

Algoritmos e Estruturas de Dados I

Quinta Lista de Exercícios - Funções

1. Faça uma função para verificar se um número é positivo ou negativo. Sendo que o valor de retorno será 1 se positivo, -1 se negativo e 0 se for igual a 0.
2. Faça uma função para verificar se um número é um quadrado perfeito. Um quadrado perfeito é um número inteiro não negativo que pode ser expresso como o quadrado de outro número inteiro. Ex: 1, 4, 9...
3. Faça uma função que receba uma temperatura em graus Celsius e retorne-a convertida em graus Fahrenheit. A fórmula de conversão é: $F = C * (9.0/5.0) + 32.0$, sendo F a temperatura em Fahrenheit e C a temperatura em Celsius.
4. Elabore uma função que receba três notas de um aluno como parâmetros e uma letra. Se a letra for A, a função deverá calcular a média aritmética das notas do aluno; se for P, deverá calcular a média ponderada, com pesos 5, 3 e 2.
5. Faça uma função que receba dois valores numéricos e um símbolo. Este símbolo representará a operação que se deseja efetuar com os números. Se o símbolo for + deverá ser realizada uma adição, se for - uma subtração, se for / uma divisão e se for * será efetuada uma multiplicação.
6. Crie um programa que receba três valores (obrigatoriamente maiores que zero), representando as medidas dos três lados de um triângulo. Elabore funções para:
 - a) Determinar se eles lados formam um triângulo, sabendo que:
 - O comprimento de cada lado de um triângulo é menor do que a soma dos outros dois lados.
 - b) Determinar e mostrar o tipo de triângulo, caso as medidas formem um triângulo. Sendo que:
 - Chama-se equilátero o triângulo que tem três lados iguais.
 - Denominam-se isósceles o triângulo que tem o comprimento de dois lados iguais.
 - Recebe o nome de escaleno o triângulo que tem os três lados diferentes.
7. Escreva uma função que gera um triângulo de altura e lados n e base $2*n-1$. Por exemplo, a saída para n = 6 seria:

```
      *
     ***
    *****
   ********
  **********
 **********
 **********
```

8. Faça uma função que receba um inteiro N como parâmetro, calcule e retorne o resultado da seguinte série:
$$S = 2/4 + 5/5 + 10/6 + \dots + (N^2 + 1)/(N + 3)$$
9. Faça uma função chamada 'simplifica' que recebe como parâmetro o numerador e o denominador de uma fração. Esta função deve simplificar a fração recebida dividindo o numerador e o denominador pelo maior fator possível. Por exemplo, a fração 36/60

simplifica para $\frac{3}{5}$ dividindo o numerador e o denominador por 12. A função deve modificar as variáveis passadas como parâmetro.

10. Faça uma função não-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: $5!! = 1 * 3 * 5 = 15$.

11. Faça uma função não-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 $sf(4) = 1! * 2! * 3! * 4! = 288$.

12. Faça uma função que receba um vetor de inteiros e retorne quantos valores pares ele possui.

13. Faça uma função que receba um vetor de reais e retorne a média dele.

14. Crie um programa contendo as seguintes funções que recebem um vetor V números reais como parâmetro:

- Impressão normal do vetor.
- Impressão inversa.
- Função que retorna o maior e o menor elemento do vetor.

15. Faça uma função que receba uma matriz 4 x 4 e retorne quantos valores maiores do que 10 ela possui.

16. Faça uma função que receba uma matriz de 3 x 3 elementos. Calcule e retorne a soma dos elementos que estão abaixo da diagonal principal.

17. Escreva uma função que recebe uma matriz quadrada de ordem 10 e calcule a sua transposta. Crie ainda uma função para imprimir as matrizes.

18. Faça uma função que recebe, por parâmetro, uma matriz $A[7][6]$ e uma coluna N e retorne a soma dos elementos dessa coluna.

19. Escreva uma função que compare e retorne verdadeiro, caso uma string seja anagrama da outra, e falso, caso contrário.

20. Crie uma função que calcula o comprimento de uma string e que possui o seguinte protótipo: `void tamanho(char *str, int *strsize)`.

21. Crie uma função que compara duas strings e que retorna se elas são iguais ou diferentes.

22. Faça uma função que receba duas strings e retorne a intercalação letra a letra da primeira com a segunda string. A string intercalada deve ser retornada na primeira string.

23. Um racional é qualquer número da forma p/q , sendo p inteiro e q inteiro não nulo. É conveniente representar um racional por um registro:

```
struct racional{
    int p, q;
};
```

Vamos convencionar que o campo q de todo racional é estritamente positivo e que o máximo divisor comum dos campos p e q é 1. Escreva:

- a. uma função reduz que receba inteiros a e b e devolva o racional que representa a/b ;
- b. uma função neg que receba um racional x e devolva o racional -x;
- c. uma função soma que receba racionais x e y e devolva o racional que representa a soma de x e y;
- d. uma função mult que receba racionais x e y e devolva o racional que representa o produto de x por y;
- e. uma função div que receba racionais x e y e devolva o racional que representa o quociente de x por y;

24. Foi realizada uma pesquisa de algumas características físicas de cinco habitantes de certa região. De cada habitante foram coletados os seguintes dados: sexo, cor dos olhos (A – Azuis ou C – Castanhos), cor dos cabelos (L – Louros, P – Pretos ou C – Castanhos) e idade.

- Faça uma função que leia esses dados em um vetor.
- Faça uma função que determine a média de idade das pessoas com olhos castanhos e cabelos pretos.
- Faça uma função que determine e devolva ao programa principal a maior idade entre os habitantes.
- Faça uma função que determine e devolva ao programa principal a quantidade de indivíduos do sexo feminino cuja idade está entre 18 e 35 (inclusive) e que tenham olhos azuis e cabelos louros.

25. Crie uma função recursiva que receba um número inteiro positivo N e calcule o somatório dos números de 1 a N.

26. Escreva uma função recursiva que calcule a soma dos primeiros n cubos: $S(n) = 1^3 + 2^3 + \dots + n^3$.

27. 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.

28. 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.

29. Escreva uma função recursiva `SomaSerie(i,j,k)`. Esta função devolve a soma da série de valores do intervalo $[i,j]$, com incremento k.

30. 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: $5!! = 1 * 3 * 5 = 15$.

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

32. 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...

33. 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 $\text{mdc}(x,y)$ é x ; caso contrário, $\text{mdc}(x,y)$ é $\text{mdc}(y, x\%y)$, onde $\%$ é o operador resto.

34. Faça uma função recursiva para calcular os números de Pell. Os números de Pell são definidos pela seguinte recursão:

$p(n) = 0$ se $n = 0$;

$p(n) = 1$ se $n = 1$;

$p(n) = 2p(n-1) + p(n-2)$ se $n > 1$;

35. Crie um programa que receba um vetor de números reais com 100 elementos. Escreva uma função recursiva que inverta a ordem dos elementos presentes no vetor.