# Recursividade

UM IMPORTANTE PARADIGMA DE PROGRAMAÇÃO

# Objetivo

Apresentar um dos mais importantes paradigmas de programação: a recursividade

# O que é recursividade

É um método de solução de problemas que busca dividir o problema original em problemas menores que tem as mesmas características do problema original que por isso podem ser novamente divididos.

O processo de subdivisão termina quando chegar a um problema suficientemente pequeno, cuja  solução seja conhecida.

A solução de cada divisão é então usada para compor a solução do problema original, juntando as soluções dos menores problemas para resolver problemas maiores, até que o problema original seja resolvido.

# Exemplo de recursividade

Fatorial de um número.

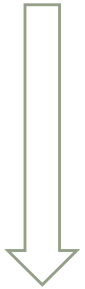
5! = 5 x **4 x 3 x 2 x 1** = 5 x 4!

4! = 4 x **3 x 2 x 1** = 4 x 3!

3! = 3 x **2 x 1** = 3 x 2!

2! = 2 x **1** = 2 x 1!

1! = 1 (problema com solução conhecida)



Problemas são divididos em problemas menores
Até que se encontre um problema com solução conhecida

# Exemplo de recursividade

Fatorial de um número.

5! = 5 x **4 x 3 x 2 x 1** = 5 x 4! = 5 x 24 = 120

4! = 4 x **3 x 2 x 1** = 4 x 3! = 4 x 6 = 24

3! = 3 x **2 x 1** = 3 x 2! = 3 x 2 = 6

2! = 2 x **1** = 2 x 1! = 2 x 1 = 2

1! = 1 (problema com solução conhecida)

Soluções dos problemas menores
São usadas para resolver os problemas maiores

# Função Recursiva ou recursão matemática

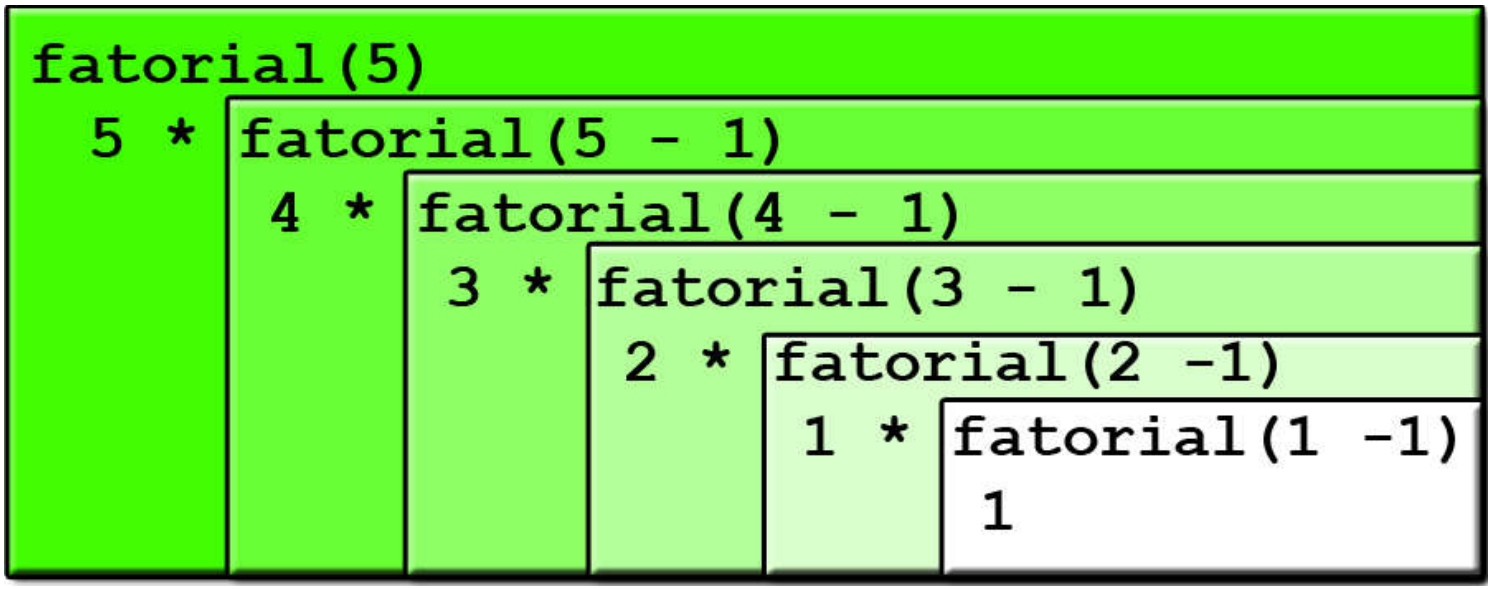
$$N! = \begin{cases} N \times (N-1)! & , se\ N > 1 \\ 1 & , se\ N = 1 \end{cases}$$

