



## Exercícios: Recursão

1. Faça uma função recursiva que calcule e retorne o fatorial de um número inteiro N.
2. Faça uma função recursiva que calcule e retorne o N-ésimo termo da sequência Fibonacci. Alguns números desta sequência são: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89...
3. Faça uma função recursiva que permita inverter um número inteiro N. Ex: 123 - 321
4. Faça uma função recursiva que permita somar os elementos de um vetor de inteiros.
5. Crie uma função recursiva que receba um número inteiro positivo N e calcule o somatório dos números de 1 a N.
6. Crie um programa em C, que contenha uma função recursiva que receba dois inteiros positivos k e n e calcule  $k^n$ . Utilize apenas multiplicações. O programa principal deve solicitar ao usuário os valores de k e n e imprimir o resultado da chamada da função.
7. Crie um programa em C que receba um vetor de números reais com 100 elementos. Escreva uma função recursiva que inverta ordem dos elementos presentes no vetor.
8. O máximo divisor comum dos inteiros x e y é o maior inteiro que é divisível por x e y. Escreva uma função recursiva mdc em C, que retorne o máximo divisor comum de x e y. O mdc de x e y é definido como segue: se y é igual a 0, então mdc(x,y) é x; caso contrário, mdc(x,y) é mdc(y, x%y), onde % é o operador resto.
9. Crie uma função recursiva que receba um número inteiro positivo N e calcule o somatório dos números de 1 a N.
10. Escreva uma função recursiva que determine quantas vezes um dígito K ocorre em um número natural N. Por exemplo, o dígito 2 ocorre 3 vezes em 762021192.
11. A multiplicação de dois números inteiros pode ser feita através de somas sucessivas. Proponha um algoritmo recursivo Multip\_Rec(n1,n2) que calcule a multiplicação de dois inteiros.
12. Faça uma função recursiva que receba um número inteiro positivo N e imprima todos os números naturais de 0 até N em ordem crescente.
13. Faça uma função recursiva que receba um número inteiro positivo N e imprima todos os números naturais de 0 até N em ordem decrescente.
14. Faça uma função recursiva que receba um número inteiro positivo par N e imprima todos os números pares de 0 até N em ordem crescente.
15. Faça uma função recursiva que receba um número inteiro positivo par N e imprima todos os números pares de 0 até N em ordem decrescente.
16. A função fatorial duplo é definida como o produto de todos os números naturais ímpares de 1 até algum número natural ímpar N. Assim, o fatorial duplo de 5 é

$$5!! = 1 * 3 * 5 = 15$$

Faça uma função recursiva que receba um número inteiro positivo impar N e retorne o fatorial duplo desse número.

17. O fatorial quádruplo de um número N é dado por:

$$\frac{(2n)!}{n!}$$

Faça uma função recursiva que receba um número inteiro positivo N e retorne o fatorial quádruplo desse número.

18. O superfatorial de um número N é definida pelo produto dos N primeiros fatoriais de N. Assim, o superfatorial de 4 é

$$sf(4) = 1! * 2! * 3! * 4! = 288$$

Faça uma função recursiva que receba um número inteiro positivo N e retorne o superfatorial desse número.

19. O hiperfatorial de um número N, escrito H(n), é definido por

$$H(n) = \prod_{k=1}^n k^k = 1^1 * 2^2 * 3^3 \dots (n-1)^{n-1} * n^n$$

Faça uma função recursiva que receba um número inteiro positivo N e retorne o hiperfatorial desse número.

20. Um fatorial exponencial é um inteiro positivo N elevado à potência de N-1, que por sua vez é elevado à potência de N-2 e assim em diante. Ou seja,

$$n^{(n-1)^{(n-2)} \dots}$$

Faça uma função recursiva que receba um número inteiro positivo N e retorne o fatorial exponencial desse número.

21. Os números tribonacci são definidos pela seguinte recursão

$$\begin{aligned} f(n) &= 0 \text{ se } n = 0 \\ f(n) &= 0 \text{ se } n = 1 \\ f(n) &= 1 \text{ se } n = 2 \\ f(n) &= f(n-1) + f(n-2) + f(n-3) \text{ se } n > 3 \end{aligned}$$

Faça uma função recursiva que receba um número N e retorne o N-ésimo termo da sequência de tribonacci.

22. Os números tetranacci iniciam com quatro termos pré-determinados e a partir daí todos os demais números são obtidos pela soma dos quatro números anteriores. Os primeiros números tetranacci são: 0, 0, 0, 1, 1, 2, 4, 8, 15, 29, 56, 108, 208...

Faça uma função recursiva que receba um número N e retorne o N-ésimo termo da sequência de tetranacci.

