



Laboratório de Estrutura de Dados

Atividade com Implementação 1 (AI1)

Ponteiros e funções

Semestre: 2022.1

Disponibilizado em: 24/04/2022

Data e local de entrega: 27/04/2022 (SIGAA)

Valor: 5 pontos

Observações (LEIA COM ATENÇÃO)

- A atividade é **individual**.
- Esta atividade faz parte das ATIVIDADES COM IMPLEMENTAÇÕES que compõem 25% do total de pontos distribuídos ao longo do semestre.
- A **linguagem C** deverá ser utilizada para implementar a solução para os exercícios.
- Para **cada** questão deverá ser implementada uma **função principal (main)** para testar as funcionalidades.
- Os exercícios **poderão** ser desenvolvidos em qualquer IDE ou utilizando o prompt de comandos.
- **Somente** o código-fonte em C deverá ser entregue (arquivos com extensão .c). O aluno deverá criar uma pasta com o seu nome (Exemplo: Anderson) onde serão armazenados os códigos-fonte de todas as questões. A pasta deverá ser compactada e enviada no link da tarefa no SIGAA.
- Utilize **boas práticas de programação**, tais como indentação e comentários relevantes.
- *Para aqueles tiverem dúvidas, deem uma olhada na playlist com alguns vídeos curtos sobre ponteiros que deixei como sugestão no SIGAA. O link está disponível [AQUI](#).*

QUESTÕES

1. [1,5 ponto] Implemente uma função que utilize passagem de parâmetros por valor e por referência. A função terá os parâmetros A, B que permitirão receber dois valores inteiros e um terceiro parâmetro C do tipo ponteiro para inteiro. Utilize a variável C para armazenar e "retornar" o resultado do cálculo do produto de A por B. Esta função deverá possuir retorno do tipo `void`. Segue uma sugestão de protótipo para a função:

```
void produto(int a, int b, int *c).
```

Faça um programa principal para testar a funcionalidade. Declare três variáveis inteiras A, B e C. Inicialize a terceira variável com o valor 0. Leia dois valores inteiros e os armazenem nas variáveis A e B. Efetue uma chamada à função e passe como argumentos as três variáveis criadas, sendo que a terceira (C) deverá ser passada por referência para poder ser acessada fora do escopo da função. Imprima os valores de A, B e C antes e após a chamada à função.

3. [1,5 ponto] Crie um programa que preencha um vetor do tipo `int` de N posições, onde N é definido pelo usuário. Os valores "aleatórios" para preenchimento deverão ser gerados aleatoriamente no intervalo [10, 100]. Após o preenchimento, imprima o conteúdo de cada posição desse vetor separados por um espaço.

2. [2 pontos] Implemente uma função para inverter a ordem dos elementos presentes em um vetor. A inversão deve ser feita utilizando o próprio vetor, ou seja, sem utilizar estruturas auxiliares. Após a utilização da função, o primeiro elemento do vetor tem que ter sido trocado com o último, o segundo com o penúltimo e, assim, sucessivamente. Esta função poderá ser definida sem valor de retorno e com dois parâmetros: um para representar o vetor e outro para o seu tamanho. Possível protótipos para a função:

```
void inverter(int vet[], int tam)
```