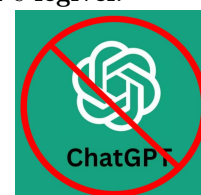




### Instruções para entrega do roteiro:

- Entregue o roteiro apenas no formato *.pdf* com o nome ***Y\_roteiroX.pdf***, onde **X** é o número do roteiro e **Y** é o número da sua matrícula. Não serão aceitos outros formatos.
- Inclua nome e matrícula, e mantenha a resolução dos exercícios **ordenada** e **legível**.
- Códigos completos (com `int main`), compiláveis e executáveis.  
**Para cada um, apresente uma imagem da tela de saída do seu programa.**
- Após a data de entrega, a nota da entrega é 0.
- Em caso de dúvidas, procurem os monitores. Haverá um monitor após as aulas de laboratório para tirar dúvidas sobre a lista.



## Roteiro 1

### Revisão de ponteiros e recursividade

Data máxima de entrega: 08/09/2023  
(Entrega: pelo SIGAA, na sua turma de laboratório.)

## 1 Ponteiros

- 1.1 Implemente uma função que receba como parâmetro um array de números reais (VET) de tamanho N e retorne quantos números negativos existem nesse array. Essa função deve obedecer ao cabeçalho: **`int negativos(float *vet, int N)`**; Teste a função com um programa main.
- 1.2 Implemente um programa que preenche dois vetores diferentes com número aleatórios ( **`rand()`** ) e que contenha 3 funções: (a) Retorna o maior elemento do vetor; (b) Retorna o menor elemento do vetor e (c) Retorna a média dos elementos do vetor. Teste a função com um programa main.
- 1.3 Implemente uma função que receba um vetor de Alunos (Nome, matrícula e Nota) e imprima todas as informações do struct do Aluno que possui a maior nota e todas as informações do aluno que possui a menor nota. Teste a função com um programa main.
- 1.4 Implemente uma função que calcule as raízes de uma equação do segundo grau do tipo  $Ax^2 + Bx + C = 0$ .

Essa função deve obedecer ao seguinte protótipo: **`int raizes(float A, float B, float C, float * X1, float * X2)`**;

Essa função deve ter como retorno o número de raízes reais e distintas da equação. Se existirem raízes reais, seus valores devem ser armazenados nas variáveis apontadas por X1 e X2. Teste a função com um programa main. Lembrando que:

## 2 Recursividade

- 2.1 Escreva uma função recursiva que mostre na tela os números inteiros de 1 a 5 em ordem crescente. Outra para imprimir em ordem decrescente. Teste a função com um programa main.
- 2.2 Escreva uma função recursiva que imprima os elementos de um vetor de letras em ordem decrescente. Teste a função com um programa main.
- 2.3 Escreva uma função recursiva que retorne a soma dos números de um intervalo partindo-se sempre do maior número e terminando no menor. Teste a função com um programa main.
- 2.4 Escreva uma função recursiva que faça a multiplicação de dois números inteiros positivos de forma recursiva.  $N * M = M + M + M + \dots + M$  (N Vezes). Teste a função com um programa main.
- 2.5 Faça o que se pede:
  - a) Faça um algoritmo recursivo para preencher as posições de um vetor de inteiros de tamanho N.
  - b) Faça um algoritmo recursivo para imprimir de forma crescente, do primeiro até o último, (e decrescente, do último até o primeiro) os elementos desse vetor.
  - c) Faça um algoritmo recursivo para encontrar e retornar o maior elemento desse vetor.
- 2.6 Torres de Hanoi: Este problema consiste de três pinos A, B e C, (origem, trabalho e destino), e de n discos de diâmetros diferentes.

O objetivo é empilhar todos os discos no pino C, obedecendo as seguintes restrições:

  - a) apenas um disco pode ser movido de cada vez e
  - b) em nenhum momento um disco pode ser colocado sobre outro de tamanho menor.

Para aprimorar o conhecimento, analise, execute e entenda o seguinte algoritmo recursivo para 1, 2, 3, 4 e 5 discos.