# Função recursiva computacional

$$N! = \begin{cases} N \times (N-1)! & , se\ N > 1 \\ 1 & , se\ N = 1 \end{cases}$$

```
FATORIAL(N)
    SE (N > 1)
        RETORNA N * FATORIAL(N – 1)
    SENAO
        RETORNA 1
```

# Execução da função FATORIAL



# Série de Fibonacci

1, 1, 2, 3, 5, 8, 13, ..., F(N-1), F(N)

F(1) = F(2) = 1

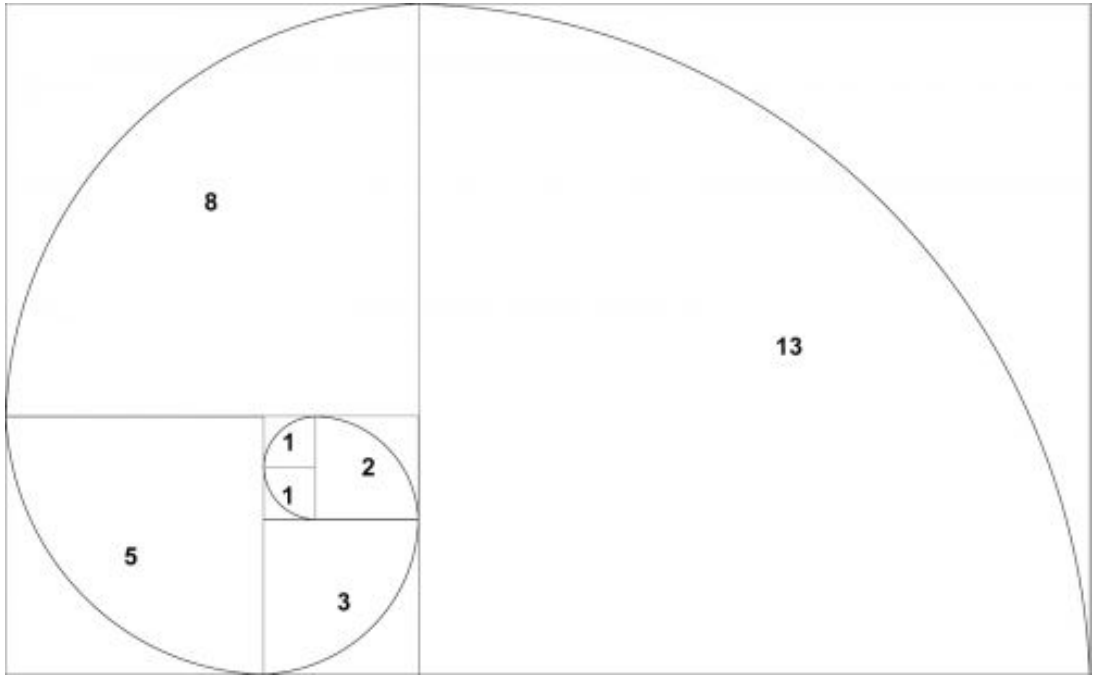
F(N) = F(N-1) + F(N-2)

