

Visão geral de algoritmos de ordenação

Questão 1. (FKM) Dado uma matriz de números, retangular. Ordene cada linha da matriz. Em seguida, ordene cada coluna da matriz. Mostre que as linhas da matriz continuam ordenadas.

Questão 2. (Manber) Em alguns casos, a entrada de um algoritmo de ordenação já está quase ordenada, o que significa que o número de elementos fora de ordem é pequeno. Descreva como os algoritmos de ordenação que você conhece se comportam com sequências quase ordenadas. Que algoritmo você usaria? (Você é encorajado a projetar o seu próprio algoritmo)

Fila de prioridade e Heapsort

Questão 3. (Manber) A entrada é um *heap* de tamanho n (em que o maior elemento está no topo), dado como um vetor, e um número real x . Projete um algoritmo para determinar se o k -ésimo maior elemento no heap é menor ou igual a x . No pior caso, seu algoritmo deve executar em tempo $O(k)$, independente do tamanho do heap. Você pode usar espaço de tamanho $O(k)$. (Note que você não tem que encontrar o k -ésimo maior elemento; você só precisa determinar sua relação com x .)

Questão 4. (Manber) A entrada são d sequências de elementos tais que cada sequência já está ordenada e há um total de n elementos. Projete um algoritmo $O(n \log d)$ para juntar todas as sequências em uma única sequência ordenada.

Questão 5. (CLRS) Exercícios: 6.1-1, 6.1-2, 6.1-3, 6.1-4, 6.1-5, 6.1-6, 6.2-1, 6.2-2, 6.2-6, 6.3-2, 6.4-1, 6.4-3,

¹Esta lista deve ser feita logo após as aulas do conteúdo correspondente e serve para fixar o conteúdo, confirmar ou identificar as dúvidas. Anote suas dúvidas e procure atendimento! Os exercícios são referências ou transcrições de exercícios dos livros-textos (CLRS/Manber), ou foram gentilmente cedidos por outros professores, particularmente por Flávio Keidi Miyazawa (FKM), Cid Carvalho de Souza e Orlando Lee (CID/OL).