

Construção e Análise de Algoritmos

Exercícios adicionais (divisão e conquista)

1. Contando inversões

Considere um vetor de inteiros $V[1..n]$.

Nós dizemos que as posições i e j de V formam uma inversão se

$$i < j \quad \text{e} \quad V[i] > V[j]$$

Isto é, as posições i e j formam uma inversão se os números armazenados nessas posições estão invertidos, com relação à sua ordem relativa na versão ordenada do vetor.

O problema consiste em determinar o número total de inversões no vetor V .

A solução ingênua para esse problema consiste em comparar todos os $O(n^2)$ pares de elementos para descobrir quais deles formam uma inversão.

Apresente um algoritmo de divisão e conquista que resolve esse problema em tempo $O(n \log n)$.

2. Contando inversões significativas

Suponha que alguém modifica a definição de inversão para o seguinte:

$$i < j \quad \text{e} \quad V[i] > 2 \cdot V[j]$$

Quer dizer, agora só contam as inversões onde o número da esquerda é muito maior (i.e., ao menos o dobro) do que o número da direita.

Para não haver confusão, nós vamos chamar essa nova definição de uma *inversão significativa*.

Apresente um algoritmo de divisão e conquista que determina o número de inversões significativas em um vetor de inteiros $V[1..n]$ em tempo $O(n \log n)$.

Você consegue resolver esse problema de maneira mais eficiente quando o vetor já se encontra ordenado?

3. Par de pontos mais próximos

Suponha que você tem uma coleção de n pontos no plano

$$C = \{ (x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_n, y_n) \}$$

O problema consiste em identificar o par de pontos cuja distância entre eles é a menor possível.

Lembre que a distância entre dois pontos é calculada da seguinte maneira

$$D_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Dica: Aplique a técnica da divisão e conquista ordenando a coleção de pontos e quebrando o problema em duas partes.