



**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO**

COM06842 – PROGRAMAÇÃO I

LISTA DE EXERCÍCIOS

1. Faça uma função em C que receba e exiba o seu nome.
2. Faça uma função em C que recebe uma string S e um inteiro positivo N e exiba a string S por N vezes seguidas na tela.
3. Faça uma função em C chamada CUBO que receba um valor do tipo *float* e retorne a potência elevado a 3 do mesmo.
4. Faça uma função em C chamada DOBRO que receba um valor do tipo *float* e retorne o dobro do mesmo.
5. Faça uma função em C que receba três números e usando a função DOBRO da questão anterior imprima o dobro de cada um dos números. Use no mínimo 2 funções.
6. Faça uma função em C chamada TROCA que receba duas variáveis inteiras (X e Y) e troque o conteúdo entre elas.
7. Faça uma função em C que receba uma string S e retorne o número de espaços existentes na mesma.
8. Faça uma função em C que receba uma string S e um valor inteiro N e retorne os N primeiros caracteres da string S.
9. Faça uma função em C chamada SINAL que receba como parâmetro um valor N inteiro e escreva a palavra POSITIVO se N for um número maior que zero, NEGATIVO se N for menor que zero, ou ZERO se N for igual a zero.
10. Faça uma função em C que leia um número inteiro e, usando a função SINAL (criado na questão anterior), mostre se ele é maior, menor ou igual a zero.
11. Faça uma função em C chamada METADE que divida um valor do tipo *float* (passado como parâmetro) pela metade.
12. Faça uma função em C que leia um vetor A de 30 elementos *float* e, usando o procedimento METADE (criado na questão anterior), divida todos seus elementos pela metade.
13. Faça uma função em C chamada MEDIA que retorne a média aritmética de três valores reais (X, Y e Z) passados como parâmetros.



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO

14. Faça uma função em C chamada PRIMO que verifica se um número é primo.
15. Faça uma única função em C que retorna o MENOR, o MAIOR e a SOMA de um vetor com 50 elementos passados por parâmetro. A impressão dos resultados deve ser feita no programa *main*.
16. Faça um programa em C que, para um número indeterminado de alunos, faça para cada uma deles:
 - ler o nome e as três notas do aluno (a leitura do nome FIM indica o fim dos dados - *flag*);
 - calcular a média do aluno (usando a função MEDIA criada na questão anterior);
 - exibir o nome e a média do aluno.
17. Faça uma função em C chamada POTENCIA que realize a operação de potenciação e que contenha os seguintes parâmetros formais:

VALOR: número inteiro que se deseja elevar à potência.

POT: potência elevada

Ex: Na chamada da função POTENCIA(5,3), o resultado seria 125. Na chamada da função POTENCIA(2,0), o resultado seria 1.
18. Faça uma função em C chamada SEG para receber uma medida de tempo expressa em Horas, Minutos e Segundos e retornar esta medida convertida apenas para segundos.
19. Faça uma função em C chamada HMS para receber uma medida de tempo expressa apenas em segundos em retornar esta medida convertida para horas, minutos e segundos.
20. Faça uma função em C que leia duas medidas de tempo (expressas em horas, minutos e segundos) e, usando a função SEG e o procedimento HMS, calcule e exiba a diferença (também em horas, minutos e segundos) entre elas.
21. Faça uma função em C para leitura de um valor inteiro entre 1 e 100. Se o usuário fornecer um valor fora desta faixa, deve-se repetir o processo até que seja fornecido um valor válido.
22. Faça uma função em C para ler duas matrizes e apresentar a soma. Utilize funções para ler, somar e escrever as matrizes.
23. Faça uma função em C que leia 50 valores do tipo (nome, P1, P2, Nota_trabalhos, Percentual_de_presença), e determine quantos alunos foram aprovados e quantos foram reprovados. Para aprovação é preciso ter média maior ou igual a 6 e 75% de presença. A média é calculada pela expressão: $Media = (P1 + P2) * 70\% + Nota_trabalhos * 30\%$. Use uma função para calcular a média.



