

Factorial

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

$$\text{fact}(2) = 2 \times \text{Fact}(1)$$

$$\text{fact}(3) = 3 \times \text{fact}(2)$$

$$\text{fact}(4) = 4 \times \text{fact}(3)$$

$$\text{fact}(5) = 5 \times \text{fact}(4)$$

$$\boxed{\text{fact}(N) = N \times \text{fact}(N-1)}$$

$$\left\{ \begin{array}{l} 4! = 4 \times 3 \times 2 \times 1 = 24 \\ 5! = 5 \times 4 \times 3 \times 2 \times 1 = 120 \end{array} \right\}$$

{
5
4
3
2
1

✓ Base → 1

$$\text{fact}(n) = n * \text{fact}(n-1)$$

$$[n=4]$$

$$\text{fact}(4) = 4 \times \text{fact}(3) \quad 6 = 24$$

$$\text{fact}(3) = 3 \times \text{fact}(2) \quad 2 = 6$$

$$\text{fact}(2) = 2 \times \text{fact}(1) \quad 1 = 2$$

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

Base (1)

```

int fact(int N) {
    if (N == 1) {
        return 1;
    }
    return N * fact(N-1);
}

```

A large blue curly brace groups the entire function. An orange curly brace groups the base case. A green arrow points from the recursive call to the return statement.

$$5! = 5 \times 4! = 120$$

$$4! = 4 \times 3! = 24$$

$$3! = 3 \times 2! = 6$$

$$2! = 2 \times 1! = 2$$

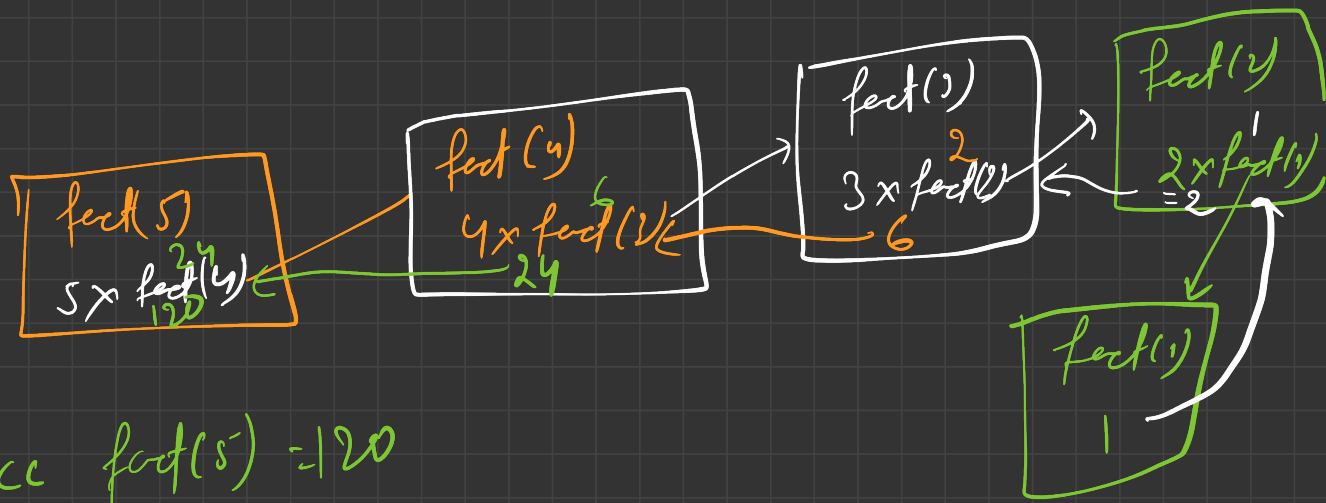
$$1! = 1$$

```

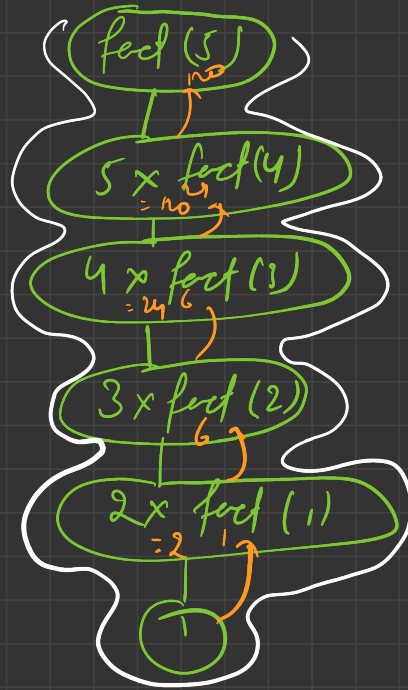
int main() {
    n = 5;
    cout << fact(n);
}

