

Recursividade

Profª Ms. Erica Lopes



Definição

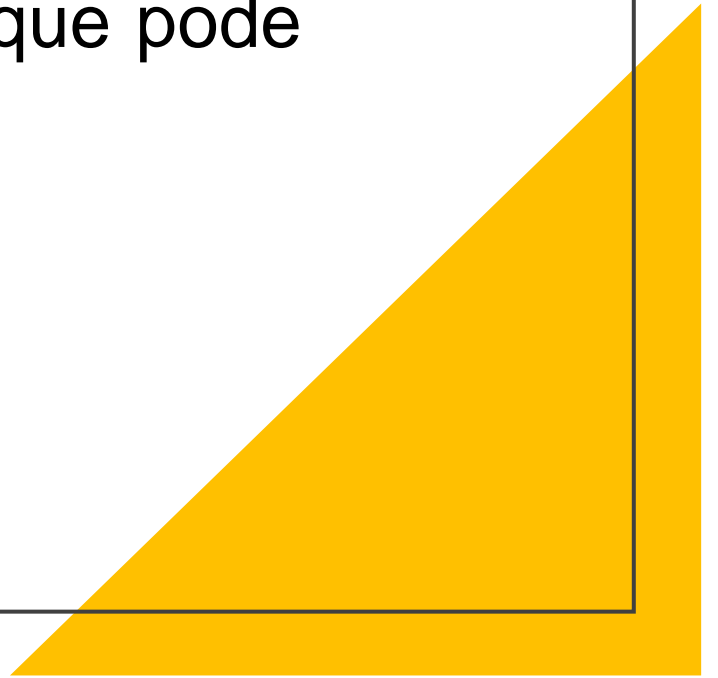
Segundo o Dicionário Houaiss da língua portuguesa:

“propriedade daquilo que se pode repetir um número indefinido de vezes.”.




Imagem recursiva

No nosso caso, ou seja, em computação, recursividade, também poderá repetir um conjunto de comandos, porém um número finito de vezes. É uma subrotina (função ou procedimento) que pode chamar a si mesma.



A recursão é uma das técnicas mais interessantes e eficazes em programação. Por ser uma função que chama a ela mesma. É criado um “loop” a cada chamada, entretanto, deve **obrigatoriamente ter uma condição de parada**, sua falta implicará na execução de um “loop” infinito, ou seja, não haverá interrupção a si mesma.

A yellow right-angled triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

A stylized blue logo consisting of the letters 'f' and 'x' in a cursive, handwritten style.

Exemplo – Cálculo Fatorial

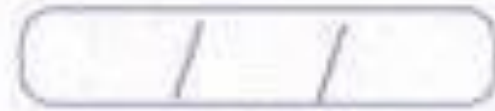
Um exemplo clássico de uma função recursiva é o cálculo do fatorial de um número inteiro qualquer. Utilizaremos uma função que irá calcular o 5!



5!

Como calcular o fatorial de um número?

Em matemática, para representar o fatorial de um número natural, (não se calcula o fatorial de números não naturais), utilizamos o ponto de exclamação. Por exemplo, para representar o fatorial do número cinco, escrevemos $5!$. Mas que operação é esta? E como é que procedo para saber o valor do fatorial de cinco? É muito simples, basta multiplicar o cinco por cada um dos números naturais menores que ele próprio, ou seja,



