

Ordenação - Prática

Marco A L Barbosa

malbarbo.pro.br

- 1) Partindo do arquivo `ordena_encadeamento.py`, implemente o algoritmo de ordenação por inserção para listas encadeadas. Faça uma versão in-loco e outra que devolve uma nova lista. Qual você achou mais simples?
- 2) Partindo do arquivo `ordena.py`, implemente uma versão do algoritmo de ordenação por inserção para arranjos em que a parte ordenada fique no final do arranjo.
- 3) Partindo do arquivo `ordena_encadeamento.py`, implemente o algoritmo de ordenação por seleção para listas encadeadas. Faça uma versão in-loco e outra que devolve uma nova lista. Qual você achou mais simples?
- 4) Partindo do arquivo `ordena.py`, implemente uma versão do algoritmo de ordenação por seleção para arranjos em que a parte ordenada fique no final do arranjo.
- 5) Crie um exemplo de arranjo que mostre que a ordenação por seleção não é estável.
- 6) Explique qual é a relação entre os algoritmos de ordenação por inserção, seleção e heap.
- 7) O arranjo [23, 17, 14, 6, 13, 10, 1, 5, 7, 12] representa um heap máximo?
- 8) Mostre passo a passo a construção de um heap mínimo com a partir do arranjo [5, 1, 7, 4, 9, 1, 3, 4, 8, 6].
- 9) Crie um exemplo de arranjo que mostre que a ordenação por heap não é estável.
- 10) Faça uma implementação iterativa da função `concerta_heap`.
- 11) Projete a função `intercala(lst: list[int], ini: int, meio: int, fim: int)` que intercala os elementos de `lst[ini:meio]` (que está ordenado) com `lst[meio:fim]` (que está ordenado) de maneira que `lst[ini:fim]` fique ordenado.
- 12) Projete uma função que implemente o esquema de particiona de Hoare.
- 13) Explique qual é a relação entre os algoritmos de ordenação por intercalação e particionamento.
- 14) Qual é o resultado do particionado do arranjo [23, 17, 14, 6, 13, 10, 1, 5, 7, 12] usando o algoritmo de Lomuto? E usando o algoritmo não in-loco do material?
- 15) Crie um exemplo de arranjo que mostre que a ordenação por seleção usando o particionamento de Lomuto é instável.
- 16) A ordenação por baldes pode ser usadas para chaves inteiras? Explique.
- 17) Um outro algoritmo de ordenação é a ordenação por flutuação (do inglês, *bubble-sort*). A ideia do algoritmo é percorrer os elementos da sequência diversas vezes e fazer o maior elemento “flutuar” para o final. Isto pode ser feito da seguinte forma: a cada passagem pelos elementos, cada par de elementos adjacentes e comparado e trocando de lugar se o primeiro for maior que o segundo. O algoritmo para quando uma passagem por todos os elementos não gerar nenhuma troca (você consegue ver porque o algoritmo funciona?). Implemente o algoritmo de ordenação por flutuação para arranjos e listas encadeadas. Determine a complexidade de tempo, se ele é in-loco e se é estável.