

OCLM - Projeto 2

Ordenar e Buscar

Individual ou dupla
Envio: até 17/11/2018

Buscar informações é parte fundamental de muitas aplicações computacionais. O estado do conjunto de dados (ordenado ou desordenado) pode influenciar muito o desempenho dos algoritmos de busca.

Neste projeto iremos ordenar dados armazenados sequencialmente na memória (em um vetor de dados) e depois verificar se uma informação faz parte ou não dos dados, usando Assembly 8086.

O projeto consistirá de 3 fases:

1. Preenchimento do vetor

O usuário poderá escolher a quantidade de dados (tamanho do vetor) e se o vetor está inicialmente:

- a) Ordenado,
- b) Inversamente Ordenado, ou
- c) Composto por valores (pseudo) aleatórios.

Após o preenchimento e ordenação do vetor, o usuário poderá visualizar o conteúdo do vetor, através de interface simples. Ex.:

Antes da Ordenação:

2 5 1 6 3 4

Após a Ordenação:

1 2 3 4 5 6

2. Ordenação

O sistema deverá então ordenar o vetor (mesmo que já esteja ordenado, para comparação das operações realizadas).

A quantidade de comparações entre elementos do vetor e a quantidade de trocas entre elementos realizadas devem ser informadas ao usuário.

3. Busca

O usuário digitará valores (até optar por sair do programa) e o sistema buscará cada valor desejado. O sistema deverá informar se o valor está presente ou não no vetor. Se estiver presente, deve informar também em qual posição foi encontrado.

Para cada busca deve ser informada a quantidade de comparações entre elementos realizadas pelo algoritmo.

PROJETO BÁSICO [3 pontos]

O vetor será composto por *bytes* (ex.: valores de 0 a 99), ordenado por método $O(n^2)$ (*BubbleSort* – turma de 3ª feira; ou *Select(ion)Sort* – turma de 5ª feira), e os valores procurados por busca sequencial.

PROJETO INTERMEDIÁRIO [5 pontos]

O vetor será composto por *words* (ex. valores de 1 a 30.000), ordenados como no projeto básico, e procurados por busca binária.

PROJETO COMPLETO [8 pontos]

Além das características do Projeto Intermediário, a ordenação será feita por método $O(n \cdot \log_2 n)$, como *Quicksort*.

OPCIONAIS:

APRESENTAÇÃO GRÁFICA SIMPLES [até 2 pontos]

Os vetores serão apresentados na forma gráfica tradicional. Ex:

1	2	3	4	5	6
---	---	---	---	---	---

APRESENTAÇÃO GRÁFICA [até 5 pontos]

As operações de ordenação e busca serão apresentadas graficamente ao serem realizadas. Ex.:

	⇓			⇓	
1	2	3	4	5	6

Comparações: 15

Trocas: 5

Considerações Gerais:

- Será possível estabelecer um tamanho máximo para o vetor, desde que esse tamanho seja maior ou igual a 20.000 posições. Para vetores grandes, a visualização do conteúdo pode ser restrita às primeiras posições.
- Será possível estabelecer previamente o domínio dos valores do vetor (ex. [1 .. 100]) ou pedir que o usuário faça isso.
- Caso a nota deste projeto ultrapasse 10 pontos, os pontos excedentes poderão ser adicionados ao primeiro projeto, até o limite que eleve a média da parte prática a 10 pontos.
- O projeto deverá ser enviado para o monitor da turma até 17/10/2018. De 18/11/2018 até 24/11/2018 valerá 60% da nota.