

Acadêmico(a)			RA:
Curso	Tecnologia em Sistemas para Internet	Período:	
Disciplina	TSI32B - Estrutura, Pesquisa e Ordenação de Dados		
Professor	Eduardo Pena		
Lista primária 2 - Análise e Ordenação			
Orientações gerais:			
1 - Resolva cada exercício em sua respectiva classe, cujo nome deve estar no formato EX"NumExercício"_"TítuloExercício".java.			
2 - A interpretação das questões é parte do processo de avaliação.			
3 - Enviar todos os arquivos (apenas .java) em um arquivo compactado .zip. Verificar prazos no Moodle Institucional.			

- Suponha que recebamos três conjuntos,  $A$ ,  $B$  e  $C$ , armazenados em três vetores de inteiros diferentes. Assumimos que cada um dos três conjuntos não contém valor duplicado individualmente, porém é possível existir números duplicados entre dois ou três conjuntos. Escreva um programa “TresConjuntosDisjuntos” que determine se a intersecção dos três conjuntos  $A$ ,  $B$  e  $C$  é vazia. Formalmente, não existe  $x$  tal que  $x \in A$ ,  $x \in B$ , e  $x \in C$ .

(a) Escreva um algoritmo cujo pior caso seja  $O(N^3)$ .

(b) É possível termos um algoritmo cujo pior caso seja  $O(N^2)$ . Escreva tal algoritmo.

- É nos dado um vetor com  $N$  elementos e nos perguntado se todos os elementos daquele vetor são distintos um dos outros.

(a) Escreva um algoritmo cujo pior caso seja  $O(N^2)$ .

(b) O método `Arrays.sort(vetor)` ordena um vetor em ordem ascendente. Os melhores algoritmos de ordenação (incluídos em `Arrays.sort(vetor)`) estão em  $O(N \log(N))$ . Assuma que o método `Arrays.sort(vetor)` possa ser utilizado e escreva um novo algoritmo/programa para o exercício acima. Qual a complexidade do novo algoritmo? Explique.

- O algoritmo bubble sort ordena um vetor realizando múltiplas passagens pelo mesmo. O algoritmo compara itens adjacentes e troca aqueles que estão fora de ordem. Em cada passagem pela lista, o algoritmo coloca o próximo maior valor na sua posição correta. A ideia geral do bubble sort é que cada elemento se desloca como uma “bolha” para a posição ordenada à qual pertence.

A Figura 3 mostra a primeira iteração do bubble sort em um vetor com 9 elementos. Os elementos sombreados são aqueles sendo comparados para verificar se estão ou não fora de ordem. Se existem  $n$  elementos no vetor, então existem  $n-1$  pares de elementos que precisam ser comparados na primeira passagem. Observe que caso o maior valor no vetor esteja em alguma comparação, ele será continuamente empurrado até o fim da iteração—efeito bolha.

Implemente o algoritmo bubble sort em java. **Claramente, existem códigos prontos disponíveis online. Porém, a ideia desse exercício é que o aluno escreva/teste/valide o algoritmo por conta própria.**

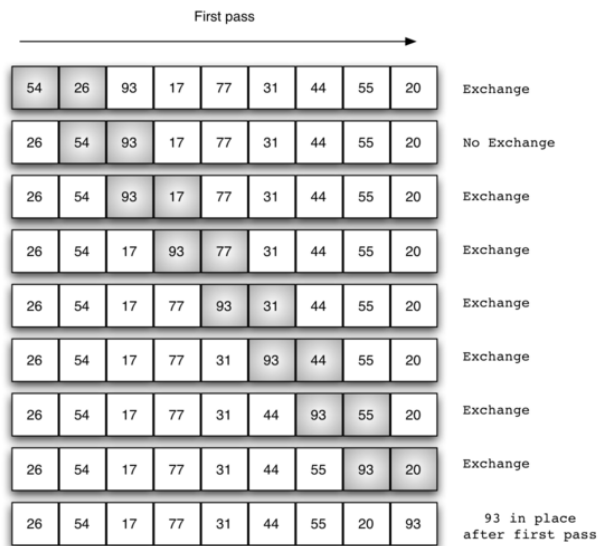


Figura 1: Primeira passagem do bubble sort em um exemplo.