

RECURSÃO

Os métodos e funções recursivos aqueles que:

1. Chamam-se a si mesmos;
2. São úteis para alguns problemas, definindo uma chamada ao próprio método/função;
3. Podem ser chamados direta ou indiretamente por um outro método/função.
4. É útil para implementar definições indutivas, por ex: Fatorial, Fibonacci, ordenação, etc.

RECURSÃO

CUIDADO

- Para não gerar uma recursão infinita (looping), devemos certificar de criar uma condição de parada.
- Esta condição de parada é similar aquelas utilizadas nos de laços de repetição (while, for, etc.).
- Permite definir quando a função deverá parar de chamar ela mesma.

EXERCÍCIO PRÁTICO 1

IMPRIMIR INTERVALO DE NÚMEROS

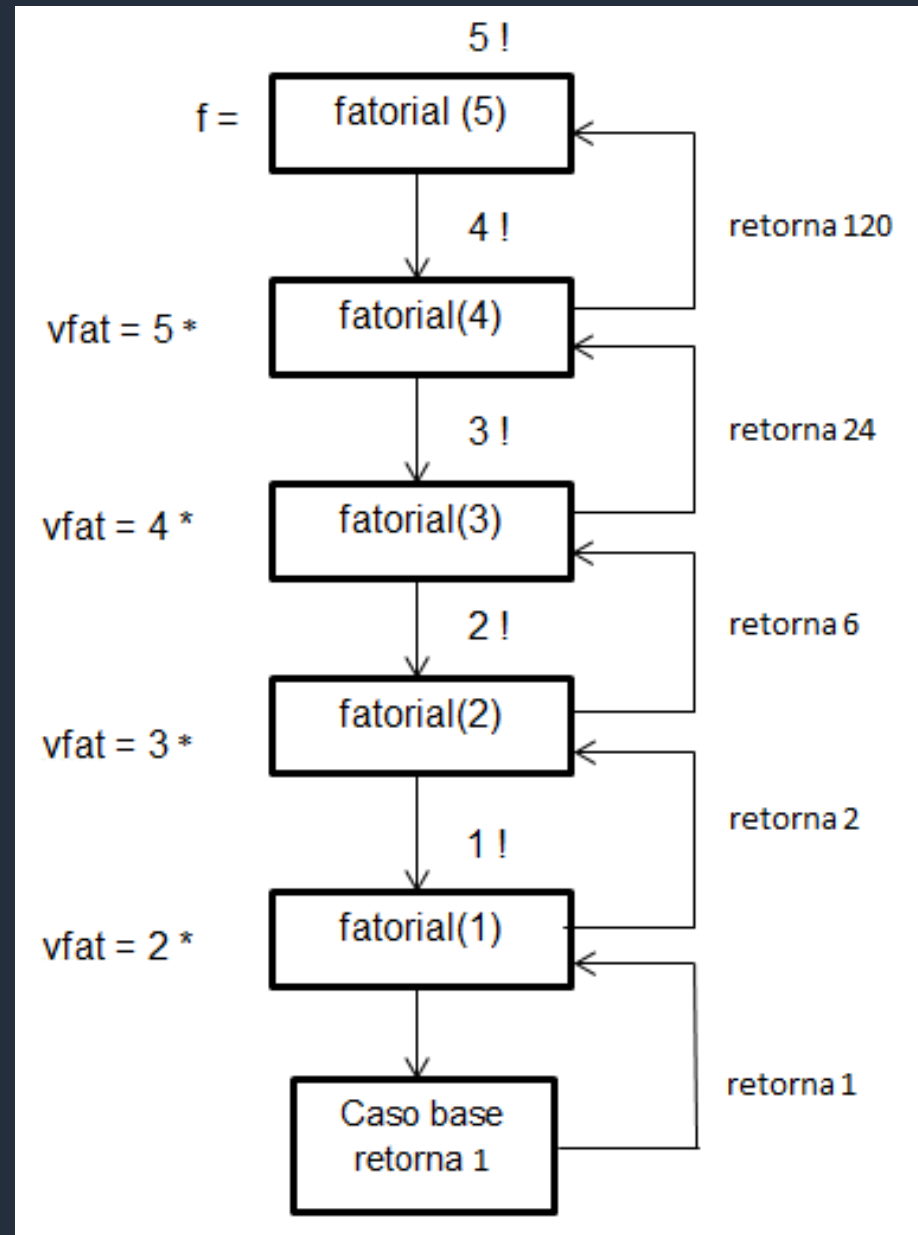
Criar uma função recursiva em C que recebe dois números, sendo o número inicial e o final, a função deve imprimir todos os números dentro deste intervalo.

