

**AULA PRÁTICA 06 – ALGORITMOS DE ORDENAÇÃO**

- Data de entrega: Até 9 de Abril às 23:55.

- Procedimento para a entrega:.

1. Submissão: via **Moodle**.
2. Os nomes dos arquivos e das funções devem ser especificados considerando boas práticas de programação.
3. Funções auxiliares, complementares aquelas definidas, podem ser especificadas e implementadas, se necessário.
4. A solução deve ser devidamente modularizada e separar a especificação da implementação em arquivos *.h* e *.c* sempre que cabível.
5. Os arquivos a serem entregues, incluindo aquele que contém *main()*, devem ser compactados (*.zip*), sendo o arquivo resultante submetido via **Moodle**.
6. Dentre os arquivos submetidos, deve existir um intitulado *compilcao.txt*, contendo os comandos especificados no *prompt/console* para compilar e executar seu programa.
7. Caracteres como acento, cedilha e afins não devem ser utilizados para especificar nomes de arquivos ou comentários no código.

- Bom trabalho!

Questão 01

Implemente o algoritmo *InsertionSort* para resolver o problema a seguir:

Entrada: A entrada será dada por vários casos de teste. A primeira linha da entrada contém o número **T** de casos de teste, seguido por uma linha em branco. Cada caso teste é iniciado por um inteiro, indicando o número **N** de alunos. Cada uma das **N** linhas seguintes é composta por: (1) matrícula do aluno (apenas números), (2) nota na primeira prova e (3) nota no trabalho prático. Existe uma linha em branco entre dois casos de teste.

Saída: A nota final do aluno é composta pela média ponderada das notas da **prova** e do **TP**. A prova tem peso **0,7** e o **TP** tem peso **0,3**. Na saída, seu programa deve retornar a matrícula e a nota final do aluno, ordenados pela melhor nota final.

Exemplo de entrada e saída

Entrada	Saída
2	789 8.6 456 7.65 123 7.0
3	
123 10.0 0.0	
456 7.5 8.0	1 7.0
789 8.0 10.0	2 7.0
2	
1 10.0 0.0	
2 7.0 7.0	

Questão 02

Implemente o algoritmo **MergeSort** para resolver o problema apresentado na questão anterior.

Questão 03

Num arquivo *.pdf* ou *.txt*, responda se é possível que o resultado impresso pelo programa seja diferente entre os algoritmos *BubbleSort*, *InsertionSort*, *SelectionSort* e *MergeSort*? Por quê?