

Lista de Exercícios (Lógica de Programação)

Para a resolução desses exercícios busque encontrar soluções cujo custo computacional (números de testes, multiplicações, etc.) seja o menor possível. A maioria tem soluções muito mais rápidas que o trivial.

1. Escreva uma função que faça a multiplicação de dois números inteiros usando somas e subtrações.
2. Escreva uma função que faça a divisão de dois números inteiros usando somas e subtrações.
3. Escreva uma função que calcule a potência de um número inteiro x^y usando multiplicações.
4. Escreva uma função que receba um número inteiro e retorne o inverso desse número. Por exemplo, se a função receber 123456 deve retornar 654321.
5. Escreva uma função que receba um número inteiro e retorne 1 se esse for primo e 0 caso contrário.
6. Escreva uma função que receba um número inteiro e retorne seu fatorial.
7. Escreva uma função que receba uma *string* (cadeia de caracteres) e retorne 1 se essa for um palíndromo e 0 caso contrário. Um palíndromo é uma palavra ou frase que se lê da mesma maneira nos dois sentidos - normalmente, da esquerda para a direita ou ao contrário, da direita para a esquerda. Por exemplo, OVO; ARARA; SOCORRAM-ME, SUBI NO ÔNIBUS EM MARROCOS.
8. Escreva uma função que imprima todos os anagramas de uma palavra que contenha letras não repetidas. Uma anagrama são todas as possíveis permutações das letras da palavra original que forma ou não palavras reais. Por exemplo, a palavra cat, geraria os anagramas: cat, cta, atc, act, tca, tac.
9. Escreva uma função que receba dois argumentos, um vetor contendo valores inteiros, e um outro argumento inteiro maior que o menor número contido nesse vetor. Essa função deve imprimir quais dos valores contidos no vetor devem ser somados para se obter o valor do segundo argumento. Por exemplo, se a função receber o vetor [11, 8, 7, 6, 5] e o segundo argumento 20, essa deveria imprimir os números 8, 7, 5 uma vez que $8 + 7 + 5 = 20$. Se não for possível obter o valor final, não imprima nada.
10. Escreva uma função que dados dois números inteiros x e y , com $x < y$, determine quantos números primos existem entre esses.
11. Escreva uma função que dados dois números inteiros x e y , com $x < y$, determine quantas vezes um determinado algoritmo z aparece em todos os números definidos no intervalo entre x e y .
12. Escreva uma função que receba um vetor de números inteiros e retorne o MMC desses.
13. Faça um programa para receber dois números do usuário e calcular o seu MDC utilizando o método de Euclides. O programa deve continuar pedindo dois números até que 0 e 0 sejam fornecidos.
14. Escreva uma função que receba um número inteiro e imprima seus fatores primos.
15. Escreva uma função que dado um ângulo em graus, retorne seu seno. Para isso use as séries de MacLaurin. Quando um ângulo α é dado em radianos, a série de MacLaurin para seno é:

$$\sin \alpha = \alpha - \frac{\alpha^3}{3!} + \frac{\alpha^5}{5!} - \frac{\alpha^7}{7!} + \dots + (-1)^n \frac{\alpha^{2n+1}}{(2n+1)!}$$

$$\alpha_{\text{radianos}} = \frac{\pi * \alpha_{\text{graus}}}{180}$$

16. Escreva uma função que dado um ângulo em graus, retorne seu coseno. Para isso use as séries de MacLaurin. Quando um ângulo α é dado em radianos, a série de MacLaurin para coseno é:

$$\cos \alpha = 1 - x + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!}$$

$$\alpha_{\text{radianos}} = \frac{\pi * \alpha_{\text{graus}}}{180}$$

17. Escreva uma função que receba um inteiro e imprima esse valor em algarismos romanos. Por exemplo, se essa função receber 13 deve imprimir *XIII*.
18. Escreva uma função que receba um número inteiro positivo x e retorne a soma de todos os inteiros positivos entre 1 e x .
19. Escreva uma função que receba um número positivo e imprima sua representação binária.
20. Escreva uma função que receba duas matrizes e seus respectivos tamanhos de linhas e colunas e imprima, quando possível, o resultado da multiplicação dessas.
21. Escreva uma função que receba três valores inteiros e retorne 0 se esses não forma um triângulo, 1 se o triângulo é equilátero, 2 se é isósceles e 3 se é escaleno.
22. Escreva uma função que receba um número e retorne o menor inteiro positivo cujo quadrado é superior ao valor dado.
23. Escreva uma função que receba dois números x e y , com $x < y$, e imprima quais números nesse intervalo que divididos por 13 dão um resto igual a 5.
24. Escreva uma função que receba dois números inteiros n e r e retorne o número de combinações de n objetos tomados r de cada vez. O número de combinações é dado pela fórmula:

