun (x + 11))
}
```

fun(fun(111))

return fun(101)

= 91

fun(111)



Slide

Nested Recursion is a Recursion in which Recursive Call is passed as parameter.

$\text{factorial}(\underset{\uparrow}{5})$, $\text{fibonacci}(\underset{\uparrow}{4})$

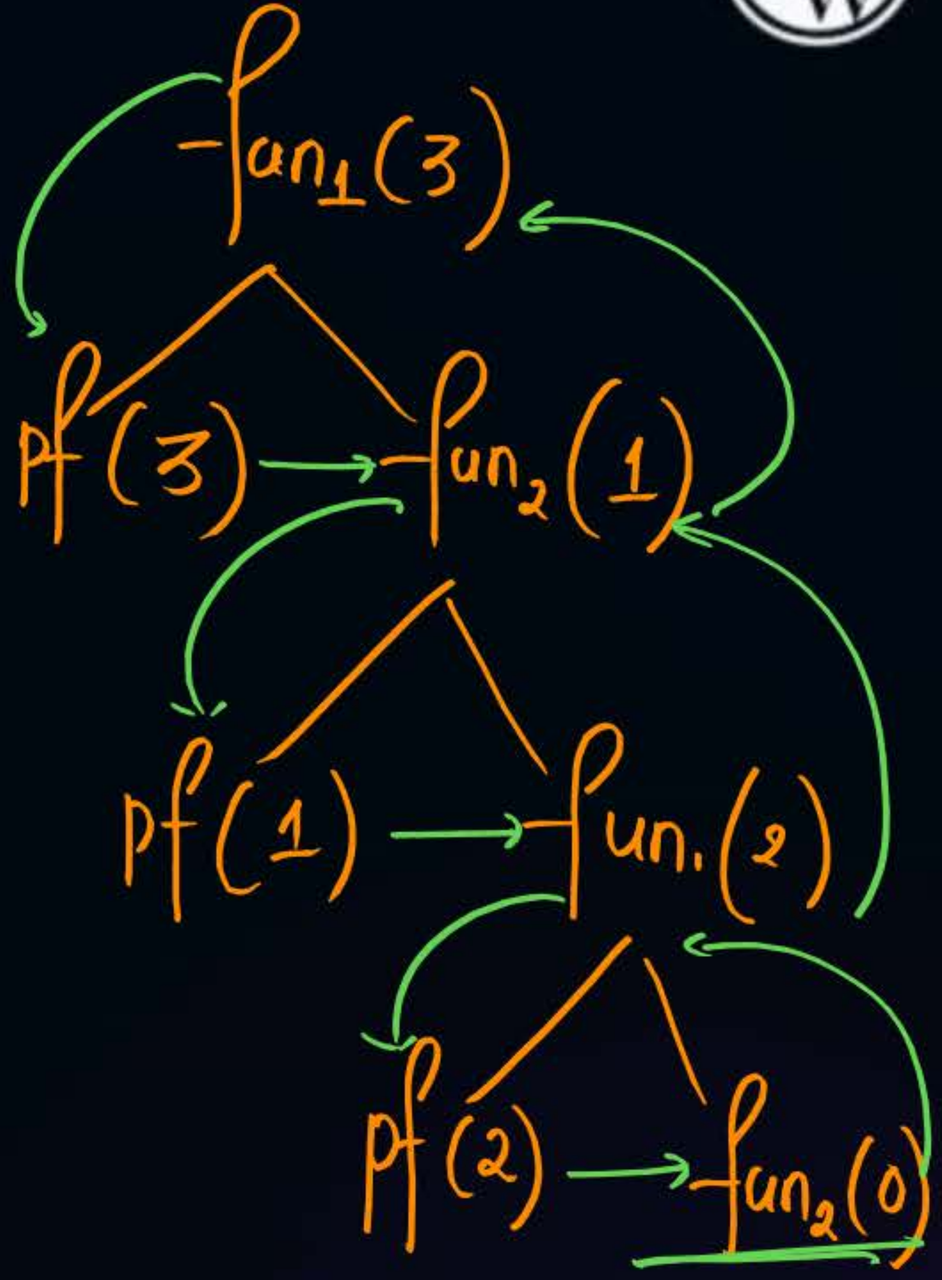
Indirect Recursion