$$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$



Agora vamos
fazer a
mesma coisa
em C. rs

$$5! = ?$$

$$\text{Fatorial} (5) = 5 * \text{fatorial} (4)$$

$$\text{Fatorial} (4) = 4 * \text{fatorial} (3)$$

$$\text{Fatorial} (3) = 3 * \text{fatorial} (2)$$

$$\text{Fatorial} (2) = 2 * \text{fatorial} (1)$$

$$\text{Fatorial} (1) = 1 * \text{fatorial} (0)$$

$$\mathbf{\text{Fatorial} (0) = 1}$$

$$\text{Fatorial} (1) = 1 * 1$$

$$\text{Fatorial} (2) = 2 * 1$$

$$\text{Fatorial} (3) = 3 * 2$$

$$\text{Fatorial} (4) = 4 * 6$$

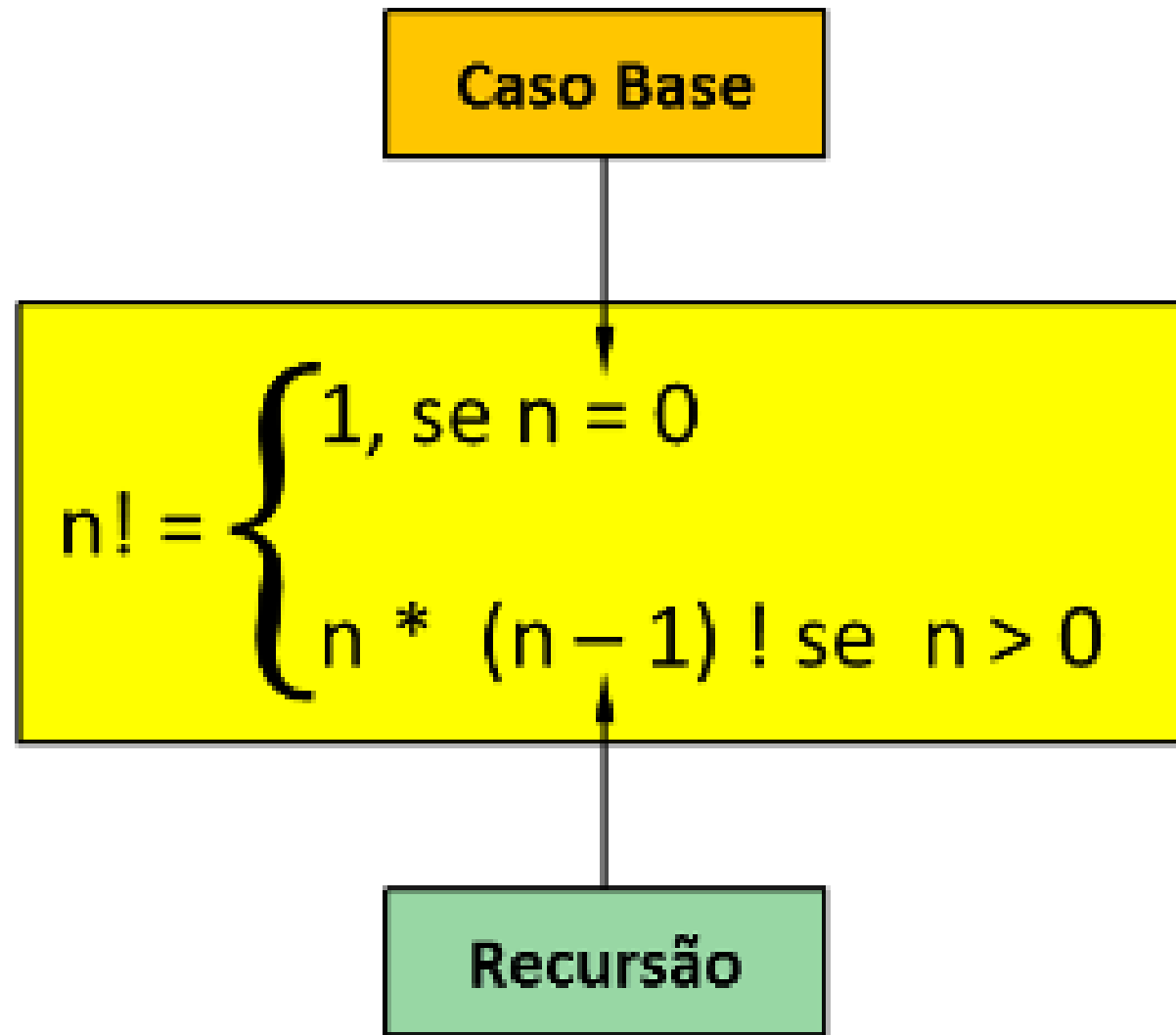
$$\text{Fatorial} (5) = 5 * 24$$

$$\mathbf{5! = 120}$$

```
1 #include <stdio.h>
2 #include <conio.h>
3
4 int fat( int n )
5 {
6     if ( n == 0 )
7         return 1;
8     else
9         return n * fat (n - 1);
10 }
11
```

Funções e Procedimentos recursivos





Com base nestas definições podemos criar uma função para calcular o fatorial de um número qualquer.