EXERCÍCIO PRÁTICO 2 – CALCULAR O FATORIAL

Fatorial de n , ou $n!$ é o produto:

- $n * (n - 1) * (n - 2) \dots 1$ com $1!$ e $0!$ definido como igual a 1 (um).
- Solução Iterativa: $5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$
- Solução Iterativa: $5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$
- Solução recursiva: $n! = n * (n - 1)!$

EXEMPLO CALCULAR O FATORIAL



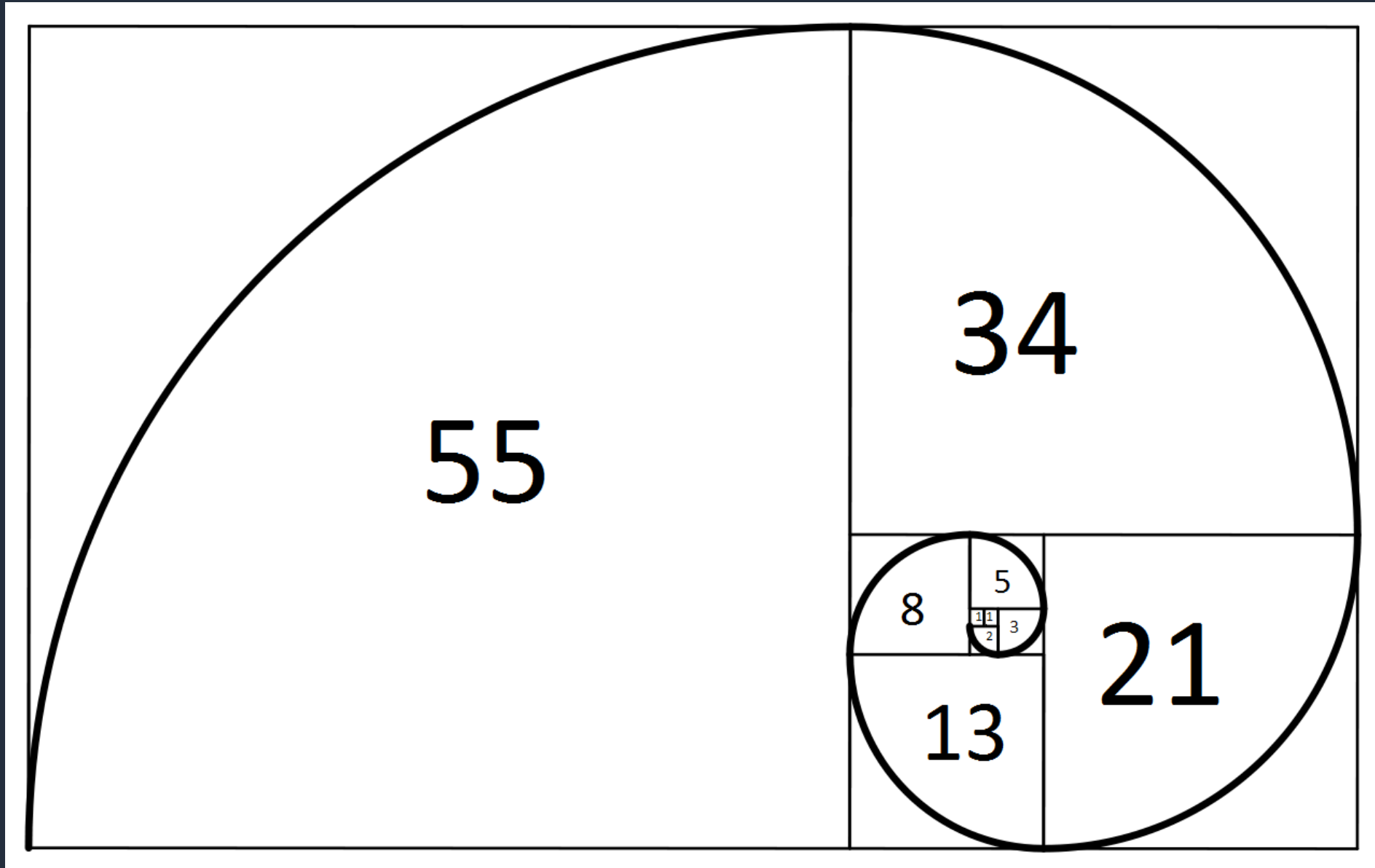
EXERCÍCIO PRÁTICO 3 – CALCULAR E IMPRIMIR FIBONACCI

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

Essa sequência é observada em diversos fenômenos na natureza, ex: caracóis, anatomia humana, espiral nas folhas, reprodução das abelhas, etc.

O número "mágico / misterioso" desta sequência inicia com 0 e 1 e a cada número subsequente é a soma dos dois.

EXERCÍCIO PRÁTICO 3 – CALCULAR E IMPRIMIR FIBONACCI



EXERCÍCIO PRÁTICO 3 – CALCULAR E IMPRIMIR FIBONACCI

A série de Fibonacci pode ser definida recursivamente como:

- fibonacci(0)
- fibonacci(0) = 0
- fibonacci(1) = 1
- fibonacci(n) = fibonacci(n - 1) + fibonacci(n - 2)

EXERCÍCIO PRÁTICO 4 – CALCULAR FUNÇÃO F(X)

Escrever um método recursivo para a seguinte função:

$$f(x) = \begin{cases} x - 3, & x \leq 4 \\ f(x - 2) + 5, & x > 4 \end{cases}$$

