

LISTA DE RECURSÃO

1. Escreva uma função recursiva que exiba, em ordem decrescente, os números do intervalo $[1, n]$ na saída padrão.
2. Escreva uma função recursiva que exiba, em ordem crescente, os números do intervalo $[1, n]$ na saída padrão.
3. Faça uma função recursiva que lê do usuário um vetor e informa a quantidade de elementos que o vetor possui.
4. Faça uma função recursiva que recebe um vetor e armazene em outro vetor somente os elementos que não repetem.
5. Escreva um algoritmo recursivo que calcule a soma dos elementos de um vetor
6. Escreva um procedimento recursivo que recebe uma string como parâmetro e a exibe invertida.
7. Faça uma função recursiva que seja capaz de ordenar em ordem crescente um vetor de inteiros.
8. Faça um programa que leia do usuário um vetor de inteiros, no seu programa deve conter uma função recursiva que verifica se determinado número informado pelo usuário está no vetor.
9. Escreva uma função recursiva para calcular os valores, mínimo e máximo de um vetor de n elementos inteiros.
10. Escreva uma função recursiva para calcular n^x .
11. Escreva uma função recursiva **sorted** que, dado um vetor e seu tamanho, retorne 1 se o mesmo estiver ordenado, ou 0 caso contrário.
12. Faça uma função recursiva que gere a seguinte soma:
Entrada: 3
Saída: $3 + 2 + 1 + 1 + 2 + 3 = 12$.

13. Considere a seguinte função recursiva:

```
int f( int n )
{
    If ( n < 4)
        return 3 * n)
    else
        return 2 * f ( n - 4) + 5;
}
```

Quais os valores para $f(3)$ e $f(7)$?

14. Escreva uma função recursiva que conte quantos elementos ímpares existe em um vetor fornecido pelo usuário.

15. A função abaixo promete encontrar o valor de um elemento máximo de $v[0 .. n-1]$. A função cumpre a promessa?

```
int maxi (int n, int v[])
{
    int j, m = v[0];
    for(j = 1; j < n; j++)
        if( v[j - i] < v[ j ])
            m = v[ j ];
    return m;
}
```

16. Escreva uma função recursiva que calcule a soma dos dígitos de um inteiro positivo n . A soma dos dígitos de 132, por exemplo, é 6.

17. Faça uma função recursiva que calcula a soma dos N primeiros números

Ex: Entrada 120

Saída 7260

18. Escreva uma função recursiva que calcule o piso do logaritmo de N na base 2.

19. Qual o resultado da execução do programa abaixo?

```
int ff(int n)
{
    if( n == 1)
        return 1;
    if(n % 2 == 0)
        return ff( n / 2);
    return ff(( n -1) / 2) + ff ((n + 1) / 2);
}

int main(void)
{
    printf(" %d", ff( 7 ));
    return EXIT_SUCESS;
}
```

20. A seguinte função calcula o maior divisor comum dos inteiros positivos m e n .
Escreva uma função recursiva equivalente.

```
int Euclides ( int m, int n)
{
    int r;
    do{
        r = m % n;
        m = n;
        n = r;
    }while (r != 0);
    Return m;
}
```

21. Critique a seguinte função recursiva

```
int xx ( int n)
{
    If( n == 0 )
        return 0;
    else
        return xx (n / 3 + 1) + n;
}
```

22. Escreva uma função sega-sequencia: ela deva receber um inteiro n positivo e retornar a lista (1 -1 2 -2 3 -3 n - n).

23. Escreva um programa que lê do usuário um vetor de inteiros e um inteiro n , faça uma função recursiva que receba o vetor e o n e mostre todos os elementos maiores que n existentes no vetor.

Ex: (maiores que 10 (4 6 3, 3 15 3 10 7)) → (30 15).

24. Escreva um programa que lê do usuário um vetor de inteiros e um inteiro n , faça uma função que verifica se existe uma única ocorrência de n entre os elementos do vetor.

Ex: única-ocorrencia 2 [1,2,3,2] = FALSE única-ocorrencia 2 [3,1] = FALSE

única-ocorrencia 2 [2] = TRUE.

