



## PRÁTICA DE APRENDIZAGEM

**Disciplina:** Paradigmas de Programação

**Tema:** Paradigma Imperativo

**Professores(as):** Carlo Kleber

**Aluno:** \_\_\_\_\_

**Matrícula nº:** \_\_\_\_\_ **Data:** \_\_\_\_/\_\_\_\_/\_\_\_\_

### ORIENTAÇÃO GERAL

- 1) Leia **ATENTAMENTE** as questões.
  - 2) Os códigos desta atividade, quando ofertados, estão na **linguagem de programação C** e focam o **PARADIGMA DE PROGRAMAÇÃO IMPERATIVO**. Se necessário, consulte os seguintes sítios para ter a definição desse paradigma e de importantes conceitos afins, como **PARADIGMA DE PROJETO DE ALGORITMOS**:
    - [https://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o\\_imperativa](https://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o_imperativa)
    - [https://en.wikipedia.org/wiki/Imperative\\_programming](https://en.wikipedia.org/wiki/Imperative_programming)
    - [https://pt.m.wikipedia.org/wiki/Paradigma\\_de\\_programa%C3%A7%C3%A3o](https://pt.m.wikipedia.org/wiki/Paradigma_de_programa%C3%A7%C3%A3o)
    - [https://pt.wikipedia.org/wiki/Projeto\\_de\\_algoritmos](https://pt.wikipedia.org/wiki/Projeto_de_algoritmos)
  - 3) Caso deseje utilizar uma outra linguagem de programação diferente do C, considere o propósito dos códigos oferecidos e converta-os para a **linguagem de sua preferência**.
  - 4) Utilize a ferramenta **Dev-C++** (<http://www.bloodshed.net/>) para compilação/execução dos códigos ou, se desejar, uma **outra de sua preferência**.
  - 5) Caso sinta a necessidade, **consulte também outros sítios** da Internet.
  - 6) A atividade é **individual** e estimulam-se: **autonomia, independência e responsabilidade**.
- Boa sorte! 😊

### QUESTÃO 1

Baixe para sua máquina os arquivos **anexos** < PA-01-1-1-MAX.cpp > e < vetor.txt >

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite duas características que definem esse código como de **paradigma de programação imperativo**? Justifique suas respostas.

### QUESTÃO 2

Baixe para sua máquina os arquivos **anexos** < PA-01-1-4-MAX-MIN-3.cpp > e < vetor.txt >

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite três características que definem esse código como de **paradigma de programação imperativo**? Qual a diferença de propósito entre este programa e aquele da questão anterior? O que foi modificado no código (estruturas de dados, comandos, etc.?) para ter-se o novo propósito? Justifique suas respostas.

### QUESTÃO 3

Baixe para sua máquina os arquivos **anexos** < PA-01-1-5-ORDENA.cpp > e < vetor.txt >

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite três características que definem esse código como de **paradigma de programação imperativo**? Considerando o propósito identificado por você, seria possível realiza-lo de forma mais eficiente (i.e., menor complexidade)? Em caso



positivo, reescreva esse código da forma mais eficiente imaginada por você. Essa nova forma ainda se conforma ao **paradigma de programação imperativo**? Essa nova forma ainda se conforma ao **paradigma de projeto de algoritmo original**? Justifique suas respostas. Se necessário, considere consultar o sítio a seguir:

- [https://pt.wikipedia.org/wiki/Algoritmo\\_de\\_ordena%C3%A7%C3%A3o](https://pt.wikipedia.org/wiki/Algoritmo_de_ordena%C3%A7%C3%A3o)

#### **QUESTÃO 4**

Baixe para sua máquina os arquivos **anexos** < PA-01-1-8-PASS-PARAMETRO.cpp >.

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite três características que definem esse código como de **paradigma de programação imperativo**? Você deve ter percebido que são usados dois modos distintos para realizar a passagem de parâmetros, qual a diferença entre esses dois modos e quais as vantagens e desvantagens de cada um? Esses dois modos se aplicam igualmente aos distintos **paradigmas de projeto de algoritmos**? Justifique suas respostas. Se necessário, considere consultar o sítio a seguir:

- <https://www.ime.usp.br/~elo/IntroducaoComputacao/Funcoes%20passagem%20de%20parametros.htm>

#### **QUESTÃO 5**

Baixe para sua máquina os arquivos **anexos** < PA-01-2-3-2-4-FIBONACCI.cpp >.

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite três características que definem esse código como de **paradigma de programação imperativo**? Você deve ter percebido que são usados dois modos distintos para realizar o mesmo propósito, qual a diferença entre esses dois modos e quais as vantagens e desvantagens de cada um? Esses dois modos se aplicam igualmente aos distintos **paradigmas de projeto de algoritmos**? Justifique suas respostas. Se necessário, considere consultar o sítio a seguir:

- [https://pt.wikipedia.org/wiki/Sequ%C3%Aancia\\_de\\_Fibonacci](https://pt.wikipedia.org/wiki/Sequ%C3%Aancia_de_Fibonacci)

#### **QUESTÃO 6**

Baixe para sua máquina os arquivos **anexos** < PA-01-2-9-MERGESORT.cpp >.

Estude e analise o código. Compile e execute. O que esse programa faz? Qual é a entrada? Qual é o resultado (saída)? Encontrou algum erro? Se encontrou, conserte-o! Qual a complexidade assintótica de pior caso (tempo e espaço)? Cite três características que definem esse código como de **paradigma de programação imperativo**? Qual o paradigma de projeto de algoritmo utilizado? Justifique suas respostas. Se necessário, considere consultar o sítio a seguir:

- [https://pt.wikipedia.org/wiki/Merge\\_sort](https://pt.wikipedia.org/wiki/Merge_sort)

FIM