# Função recursiva computacional

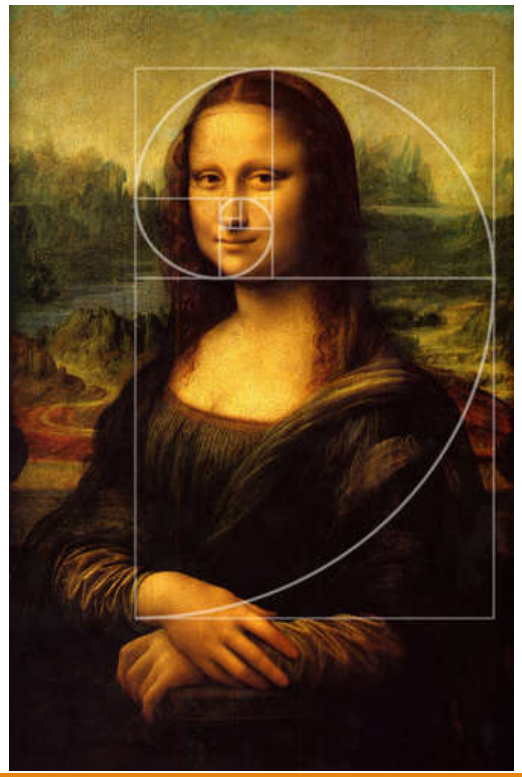
$$F(N) = \begin{cases} F(N-1) + F(N-2) & , se\ N > 2 \\ 1 & , se\ N = 1\ ou\ N = 2 \end{cases}$$

```
F(N)
    SE (N > 2)
        RETORNA F(N - 1) + F (N – 2)
    SENAO
        RETORNA 1
```