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO

24. Faça uma função em C para multiplicar 2 matrizes A ($M \times N$) e B ($N \times K$) e retornar a matriz resultado ($M \times K$).
25. Faça uma função em C que receba um número inteiro e imprima o mês correspondente ao número. Por exemplo, 2 corresponde à "fevereiro". A função deve mostrar uma mensagem de erro caso o número recebido não faça sentido. Gere também um algoritmo que leia um valor e chame o procedimento criado.
26. Faça uma função em C que receba um número inteiro e o imprima na forma extensa. Por exemplo, para 1 a saída desejada é "Um". A função deve ser capaz de gerar o extenso dos números de 0 até 10, inclusive. Caso um número não compatível seja recebido a função deve mostrar uma mensagem de erro. Crie também um algoritmo que leia um valor inteiro e chame a função criada acima para a impressão do número extenso.
27. Faça uma função em C que receba um número arábico inteiro e imprima o corresponde número em romano. Por exemplo, para 5 a saída desejada é "V". A função deve ser capaz de gerar o número romano para os 50 primeiros inteiros. Uma mensagem de erro deve ser mostrada caso um número fora dessa faixa seja recebido. Crie também um algoritmo que leia um valor inteiro e chame o procedimento criado acima para a impressão do número romano.
28. Faça uma função em C que receba um número natural e imprima os três primeiros caracteres do dia da semana correspondente ao número. Por exemplo, 7 corresponde à "SAB". A função deve mostrar uma mensagem de erro caso o número recebido não corresponda à um dia da semana. Gere também um algoritmo que utilize essa função, chamando-a, mas antes lendo um valor para passagem de parâmetro.
29. Faça uma função em C que receba dois números inteiros x e y . Essa função deve verificar se x é divisível por y . No caso positivo, a função deve retornar 1, caso contrário 0. Escreva também um algoritmo para testar tal função.
30. Um número é dito ser regular caso sua decomposição em fatores primos apresenta apenas potências de 2, 3 e 5. Faça uma função em C que verifique se um número é (retorne 1) ou não (retorne 0) regular. Escreva também um algoritmo para testar tal função.
31. Faça uma função em C que determine se um caractere, recebido como parâmetro, é ou não uma letra do alfabeto. A função deve retornar 1 caso positivo e 0 em caso contrário. Escreva também um algoritmo para testar tal função.
32. Um número é dito ser **capicua** quando lido da esquerda para a direita é o mesmo que quando lido da direita para a esquerda. O ano 2002, por exemplo, é **capicua**. Então, elabore uma função em C para verificar se um número possui essa característica. Caso o número seja **capicua**, a função deve retornar 1 e 0 em caso contrário. Escreva também um algoritmo para testar tal função.
33. Faça uma função (não recursiva) em C que calcule e retorne o valor do fatorial de um número natural. A função deve retornar -1 caso não seja possível calcular o valor do fatorial. Escreva também um algoritmo para testar tal função.
34. Faça uma função em C que calcule e retorne o número de arranjos de n elementos p a p . A fórmula do arranjo é a seguinte:

$$A = n! / (n-p)!$$

Caso não seja capaz de calcular tal arranjo a função deve retornar -1. Um algoritmo para testar tal função também deve ser escrito.



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO

35. Faça uma função em C que calcule e retorne o número de combinações de n elementos p a p . A fórmula de combinação é a seguinte:
$$C = n! / (p! * (n-p)!)$$

Caso não seja capaz de calcular tal combinação a função deve retornar -1. Um algoritmo para testar tal função também deve ser escrito.
36. Faça uma função em C que calcule e retorne o **MAIOR** entre dois valores recebidos como parâmetro. Um algoritmo para testar tal função deve ser criado.
37. Faça uma função em C que verifique quantas vezes um número inteiro x é divisível por um número inteiro y . A função deve retornar -1 caso não seja possível calcular. Escreva também um algoritmo para testar tal função.
38. Faça uma função em C que receba dois valores inteiros a e b , retorne o quociente inteiro e o resto da divisão de a por b . A função deve retornar -1 caso não seja possível realizar as operações e 0 caso seja possível. Um algoritmo para utilizar tal função deve ser criado, tratando o retorno da função.
39. Faça uma função em C que receba cinco valores e determine o 2º e o 4º maior valores dentre eles. Construa também um algoritmo para ler tais valores, e imprimir o resultado obtido com a chamada da função.
40. Faça uma função em C que realize a conversão para Radianos (**rad**) a partir de Graus (**grad**), onde **grad** é passado como parâmetro e **rad** é retornado. Sabe-se que 180º (graus) está para 3,14 radianos. Crie também um algoritmo para testar tal função.
41. Faça uma função em C que realize a conversão de Fahrenheit (**F**) para graus Celsius (**C**), onde **F** é passado como parâmetro e **C** é retornado. Sabe-se que os pontos de fusão e ebulição nas escalas Celsius e Fahrenheit são: 0ºC e 100ºC, e 32ºF e 212ºF, respectivamente. Crie também um algoritmo para testar tal função.
42. Faça uma função em C que realize a conversão de Polegadas (**pol**) para Centímetros (**cm**), sendo que **pol** é passado como parâmetro e **cm** é retornado. Sabe-se que 1 polegada está para 2,54 centímetros. Crie também um algoritmo para testar tal função.
43. Faça uma função em C que realize a conversão de pés (**feet**) para metros (**m**), sendo que **feet** é passado como parâmetro e **m** é retornado. Sabe-se que 1 metro está para 3,281 pés. Crie também um programa para testar tal função.
44. Faça uma função recursiva em C chamada FIB(n) que receba o inteiro n e devolva o n-ésimo termo da sequência de Fibonacci, definida recursivamente por:

$$fib(n) = fib(n-1) + fib(n-2) \quad \text{se } n \geq 3$$

$$fib(n) = 1 \quad \text{se } 0 < n < 3$$