

MAC 5711 - Análise de Algoritmos
Departamento de Ciência da Computação
Primeiro semestre de 2025

Lista 5

1. **(CLRS 8.3-2)** Quais dos seguintes algoritmos de ordenação são instáveis: insertionsort, mergesort, heapsort, e quicksort. Descreva uma maneira simples de deixar qualquer algoritmo de ordenação estável. Quanto tempo e/ou espaço adicional a sua estratégia usa?
2. **(CLRS 8.2-1)** Simule a execução do COUNTINGSORT usando como entrada o vetor

$$A[1..11] = \langle 6, 0, 2, 0, 1, 3, 4, 6, 1, 3, 2 \rangle.$$

3. **(CLRS 8.2-2)** Mostre que o COUNTINGSORT é estável.
4. **(CLRS 8.2-3)** Suponha que o **para** da linha 7 do COUNTINGSORT é substituído por

para $j \leftarrow 1$ até n faça

Mostre que o COUNTINGSORT ainda funciona. O algoritmo resultante continua estável?

5. **(CLRS 8.2-4)** Escreva um algoritmo que, dados n inteiros no intervalo de 1 a k , preprocesse sua entrada e então responda em $O(1)$ qualquer consulta sobre quantos dos n inteiros dados caem em um intervalo $[a..b]$. O preprocessamento efetuado pelo seu algoritmo deve consumir tempo $O(n + k)$.
6. **(CLRS 8.3-4)** Mostre como ordenar n inteiros no intervalo de 0 até $n^3 - 1$ em tempo $O(n)$.
7. **(CLRS 8.4-1)** Simule a execução do BUCKETSORT com o vetor

$$A[1..10] = \langle 0.79, 0.13, 0.16, 0.64, 0.39, 0.20, 0.89, 0.53, 0.71, 0.42 \rangle.$$

8. **(CLRS 8.4-2)** Qual é o consumo de tempo de pior caso para o BUCKETSORT? Que simples ajuste do algoritmo melhora o seu pior caso para $O(n \lg n)$ e mantém o seu consumo esperado de tempo linear.