```
void funl (int n) {

if (n = =0 ) return;

printf ("%d" , n);

fun2 (n - 2);

}

int main(){

fun1(3);

}
```

fun₂(3), fun₁(4)

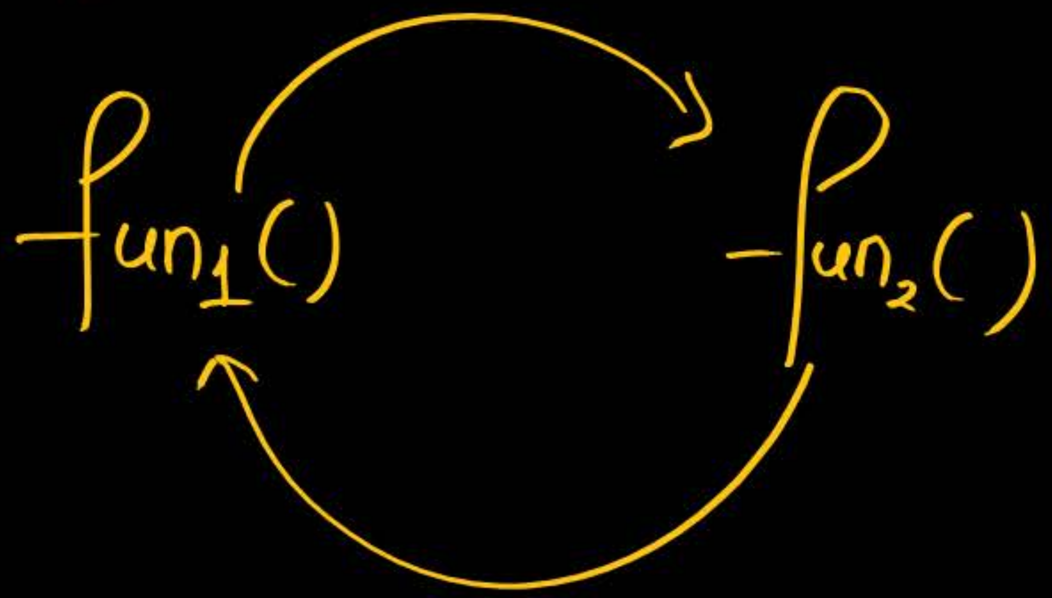
```
void fun2 (int n) {

if (n = = 0) return ;

printf ("%d" , n);

fun1(++n) ;

}
```

3, 1, 2



Indirect Recursion more than 1 Recursion is present

and they call each other in Circular manner.



Indirect Recursion

```
void funl (int n) {

if (n = =0 ) return;

printf ("%d" , n);

fun2 (n - 2);

}

int main(){

fun1(3);

}
```

```
void fun2 (int n) {

if (n = = 0) return ;

printf ("%d" , n);

fun1(++n) ;