```

A green arrow points to the recursive call in the main function.



cost cc $fact(5) = 120$



Sum of First N number

$$\rightarrow \text{sum}(1) = 1$$

$$\text{sum}(2) = 2 + \text{sum}(1)$$

$$\text{sum}(3) = 3 + \text{sum}(2)$$

$$\text{sum}(4) = 4 + \text{sum}(3)$$

$$\text{sum}(N) = N + \text{sum}(N-1)$$

$$n = 5$$

$$\begin{cases} 5 = 5 + \text{sum}(4) \\ 4 = 4 + \text{sum}(3) \\ 3 = 3 + \text{sum}(2) \\ 2 = 2 + \text{sum}(1) \\ 1 = 1 \end{cases}$$

15

```
int sum(int n){
```

```
    if (n == 1) {  
        → return 1;  
    }
```

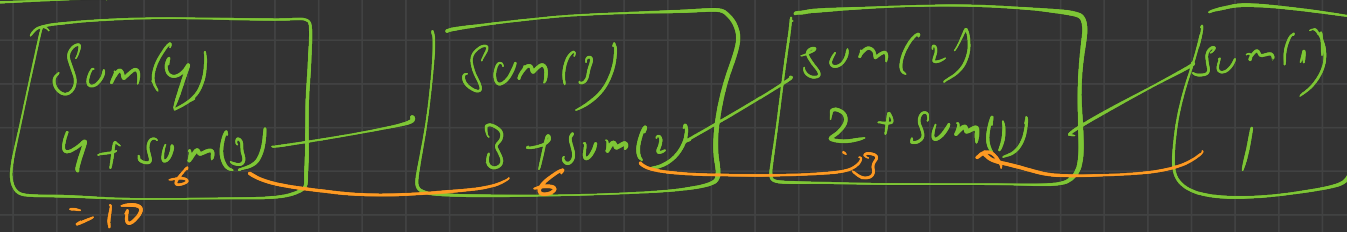
```
    return n + sum(n-1);  
}
```

```
int main() {
```

```
    n = 5
```

```
    cout << sum(n);  
}
```

n = 4

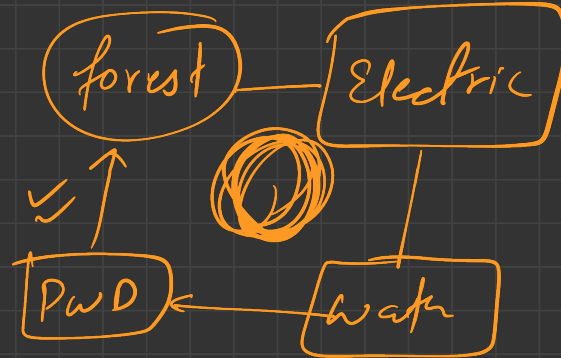
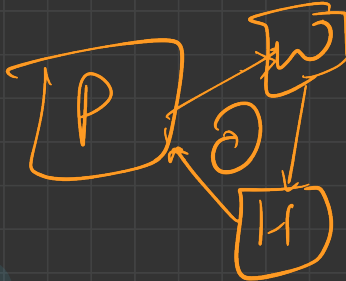


✓✓

$$4 + 3 + 2 + 1 = 10$$

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

✓✓



Power of 2

$$\text{Pow}(2, 1) = 1$$

$$\text{Pow}(2, 2) = 2 \times \text{Pow}(2, 1)$$

$$\text{Pow}(2, 3) = 2 \times \text{Pow}(2, 2)$$

$$\text{Pow}(2, 4) = 2 \times \text{Pow}(2, 3)$$

$$\text{Pow}(2, n) = 2 \times \text{Pow}(2, n-1)$$

num	n
2	5
2	4
2	3
2	2
2	1

```
int pow(int num, int n){
```

```
    if (n == 1){
```

```
        return num;
```

```
    }
```



```
    return 2 * pow(2, n-1)
```

```
}
```

Square of N Number

$$Sq(1) = 1$$

$$Sq(2) = 2^2 + Sq(1)$$

$$Sq(3) = 3^2 + Sq(2)$$

$$Sq(4) = 4^2 + Sq(3)$$

$$Sq(n) = n^2 + Sq(n-1)$$

→ 4

3

2

→ 1

Base

```
int sqsum(int n){  
    if(n==1){  
        return 1;  
    }  
}
```

$n \geq 1$

```
return n * sqsum(n-1)  
}
```

```
int main(){  
    int n = 4  
    cout << sqsum(n);  
}
```