

Lista 3: Recursão

Todos exercícios devem ser feitos com funções recursivas

- 1) Faça uma função recursiva que gere uma sequência de números de 1 a 50.
- 2) Faça um função recursiva que some todos elementos de um vetor.
- 3) Faça uma função recursiva de incremento. Dado um número inteiro qualquer, menor que 100, a função recursiva deverá incrementar o número em uma unidade até que o número seja igual a 100.
- 4) 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.
- 5) Faça uma função recursiva que receba um número inteiro positivo ímpar N e retorne o fatorial duplo desse número. O fatorial duplo é definido como o produto de todos os números naturais ímpares de 1 até algum número natural ímpar N. Assim, o fatorial duplo de 5 é: $5!! = 1 * 3 * 5 = 15$.
- 6) Faça uma função recursiva que receba um número inteiro positivo N e retorne o superfatorial desse número. O superfatorial de um número N é definida pelo produto dos N primeiros fatoriais de N. Assim, o superfatorial de 4 é: $sf(4) = 1! * 2! * 3! * 4! = 288$.
- 7) Faça uma função recursiva para calcular os números de Pell. Os números de Pell são definidos pela seguinte recursão:
$$p(n) = 0 \text{ se } n = 0;$$
$$p(n) = 1 \text{ se } n = 1;$$
$$p(n) = 2p(n-1) + p(n-2) \text{ se } n > 1;$$
- 8) A série de Ricci difere da série de Fibonacci porque os dois primeiros termos são fornecidos pelo usuário. Os demais termos são gerados da mesma forma que a série de Fibonacci. Faça um programa que imprima os N primeiros termos da série de Ricci e a soma dos termos impressos, sabendo-se que para existir esta série serão necessários pelo menos três termos.

- 9) A série de Fetuccine é gerada da seguinte forma: os dois primeiros termos são fornecidos pelo usuário; a partir daí, os termos são gerados com a soma ou subtração dos dois termos anteriores, ou seja:

$$A_i = A_{i-1} + A_{i-2} \text{ para } i \text{ ímpar}$$

$$A_i = A_{i-1} - A_{i-2} \text{ para } i \text{ par}$$

Crie uma função recursiva que gere a série Fetuccine até o N-ésimo termo. O usuário deverá informar qual é o N-ésimo termo.

- 10) Seja a seguinte série: 1, 4, 9, 16, 25, 36, ... Escreva um programa que gere esta série até o N-ésimo termo. Este N-ésimo termo é digitado pelo usuário.
- 11) Seja a seguinte série: 1, 4, 4, 2, 5, 5, 3, 6, 6, 4, 7, 7, ... Escreva um programa que seja capaz de gerar os N termos dessa série. Esse número N deve ser lido do teclado.
- 12) Faça uma função recursiva que receba um número N e retorne o N-ésimo termo da sequência de Tetranacci. 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...
- 13) 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.