}
```

3, 1, 2

if No. of values printed by $\text{fun}_2(3)$ is x and No. of values printed by $\text{fun}_1(4)$ is y the

x+y is ______ ?

Indirect Recursion

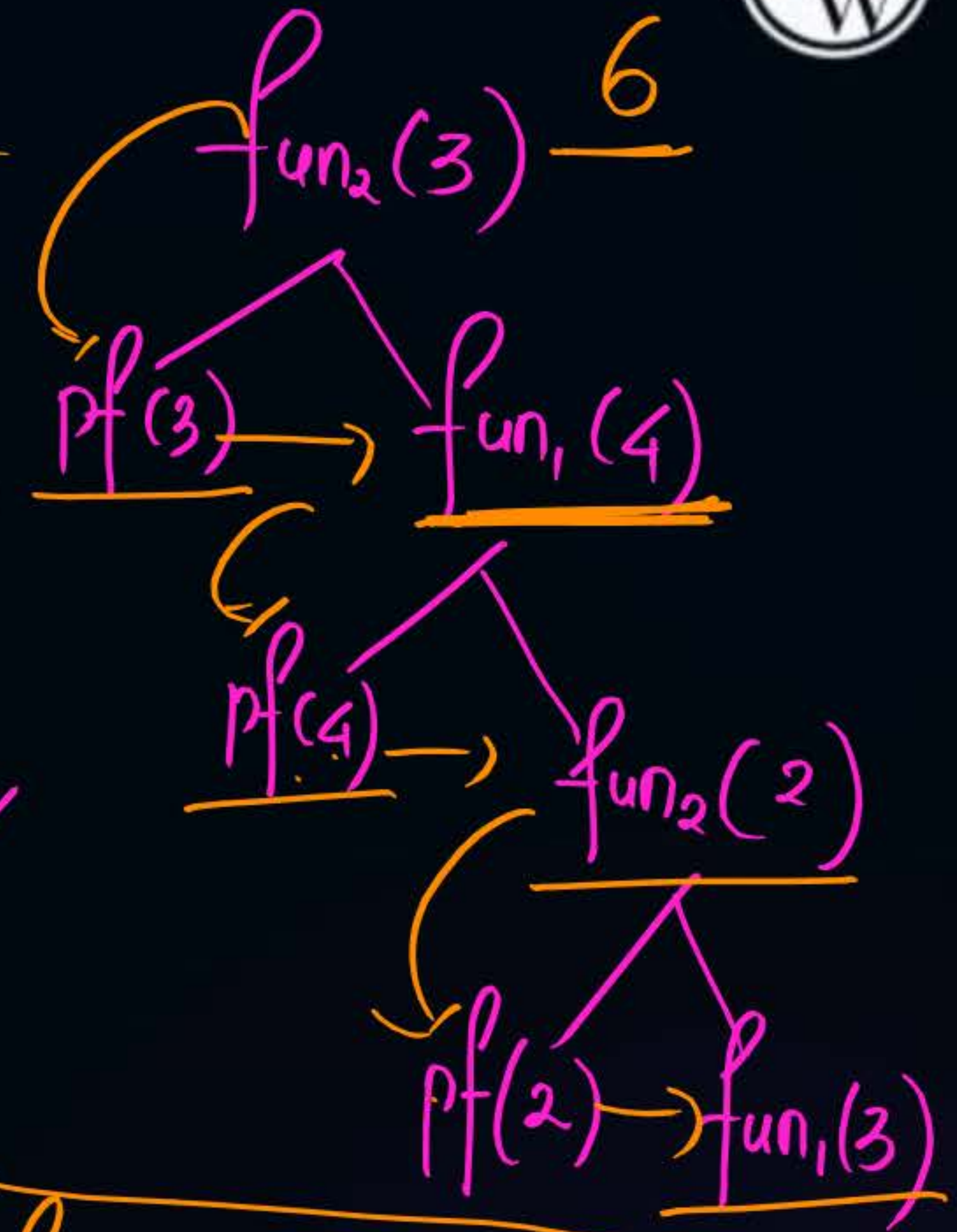
3, 4, 2, 3,2,1

```
void fun1 (int n) {

if (n = =0 ) return;

printf ("%d" , n);

fun2 (n - 2);

}

int main(){

fun1(3);

}
```

```
void fun2 (int n) {

if (n = = 0) return ;

printf ("%d" , n);

fun1(++n) ;

}
```

fun₁(3) ✓

3,1,2



3,2,1

fun₁(4) = 4,2,3,~~2,1~~

Indirect Recursion

```
void fun1 (int n) {

if (n = =0 ) return;

printf ("%d" , n);

fun2 (n - 2);

}

int main(){

fun1(3);

}
```

```
void fun2 (int n) {

if (n = = 0) return ;

printf ("%d" , n);

fun1(++n) ;