23. A sequência de Padovan é uma sequência de naturais P(n) definida pelos valores iniciais

$$P(0) = P(1) = P(2) = 1$$

e a seguinte relação recursiva

$$P(n) = P(n-2) + P(n-3) \text{ se } n > 3$$

Alguns valores da sequência são: 1, 1, 1, 2, 2, 3, 4, 5, 7, 9, 12, 16, 21, 28...

Faça uma função recursiva que receba um número N e retorne o N-ésimo termo da sequência de Padovan.

24. Os números de Pell são definidos pela seguinte recursão

$$\begin{aligned}p(n) &= 0 \text{ se } n = 0 \\p(n) &= 1 \text{ se } n = 1 \\p(n) &= 2p(n-1) + p(n-2) \text{ se } n > 1\end{aligned}$$

25. Os números de Catalan são definidos pela seguinte recursão

$$\begin{aligned}C(n) &= 1 \text{ se } n = 0 \\C(n) &= \frac{2(2n-1)}{n+1} C(n-1) \text{ se } n > 0\end{aligned}$$

Alguns números desta sequência são: 1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786...

Faça uma função recursiva que receba um número N e retorne o N-ésimo número de Catalan.

26. Uma palavra de Fibonacci é definida por

$$\begin{aligned}f(n) &= b \text{ se } n = 0 \\f(n) &= a \text{ se } n = 1 \\f(n) &= f(n-1) + f(n-2) \text{ se } n > 1\end{aligned}$$

Aqui + denota a concatenação de duas strings. Esta sequência inicia com as seguintes palavras:

b, a, ab, aba, abaab, abaababa, abaababaabaab, ...

Faça uma função recursiva que receba um número N e retorne a N-ésima palavra de Fibonacci.

27. Desenvolva algoritmos recursivos para cada um dos seguintes problemas:

- (a) Impressão de um número natural em base binária
- (b) Multiplicação de dois números naturais através de somas consecutivas.
- (c) Inversão de uma string.
- (d) Gerador da sequência:
  - $F(1)=1$
  - $F(2)=2$
  - $F(n) = 2 \cdot F(n-1) + 3 \cdot F(n-2) \leq \text{Fórmula Geral}$

Usando a formula podemos identificar que

$$\begin{aligned}F(3) &= 2 \cdot F(2) + 3 \cdot F(1) \\F(3) &= 4 + 3 = 7\end{aligned}$$

- (e) Verificar se uma palavra é palíndromo
- (f) Busca sequencial em um vetor desordenado que retorna a posição da primeira ocorrência de um elemento procurado usando uma estratégia similar a da busca binária.

**Dica:** Se não é o elemento do meio, procure dos dois lados e retorne o menor dos índices encontrados. Se for o elemento do meio, continue a busca apenas do lado esquerdo, retornando a posição de lá se encontrar e o meio senão encontrar

28. A Multiplicação À Russa Consiste Em:

- (a) Escrever Os Números A E B, Que Se Deseja Multiplicar Na Parte Superior Das Colunas.
- (b) Dividir A Por 2, Sucessivamente, Ignorando O Resto Até Chegar À Unidade, Escrever Os Resultados Da Coluna A.
- (c) Multiplicar B Por 2 Tantas Vezes Quantas Se Haja Dividido A Por 2, Escrever Os Resultados Sucessivos Na Coluna B.
- (d) Somar Todos Os Números Da Coluna B Que Estejam Ao Lado De Um Número Ímpar Da Coluna A.

Exemplo:  $27 * 82$

A	B	Parcelas
27	82	82
13	164	164
6	328	-
3	656	656
1	1312	1312

**Soma = 2214**

Programar Em C Um Algoritmo Recursivo Que Permita Fazer À Multiplicação A Russa De 2 Entradas;

29. Implemente, usando a linguagem C, a função h definida recursivamente por:

$$\begin{aligned}
 h(m,n) &= m+1, \text{ se } n = 1 \\
 h(m,n) &= n+1, \text{ se } m = 1 \\
 h(m,n) &= h(m,n-1) + h(m-1,n), \text{ se } m > 1, n > 1
 \end{aligned}$$

30. A função de Ackerman é definida recursivamente nos números não negativos como segue:

$$\begin{aligned}
 A(m,n) &= n+1 \text{ se } m = 0 \\
 A(m,n) &= A(m-1,1) \text{ se } m > 0 \text{ e } n = 0 \\
 A(m,n) &= A(m-1, A(m,n-1)) \text{ se } m > 0 \text{ e } n > 0
 \end{aligned}$$

Faça uma função recursiva em C para computar a função de Ackerman.

