



**INSTITUTO FEDERAL**

Goiano

Campus Morrinhos

---

**Estrutura de Dados II – Algoritmos de Busca e Tabela Hash**

**Professor Msc. Marcel Melo**

---

- 1- Dada a sequência de números: [3 4 9 2 5 8 1], ordene-a a lista de forma crescente segundo os seguintes algoritmos apresentados em sala de aula. Apresente a sequência obtida a cada passo do algoritmo:
  - a. BubbleSort
  - b. InsertionSort
  - c. SelectionSort
  - d. ShellSort
  - e. MergeSort
  - f. QuickSort
  
- 2- Crie um vetor de tamanho 100.000 (cem mil) e o preencha com valores randômicos entre 0 e 100.000. Calcule o tempo que cada algoritmo demandará para ordenar este vetor. Repita essa execução 10 vezes e apresente o resultado de cada execução em uma tabela comparativa.
  
- 3- Faça um programa que leia n valores, informados em tempo de execução pelo usuário, e para cada valor informado, insira-o em uma lista de forma ordenada, utilizando os conceitos do algoritmo de ordenação por inserção. **IMPORTANT!** Não é para criar uma lista desordenada e posteriormente ordená-la. Para cada elemento informado você deve adicioná-lo na posição correta.
  
- 4- Crie uma versão melhorada do algoritmo BubbleSort onde ele para sua execução quando for possível perceber que a lista está ordenada, ou seja, quando não acontece nenhuma troca de elementos em uma iteração do algoritmo. Teste-o conforme o exercício 2 e apresente o tempo de execução desta versão melhorada.
  
- 5- O algoritmo Dual-Pivot Quicksort, proposto por Yaroslavskiy (2009), particiona um array de entrada de dados em três partes, utilizando dois pivôs P1 e P2, onde:  
Parte 1: os valores menores que P1;  
Parte 2 : os valores maiores que P1 e menores que P2;  
Parte 3: os valores maiores que P2.  
Crie esta versão melhorada do algoritmo QuickSort utilizando dois pivots. Teste-o conforme o exercício 2 e apresente o tempo de execução desta versão melhorada.