

	FACULDADES DOCTUM DE CARATINGA	
	CURSO DE CIÊNCIA DA COMPUTAÇÃO	
	DISCIPLINA: Laboratório de Ordenação e Pesquisa	
	PROFESSOR: M.Sc. Elias Gonçalves	
	TURMA: 3º Período	SEMESTRE / ANO: 2º / 2022
	DATA DE ENTREGA: 15/06/2022	

QuickSort

1) (2,50) Altere o código do quickSort para contemplar a estratégia que utiliza a mediana de três elementos para escolha do pivô.

2) (5,00) Com base no quickSort crie um algoritmo chamado quickSearch. Este algoritmo deverá pesquisar e retornar o n-ésimo menor elemento de uma sequência.

Exemplo:

- Dado uma sequência $s = \{9, 7, 50, 2, 1, 8, 12, 30\}$. Suponha que os elementos estejam armazenados nessa ordem em um vetor.
- Seja $n = 3$, então o programa deve retornar o terceiro menor elemento do vetor. Nesse caso o retorno seria o número 7.
- Você não deverá ordenar a sequência s para depois encontrar o n-ésimo elemento.

Para o exemplo acima uma chamada a função quickSearch seria: quickSearch(s, menor, maior, n);

Onde:

- s é o nome do vetor;
- menor é a menor posição do vetor, no caso 0;
- maior é a maior posição do vetor, no caso 7;
- n indica que desejamos o n-ésimo menor elemento do vetor, no caso 3.

Métodos de pesquisa

3) (5,00) Conhecendo as funções de busca sequencial e busca binária e os métodos de ordenação estudados, assuma que um conjunto de dados randômicos representado por um vetor de 100 posições possua chaves que podem se repetir. Para esse caso, ao aplicar os algoritmos de busca você deve retornar, em um outro vetor, todas as posições onde a chave for encontrada.

a) Considere o vetor não ordenado e aplique a pesquisa sequencial simples.

b) Aplique um dos métodos de ordenação no vetor.

c) Considerando o vetor ordenado aplique a pesquisa ordenada.

d) Considerando o vetor ordenado aplique a pesquisa binária.

e) Compare os resultados dos 3 métodos de pesquisa aplicados. Utilize como métrica o número de comparações ao fazer a pesquisa e o tempo de execução de cada método.

4) (2,50) Modifique o algoritmo de ordenação por inserção de forma que ele utilize busca binária para encontrar a posição de inserção de um elemento no vetor destino.