

MAC121 - Algoritmos e Estruturas de Dados I

Universidade de São Paulo

Segundo Semestre de 2020

Recursão

Recursão - definição

Dizemos que uma função é **recursiva** se, em sua definição, a própria função é usada.

Alguns exemplos:



$$n! = \begin{cases} 1, & \text{se } n = 0; \\ n * (n - 1)!, & \text{caso contrário.} \end{cases}$$



$$2^n = \begin{cases} 1, & \text{se } n = 0; \\ 2 * 2^{n-1}, & \text{caso contrário.} \end{cases}$$



$$F(n) = \begin{cases} 1, & \text{se } n = 0 \text{ ou } n = 1; \\ F(n - 1) + F(n - 2), & n \geq 2. \end{cases}$$

Funções recursivas

Funções recursivas, em Computação, são aquelas que, no corpo da função, têm uma chamada para elas próprias.

```
long fatorial (int n) {  
    if (n == 0)  
        return (1);  
    return (n * fatorial (n - 1));  
}
```

Funções recursivas são simples, elegantes, fáceis de escrever e fáceis de verificar sua corretude. Como se prova que uma função recursiva está correta?

Como elas funcionam? Vamos simular...

Exemplos de funções recursivas

Problema: Dado um vetor com $n \geq 0$ elementos, devolver o elemento máximo do vetor.

Problema; Dado um inteiro $n > 0$, devolver a soma dos dígitos de n .

Para alguns problemas, recursão é muito natural.

Para saber mais

- ▶ Livro texto, capítulo 2.
- ▶ Material sobre recursão (P. Feofiloff)