



**UNIVERSIDADE FEDERAL DO CEARÁ - *CAMPUS* DE CRATEÚS**  
**FUNDAMENTOS DE PROGRAMAÇÃO 2023.1**  
**PROFA. KAMILA GOMES**

**LISTA DE EXERCÍCIOS 04**

Os problemas a seguir devem ser solucionados utilizando **Funções Não Recursivas**.

1. Faça uma função que recebe um valor inteiro e verifica se o valor é positivo, negativo ou zero. A função deve retornar 1 para valores positivos, -1 para negativos e 0 para o valor neutro.
2. Crie uma função que receba três valores, 'a', 'b' e 'c', que são os coeficientes de uma equação do segundo grau e retorne o valor do delta, que é dado por:  $b^2 - 4ac$ .
3. Dada as dimensões de um terreno:
  - a. Faça uma função para exibir sua área;
  - b. Em seguida, faça uma função que imprima o seu perímetro.
4. Crie uma função em linguagem C que receba 3 números e retorne o maior valor.
5. Faça uma função que retorne a soma de S, dada por um estrutura de repetição. Considere  $\pi=3.14$ . ( $S = -62.806873$ ).

$$S = \sin\left(\frac{5\pi}{7}\right) + 5^{-1} - \sqrt{9} + \sin\left(\frac{7\pi}{9}\right) + 5^{-2} - \sqrt{18} + \dots + \sin\left(\frac{23\pi}{25}\right) + 5^{-10} - \sqrt{90}$$

6. Crie uma função em linguagem C que receba 3 números e retorne o menor valor.
7. Criar uma função que retorna o seguinte: A função recebe 3 valores float e retorna o quadrado do 1º + a soma dos outros dois. A função deve retornar o tipo inteiro.
8. Faça um procedimento para verificar se um ano é bissexto ou não. Utilize a regra: um ano bissexto é divisível por 4, mas não por 100, ou então se é divisível por 400. Exemplo: 1988 é bissexto, pois é divisível por 4 e não é por 100; 2000 é bissexto porque é divisível por 400. O ano deve ser recebido por parâmetro.
9. Faça uma função que valide uma hora no formato de 24h, composta por horas e minutos, no padrão HH:MM. Onde H representa um dígito da hora e M representa um dígito de minuto. A função retornará 1 se a hora é válida e -1 caso contrário.
10. É considerado um número primo, aquele que somente é divisível por 1 e por ele mesmo. Nesse caso, escreva uma função que verifique se um número inteiro é primo ou não.
11. Implemente um programa que leia dois números naturais, n e k, e calcule a combinação de n, k a k (tal qual a fórmula abaixo). Dica: crie uma função que calcule o fatorial e em seguida crie uma função que calcule a combinação.

$$C_{N,K} = \frac{N!}{K!(N-K)!}$$

Os problemas a seguir devem ser solucionados utilizando **Funções Recursivas**.

12. Dado o somatório abaixo, faça uma função recursiva para realizar seu cálculo.
13. Faça uma função que receba um número de 3 dígitos e inverta sua posição. Exemplo: caso a entrada seja 321, a saída será 123.
14. O máximo divisor comum dos inteiros x e y é o maior inteiro que é divisível por x e y. Escreva uma função recursiva em C, que retorna o máximo divisor comum de x e y.
15. Implemente um algoritmo com uma função recursiva para solucionar a função de Ackermann definida para valores inteiros e não negativos m e n da seguinte forma:

$$A(m, n) = \begin{cases} n + 1, m = 0 \\ A(m - 1, 1), m > 0 \text{ e } n = 0 \\ A(m - 1, A(m, n - 1)), m > 0 \text{ e } n > 0 \end{cases}$$