

- 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 algoritmo em Python, 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 Python, que retorna 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) 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.
- 11) Uma palavra de Fibonacci é definida por

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

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.