

Lista de Recursividade

Estrutura de Dados

Prof. Roberto Cabral

9 de setembro de 2022

1. Crie um programa em C, 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.
2. Crie um programa em C que receba um vetor de números reais com n elementos. Escreva uma função recursiva que inverta a ordem dos elementos presentes no vetor.
3. 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.
4. A multiplicação de dois números inteiros pode ser feita através de somas sucessivas. Implemente um algoritmo recursivo `multRec`(n_1, n_2) que calcule a multiplicação de dois inteiros.
5. Faça uma função recursiva que receba um número inteiro positivo n e imprima todos os números naturais de 0 até n em ordem crescente.
6. Faça uma função recursiva que receba um número inteiro positivo n e imprima todos os números naturais de 0 até n em ordem decrescente.
7. Faça uma função recursiva que receba um número inteiro positivo n e imprima todos os números pares de 0 até n em ordem decrescente.
8. Faça uma função recursiva que receba um número inteiro positivo n e imprima todos os números primos de 0 até n em ordem decrescente.
9. A função fatorial duplo é definida como o produto de todos os números naturais de 1 até algum número natural ímpar n . Assim, o fatorial duplo de 5 é: $5!! = 1 * 3 * 5 = 15$. Faça uma função que receba um número inteiro positivo ímpar n e retorne o fatorial duplo desse número.

10. O fatorial quádruplo de um número inteiro n é dado por $\frac{(2n)!}{n!}$. Faça uma função recursiva que receba um número inteiro positivo n e retorne o fatorial quádruplo desse número.
11. O superfatorial de um número inteiro n é definido pelo produto dos n primeiros fatoriais de n . Assim, o superfatorial de 4 é: $\mathbf{sf}(4) = 1! * 2! * 3! * 4! = 288$. Faça uma função recursiva que receba um número inteiro positivo n e retorne o superfatorial desse número.
12. O hiperfatorial de um número n , escrito $H(n)$, é definido por:

$$H(n) = \prod_{k=1}^n k^k = 1^1 * 2^2 * 3^3 \dots (n-1)^{n-1} * n^n.$$

Faça uma função recursiva que receba um número positivo n e retorne o hiperfatorial desse número.

13. Um fatorial exponencial é um inteiro positivo n elevado à potência de $(n-1)$, que por sua vez é elevado à potência de $(n-2)$ e assim em diante. Ou seja:

$$n^{(n-1)^{(n-2)} \dots}$$

faça uma função recursiva que receba um número inteiro positivo n e retorne o fatorial exponencial desse número.

14. Crie um programe em C, 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.
15. Dado um número n na base decimal, escreva uma função recursiva que converte este número para binário.
16. A multiplicação à Russa consiste em:
- (a) Escrever os números A e B, que se deseja multiplicar na parte superior das colunas.
 - (b) Dividir A por 2, sucessivamente, ignorando o resto até chegar à unidade, escrever os resultados da coluna A.
 - (c) Multiplicar B por 2 tantas vezes quantas se haja dividido A por 2, escrever os resultados sucessivos na coluna B.
 - (d) Somar todos os números da coluna B que estejam ao lado de um número ímpar da coluna A.

Exemplo: 27×82

A	B	Parcelas
27	82	82
13	164	164
6	328	-
3	656	656
1	1312	1312

Soma: 2214

Escreva uma função recursiva que permita fazer a multiplicação à russa de duas entradas.

17. Implemente uma função recursiva para resolver o seguinte problema: dado um vetor com n números inteiros positivos e um outro número inteiro positivo x , determine se existem ou não dois elementos cuja soma é igual a x . (preferencialmente faça um algoritmo com complexidade $\theta(n \log n)$)
18. Faça uma função recursiva que permita calcular a média de um vetor de tamanho n .
19. Escreva uma função recursiva que calcule a soma dos dígitos de um número inteiro. Por exemplo, se a entrada for 357, a saída deverá ser $3+5+7 = 15$.
20. Faça uma função recursiva que calcule e retorne o fatorial de um número inteiro n .
21. 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, ...
22. Faça uma função recursiva que inverta uma string de tamanho n . Ex.: *casa* \implies *asac*.
23. Faça uma função recursiva que permita somar os elementos de um vetor de inteiros.
24. Crie uma função recursiva que receba um número inteiro positivo n e calcule o somatório dos números de 1 a n .
25. Escreva uma função recursiva que receba como entrada uma string e retorne 1 se a string for palíndroma e 0 caso contrário.
26. Escreva uma função recursiva que receba como entrada uma string e um caractere c e retorne a quantidade de vezes que c apareceu na string.
27. Faça uma função recursiva que receba um vetor ordenado e um inteiro x e retorne 1 se x pertencer ao vetor e 0 caso contrário. Seu programa deve ter complexidade $\log n$.