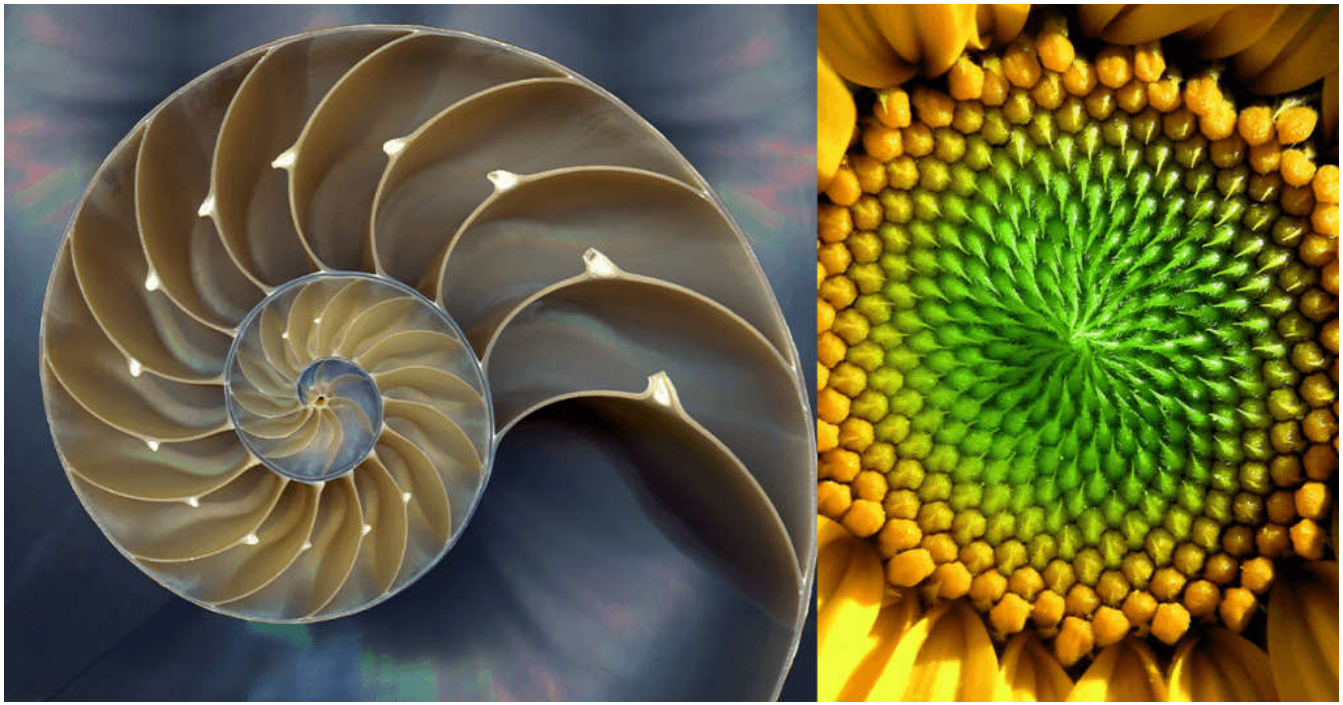
# Aplicações da série de Fibonacci



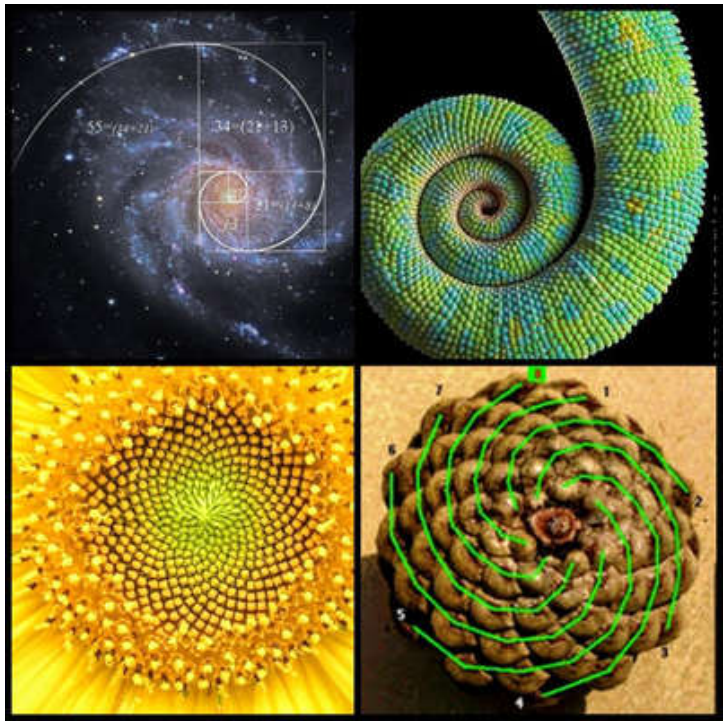
# Aplicações da Série de Fibonacci



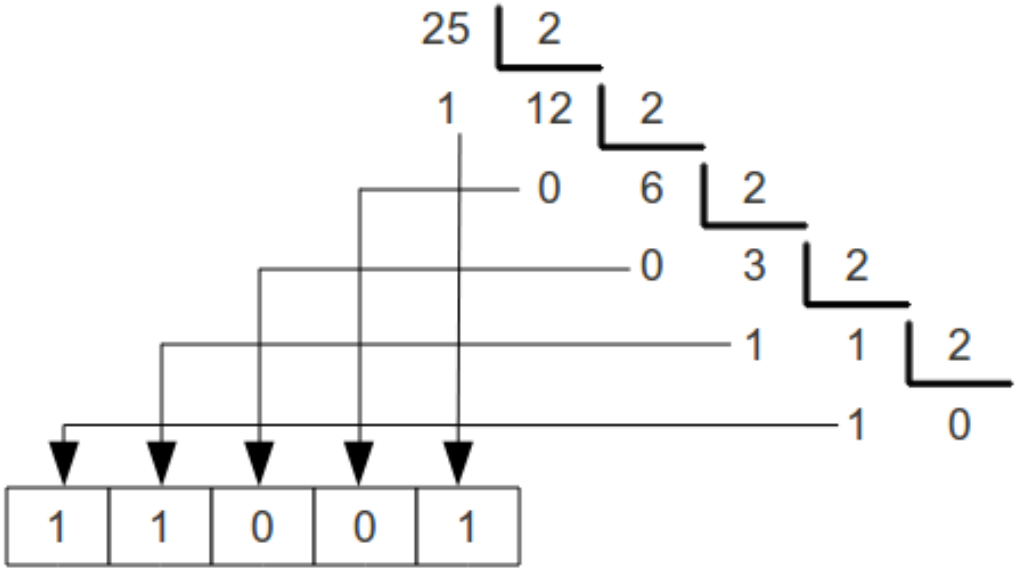
# Aplicações da Série de Fibonacci



# Aplicações da Série de Fibonacci



# Conversão Decimal - Binário



# Função recursiva computacional

```
BIN(N)
    SE (N > 1)
        IMPRIME BIN(N/2)
        IMPRIME (N % 2)
     SENAO
        IMPRIME N
```