

Ordenação Inteligente

Desafio05

Informações

- Atividade individual.
- Peso no bimestre: **10%**

Instruções

1. Ao término, entregar o projeto BlueJ no Portal COMP, compactado como .zip, e chamar o professor para apresentação e arguição.
2. Permitida consultas a materiais e Internet.
3. Permitido tirar dúvidas pontuais com colegas, atentando às regras de colaboração da disciplina, conforme postulado nos recursos básicos da disciplina.

Boa Trabalho!

1. **Generalização.** Modifique o algoritmo de ordenação de arranjos por inserção para ordenar coleções do tipo `List<Integer>` (listas de inteiros). Consulte a documentação.
2. **Ordenação mista.** Sabe-se que o algoritmo da inserção é o que possui melhor desempenho para ordenações de listas com até 20 elementos. Sabe-se também que o algoritmo de mergesort é assintoticamente menos complexo, de forma que listas com muitos elementos são ordenadas mais rapidamente por ele. Sabendo disso, programe um algoritmo que ordene listas com mergesort sempre que contiverem mais de 20 elementos, e para os outros casos, utilize o método da inserção.
3. **Ordenação inteligente.** Alguns algoritmos de ordenação conseguem ser muito rápidos quando uma lista de elementos está apenas levemente desordenada, mas são muito lentos quando o fator de desordem é alto. Por isso, implemente:
 - (a) Um método que calcule o fator de desordem, dado pela média da quantidade de elementos da lista que são maiores que o elemento seguinte (para o caso da ordenação crescente).
 - (b) Um método de ordenação que, quando a lista tem fator de ordenação menor que 20%, utilize o mergesort, caso contrário utilize o quicksort.
 - (c) Calcule a complexidade assintótica do método que calcula o fator de desordem.