

CCM0128 - Computação II - 2012

Lista 1 - Ordenação

E-mail do monitor: *ademar.mlf@gmail.com*

1. O Selectionsort é um algoritmo de ordenação onde primeiro se deve encontrar o índice do elemento máximo em uma lista (no caso, um vetor) e depois trocar este elemento com o elemento final da lista, o algoritmo é aplicado repetidamente (por meio de recursão ou loop) sempre à sublista de tamanho menor em uma unidade (excluindo o elemento final realocado na vez anterior) até termos uma lista de tamanho 1. Escreva uma função `int max(int V[], int begin, int end)` que devolve o índice do elemento máximo de V entre os valores de índice *begin* e *end* INCLUSIVE. Utilizando a função `max()` que você criou, escreva o algoritmo do Selectionsort como explicitado. Quantas chamadas a função `max()` seu algoritmo realiza? Convença-se que seu algoritmo é $O(n^2)$.
2. Simule (desenhando) os passos que cada um dos algoritmos abaixo realiza para ordenar o seguinte vetor $[1, 3, 2, 5, 4, 4, 2, 3]$. Indique quais algoritmos são estáveis (um algoritmo de ordenação estável é aquele em que a ordem com que os elementos repetidos aparecem no vetor ordenado é preservada). Dê um exemplo de aplicação real de ordenação onde estabilidade seja desejável.
 - Selectionsort
 - Mergesort
 - Quicksort
3. Argumente, da melhor forma que conseguir, que o método de intercalação do mergesort é linear, ou seja, quando aplicada a uma instância de tamanho n o algoritmo é $O(n)$.
4. Proponha uma modificação universal que melhore a performance de melhor caso para os algoritmos de ordenação que você viu até agora sem piorar a complexidade para os demais casos. Faça uma análise do custo-benefício que sua modificação traz sobre os algoritmos originais.
5. Proponha uma modificação que previna um usuário malicioso de piorar a complexidade do tempo esperado do quicksort em uma simulação em que os vetores são escolhidos por este usuário.
6. Expresse a relação entre os pares de funções $f(n)$ e $g(n)$ abaixo usando a notação- O ¹.
 - $f(n) = 2n$ e $g(n) = n/2$
 - $f(n) = n^2$ e $g(n) = n^3$
 - $f(n) = n \lg(n)$ e $g(n) = n + \lg(n)$

¹Exemplo: $f(n) = \sqrt{n}$ e $g(n) = n$: resposta $f = O(g)$ mas $g \neq O(f)$