25. Implementar uma função recursiva para calcular o máximo divisor comum de dois números naturais A e B .

se o resto (A por B) = 0 então $mdc = B$;

senão chamar a função passando ($B, A \bmod B$)

26. Implementar uma função recursiva para calcular a divisão inteira de A por B, utilizando apenas subtrações.

se $A - B < B$ então $f = 1$

senão $f = f(A - B, B) + 1$

27. Qual o resultado impresso pelo seguinte programa?

```
#include<stdio.h>
int digitsum(int n)
{
    if (n>0)
        return (n%10) +digitsum(n/10);
    else
        return 0;
}

int digitalroot(int n)
{
    if(n>9) return
        digitalroot(digitsum(n));
    else
        return n;
}

main()
{
    printf("%d %d %d",
        digitalroot(123),
        digitalroot(1234),
        digitalroot(12345));
}
```

28. Faça uma função recursiva que recebe uma string e a imprima todos com letra maiúscula. Pode ser útil saber os seguintes códigos da tabela ASCII: a = 97, z = 122, A = 65, Z = 90.

Ex:

Entrada: "ProgrAMAÇÃO de ComPutadores II"

Saída: "PROGRAMAÇÃO DE COMPUTADORES II".

29. Faça uma função que recebe uma string qualquer e imprima somente a primeira letra de cada palavra maiúscula.

Ex:

Entrada: "FulANo bELtrAnO siLVA"

Saída: "Fulano Beltrano Silva"

30. Escrever um programa que gera a codificação em código Morse de um inteiro positivo fornecido pelo usuário. A codificação segue abaixo.

0 = _ _ _ _ _

1 = . _ _ _ _

2 = .. _ _ _

3 = ... _ _

4 = _

5 =

6 = _

7 = _ _

8 = _ _ _

9 = _ _ _ _ .

Ex: para entrada igual 576 sua saída deve ser a seguinte:

..... _ _ _

31. Escreva a função recursiva **comb(n,k)** que representa o número de grupos distintos com k pessoas que podem ser formados a partir de n pessoas:

$$Comb(n, k) = \begin{cases} n & \text{se } k = 1 \\ 1 & \text{se } k = n \\ Comb(n - 1, k - 1) + Comb(n - 1, k) & \text{se } 1 < k < n \end{cases}$$

32. Escreva uma função iterativa e outra recursiva que retorne o menor número de um vetor de inteiros.

33. Compare as funções e descreva as vantagens e desvantagens de cada uma.

34. Escreva uma função recursiva que receba por parâmetro um vetor de pessoas e liste-os por ordem:

- a) Crescente do índice do vetor;
- b) decrescente do índice do vetor;

35. Elabore uma rotina recursiva que some todos elementos de uma matriz quadrada 20 x 20 de inteiros.

36. Faça uma função que inverta os elementos de um vetor de inteiros com 10 posições.

Ex: vetor original: 10 12 4 68 39 11 20 9 40 33
 resultado: 33 40 9 20 11 39 68 4 12 10

37. Crie uma rotina que receba um valor inteiro como parâmetro, e informe quantas ocorrências deste valor existem em uma matriz 10 x 10 de inteiros.
38. Escreva uma função recursiva que verifica de um inteiro n é um número primo ou não.
39. Escreva uma função recursiva eficiente que receba inteiros positivos k e n e calcule k^n . (Suponha que k^n cabe em um int.) Quantas multiplicações sua função executa aproximadamente?
40. Calcular a representação binária de um número consiste em realizar sucessivas divisões por 2 e, quando seu resto for menor que 2, imprimir do último para o primeiro, todos os restos das divisões. Por exemplo.

$$6 / 2 = 3 \text{ (resto 0)} \Rightarrow 3 / 2 = 1 \text{ (resto 1)} \Rightarrow 1 / 2 = 0 \text{ (resto 1)}$$

$$6 \text{ em binário} = 110$$

Desenvolva uma função recursiva para calcular a representação binária de um inteiro positivo n qualquer.