$$c = \frac{n!}{r! * (n - r)!}$$

25. A famosa conjectura de Goldbach diz que todo inteiro par maior que 2 é a soma de dois números primos (você aceita isso?). Testes foram feitos sem contudo encontrar um contra-exemplo. Escreva um programa mostrando que a afirmação é verdadeira para todo número par entre 700 e 1100. O programa deve imprimir cada número e seus correspondentes primos.
26. Escreva uma função recursiva que retorne a representação binária de um número inteiro.
27. Seja N um número quadrado perfeito. Se somarmos os números ímpares consecutivos ($1 + 3 + 5 + 7 + 9 + \dots$) até que esta soma seja igual a N , o número M de termos somados será igual a raiz quadrada de N . Exemplo: $N = 1616 = 1 + 3 + 5 + 7M = 4$ termos. Logo, a raiz quadrada de 16 é 4. Fazer um programa em C para ler um número inteiro e positivo N e responder se N é quadrado perfeito.
28. Escreva uma função que calcule a série de Fibonacci. Observação: Uma seqüência é dita de Fibonacci quando os dois primeiros termos forem 0 e 1 e todos os termos consecutivos tenham valor igual à soma dos dois termos anteriores.
Exemplo: 011235813213455...
29. O elemento *minimax* de uma matriz é o menor elemento da linha que contém o maior elemento da matriz. Desenvolva um algoritmo para encontrar o elemento *minimax*.
30. Dados os vetores ordenados M e N desenvolva um algoritmo que devolva o k -ésimo menor elemento de $N + M$.
31. Crie um função que faça a ordenação de um vetor.
32. A conversão de graus Fahrenheit para Celsius é obtida por $C = 5(F - 32)/9$. Fazer um algoritmo que calcule e escreva uma tabela de graus Celsius em função de graus Fahrenheit, variando um a um de 50 a 150 graus Fahrenheit.

33. Faça uma função que calcule todos os divisores de um número.
34. Crie um programa para mostrar todos os números perfeitos entre 1 e 100.
35. Implemente funções que façam o mesmo que as funções do C:
- `strlen()` - retorna o tamanho da string.
 - `strcmp()` - diz se as duas strings são iguais ou não.
 - `strncpy()` - copia os n primeiros caracteres para a string destino.
 - `strcat()` - concatena duas strings.
 - `strfind()` - procura todas as ocorrências de uma string dentro de outra. Deve informar todas as posições iniciais onde ela encontrou.
36. Torre de Hanói: considerando 3 torres, o objetivo é transferir 3 discos que estão na torre A para a torre C, usando uma torre B como auxiliar. Somente o último disco de cima de uma pilha pode ser deslocado para outra, e um disco maior nunca pode ser colocado sobre um menor. Implementar uma função recursiva que mostra a sequência de movimentos para resolver o problema da Torre de Hanói.
37. Escreva um programa que leia uma matriz $n \times m$ do usuário e a transforme em um vetor unidimensional de $n \cdot m$ posições.
38. Um dos sistemas de encriptação mais antigos é atribuído a Júlio César: se uma letra a ser encriptada é a letra de número N do alfabeto, substitua-a com a letra $(N + K)$, onde K é um número inteiro constante (César utilizava $K = 3$). Usualmente consideramos o espaço como zero e todos os cálculos são realizados com módulo-27. Dessa forma, para $K = 1$ a mensagem *Ataque ao amanhecer* se torna *bubrfabpabnboifdfs*. Faça um programa que receba como entrada uma mensagem e um valor de K e retorne a mensagem criptografada pelo código de César. Fraquezas: apenas 26 chaves possíveis.
39. Calcular a soma dos elementos de uma matriz numérica quadrada qualquer dada, que estão acima da diagonal principal.
40. Escreva um programa fazendo o uso de *struct*. Você deverá criar uma *struct* chamada Ponto, contendo apenas a posição x e y (inteiros) do ponto. Declare 2 pontos, leia a posição (coordenadas x e y) de cada um e calcule a distância entre eles. Apresente no final a distância entre os dois pontos.