}
```

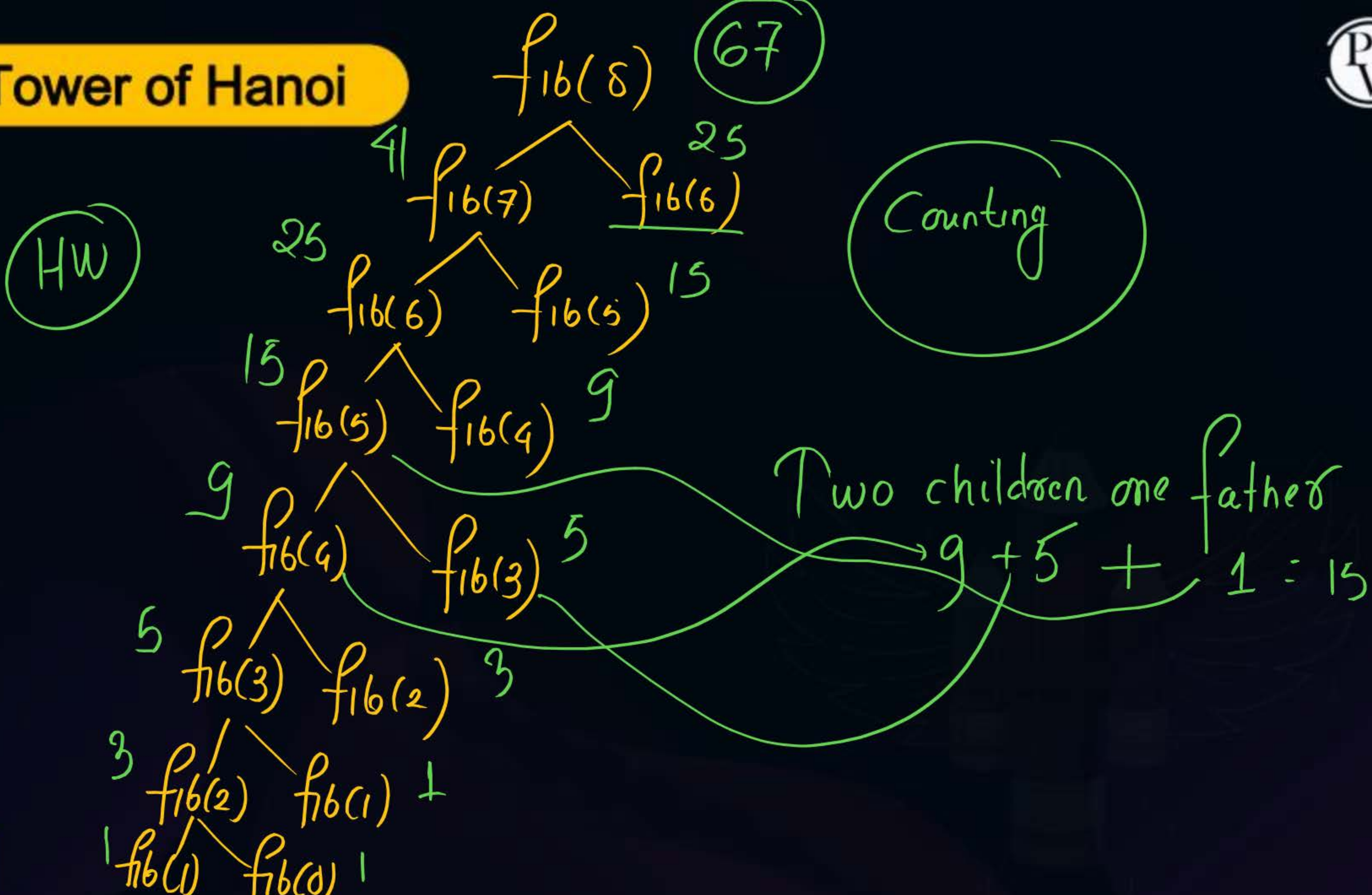
3, 4, 2, 3,2,1 = $fun_2(3) = 6$

$fun_1(4)$:- $\underline{4,2,3,2,1}$

5 = $\boxed{11}$

1

Tower of Hanoi



Slide

Tower of Hanoi

Donot use the Recursion

value calculation

Recursive function

$$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$$

$$\text{fib}(1) = 1$$

$$\text{fib}(0)$$

$\text{fib}(0) = 0$

$\text{fib}(1) = 1$

$\text{fib}(2) = 0+1 = 1$

$\text{fib}(3) = 2$

$\text{fib}(4) = 3$

$\text{fib}(5) = 5$

$\text{fib}(6) = 8$

$\text{fib}(7) = 13$

$\text{fib}(8) = 21$

$\text{fib}(9) = 34$

$\text{fib}(10) = 55$

$\text{fib}(11) = 89$

$\text{fib}(12) = 144$

$\text{fib}(13)$ $\boxed{233}$

Slide

2 mins Summary

Topic practice problem

Topic Nested Recursion

Topic Indirect Recursion

Topic

Topic

Slide

THANK - YOU