

Exercícios sobre recursão. Implemente os problemas na linguagem Java.

- 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 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 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 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) 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.
- 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) Faça uma função recursiva que receba um número inteiro positivo N e imprima todos os números naturais de 0 até N em ordem crescente.
- 12) Faça uma função recursiva que receba um número inteiro positivo N e imprima todos os números naturais de 0 até N em ordem decrescente.
- 13) 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.
- 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 decrescente.
- 15) 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 ímpar N e retorne o fatorial duplo desse número.

- 16) 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.
- 17) 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.
- 18) 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) Verificar se uma palavra é palíndromo
- 19) Escreva uma função recursiva, `SomaSerie(i,j,k: inteiro): inteiro`, que devolva a soma da série de valores do intervalo $[i,j]$, com incremento k .
- 20) Escreva uma função recursiva que exibe todos os elementos em um array de inteiros, separados por espaço.
- 21) Crie um programa 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.
- 22) Escreva uma função recursiva, `ImprimeSerie(i,j,k: inteiro)`, que imprime na tela a série de valores do intervalo $[i,j]$, com incremento k .
- 23) Dado um número n na base decimal, escreva uma função recursiva que converte este número para binário.
- 24) 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.
- 25) Uma matriz `maze` de 0s e 1s, de 10×10 , 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].

- 26) Escreva uma função recursiva que calcule a soma de dois números naturais, através de incrementos sucessivos (Ex.: $3 + 2 = ++(++3)$).
- 27) Escreva uma função recursiva que calcule a multiplicação de dois números naturais, através de incrementos sucessivos
- 28) Escreva uma função recursiva que calcule a sequência dada por: $F(1) = 1$ $F(2) = 2$
 $F(n) = 2 * F(n-1) + 3 * F(n-2)$.
- 29) 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
- 30) Escreva uma função recursiva que gere todas as possíveis combinações para um jogo da MegaSena com 6 dezenas.
- 31) 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 $1+2+3 = 6$.
- 32) Faça uma função recursiva que permita calcular a média um vetor de tamanho N .