31. Escreva, usando a linguagem C, uma função recursiva, SomaSerie(i,j,k: inteiro): inteiro, que devolva a soma da série de valores do intervalo [i,j], com incremento k.
32. Faça uma função recursiva, em linguagem C, que calcule o valor da série S descrita a seguir para um valor  $n \geq 0$  a ser fornecido como parâmetro para a mesma.

$$S = 2 + \frac{5}{2} + \frac{10}{3} + \dots + \frac{1+n^2}{n}$$

Escreva uma função recursiva em C que exibe todos os elementos em um array de inteiros, separados por espaço.

33. Crie um programa em C, que contenha uma função recursiva para encontrar o menor elemento em um vetor. A leitura dos elementos do vetor e impressão do menor elemento devem ser feitas no programa principal.
34. Escreva, usando a linguagem C, uma função recursiva, `ImprimeSerie(i,j,k: inteiro)`, que imprime na tela a série de valores do intervalo `[i,j]`, com incremento `k`.
35. Dado um número `n` na base decimal, escreva uma função recursiva em C que converte este número para binário.
36. Um palíndromo é uma string que é lida da mesma maneira da esquerda para a direita e da direita para a esquerda. Alguns exemplos de palíndromo são radar e a bola da loba (se os espaços forem ignorados) Escreva uma função recursiva que retorna 1 se a string armazenada no array for um palíndromo e 0, caso contrário. O método deve ignorar espaços e pontuação na string.
37. Uma matriz `maze` de 0s e 1s, de 10X10, representa um labirinto no qual um viajante precisa encontrar um caminho de `maze[0][0]` a `maze[9][9]`. O viajante pode passar de um quadrado para qualquer outro adjacente na mesma fileira ou coluna, mas não pode saltar quadrados nem se movimentar na diagonal. Além disso, o viajante não pode entrar num quadrado contendo um 1. `maze[0][0]` e `maze[9][9]` contêm 0s. Escreva uma rotina recursiva que aceite este labirinto `maze` e imprima uma mensagem informando a inexistência de um caminho através do labirinto, ou que imprima uma lista de posições representando um caminho de `[0][0]` a `[9][9]`.
38. Escreva uma função recursiva que calcule a soma de dois números naturais, através de incrementos sucessivos (Ex.:  $3 + 2 = ++(++3)$ ).
39. Escreva uma função recursiva que calcule a multiplicação de dois números naturais, através de incrementos sucessivos
40. Escreva uma função recursiva que calcule a sequência dada por:

$$\begin{aligned} F(1) &= 1 \\ F(2) &= 2 \\ F(n) &= 2 * F(n-1) + 3 * F(n-2) . \end{aligned}$$

41. Escreva uma função recursiva que dado um número `n`, gere todas as possíveis combinações com as `n` primeiras letras do alfabeto. Ex.: `n = 3`. Resposta: ABC, ACB, BAC, BCA, CAB, CBA
- 42) Escreva uma função recursiva que gere todas as possíveis combinações para um jogo da MegaSena com 6 dezenas
- 43) Uma sequência de Fibonacci generalizada, de `f0` a `f1` é definida como `fibg(f0, f1, 0)`, `fibg(f0, f1, 1)`, `fibg(f0, f1, 2)`, ..., onde:

$$\begin{aligned} \text{fibg}(f0, f1, 0) &= f0 \\ \text{fibg}(f0, f1, 1) &= f1 \\ \text{fibg}(f0, f1, n) &= \text{fibg}(f0, f1, n-1) + \text{fibg}(f0, f1, n-2), \text{ se } n > 1. \end{aligned}$$

Escreva uma função recursiva em C para calcular `fibg(f0, f1, n)`.

42. Escreva uma função recursiva que calcule a soma dos primeiros `n` cubos:  $S(n) = 1^3 + 2^3 + \dots + n^3$
43. Escreva uma função recursiva que calcule a soma dos dígitos de um número inteiro. Por exemplo, se a entrada for 123, a saída deverá ser  $1+2+3 = 6$ .

44. Faça uma função recursiva que permita calcular a média um vetor de tamanho N.
45. Desenvolver um programa que, através de funções adequadas, realize as seguintes tarefas:
- Ler um vetor de valores numéricos fora de ordem.
  - Descobrir o número de ocorrências de um valor X dentro do vetor DESORDENADO, através de uma função recursiva.
  - Ordenar o vetor em ordem crescente através do Quicksort recursivo.
  - Fazer a busca por um valor X dentro do vetor, através da Busca Binária recursiva, retornando a posição onde o elemento foi encontrado.
  - Imprimir o vetor.