

**Minitrabalho 1 - Data limite: 12/03/2023 (23h59m)**  
**Testes empíricos vs. Análise de Complexidade**

**O grupo deve criar o código pedido e responder às questões seguintes. O relatório deve consistir num *Jupyter notebook* e esse ficheiro deve ser submetido de acordo com as instruções na página de submissão do Moodle.**

1. Crie um segmento de código para gerar aleatoriamente uma coleção de inteiros e devolver numa list.
2. Implemente (pelo menos) os algoritmos de ordenamento mais comuns, como sejam os algoritmos de:
  - i. Bubblesort
  - ii. Selectionsort
  - iii. Insertionsort
  - iv. Shellsort
  - v. Mergesort
  - vi. Quicksort
3. Usando o resultado da alínea 1), crie uma bateria de testes para estes algoritmos e cronometre o tempo de execução de cada uma das funções de ordenamento implementada em 2. Repita cada cronometragem pelo menos 35 vezes, e registe o tempo médio de execução para cada algoritmo. Pode, e deve, gerar situações de variação da organização inicial do input para simular o aparecimento de instâncias mais pesadas (pior dos casos) ou mais simples de resolver.
4. Faça *plots* dos tempos médios de cada função, para cada um dos casos de i) a vi) no exercício 3. Pode usar vários gráficos se fizer sentido ou melhorar as visualizações.
5. Que conclusões pode retirar da análise dos gráficos?  
Nota: Deve estabelecer ligações entre os tempos médios encontrados e as ordens de complexidade conhecidas para os algoritmos, explicando qualquer discrepância que venha a ser encontrada.
6. Se ainda não o fez, gere agora sequências ordenadas de inteiros (use um `range()`, por exemplo) e faça de novo as cronometragens. Há alguma diferença? Indique qual(ais) a(s) diferença(s) e justifique em termos de análise de complexidade de algoritmos.
7. Mostre, usando o Método Indutivo, que o algoritmo da Seleção Linear está correto. Inicie por apresentar um pseudo-código para este algoritmo.

**Esquema geral de avaliação:**

Implementações claras e bem documentadas: até 3 valores; Bateria de testes eficiente e bem documentada: até 4 valores; Resposta correta à questão 5): até 7 valores; Resposta correta à questão 6): até 2 valores; Resposta correta à questão 7): até 4 valores

**Notas finais:**

- Podem discutir as implementações e tirar dúvidas com os docentes de DAA.
- As respostas incorretas ou incompletas descontam de acordo com a questão específica e a incorreção ou incompletude encontrada.
- Incorporação de outros algoritmos (variantes que tentam otimizar alguns dos algoritmos acima descritos) e análise de comparação de desempenho nos testes tendo em atenção o que seria de esperar pela análise assintótica teórica são valorizadas.