

Algoritmos e Estruturas de Dados II

Prof. Marco Aurélio

LISTA 2 – ENTREGA: ATÉ 11/04 ÀS 19H NA SECRETARIA DO DCT

Observação: você deve apresentar seus algoritmos utilizando o pseudocódigo visto em sala como linguagem.

1. Prove por indução:

(a) $1 + 3 + 5 + \dots + 2n - 1 = n^2$

(b) $1 + 3 + 6 + \dots + \frac{n(n+1)}{2} = \frac{n(n+1)(n+2)}{6}$

2. *Counting-sort* – Dada uma seqüência x a ser ordenada, crie uma outra seqüência c tal que c_j = número de elementos menores que x_j , $1 \leq j \leq n$. Escreva um algoritmo de ordenação baseado nessa idéia. Qual o custo deste algoritmo? Resolva o caso em que $\exists i, j$ tais que $x_i = x_j$.
3. *Odd-even-sort* – Passe várias vezes pela seqüência x . Na primeira vez compare x_i com x_{i+1} , para todo i ímpar; na segunda vez faça isso para todo i par e assim sucessivamente. Descreva este algoritmo de ordenação e calcule sua complexidade. Qual é a condição de parada para este algoritmo?
4. Entre todos os algoritmos vistos, incluindo os dos exercícios, reescreva-os de tal modo que o resultado seja uma seqüência não-crescente.
5. Entre todos os algoritmos vistos, incluindo os dos exercícios, faça uma tabela contendo as seguintes informações: tempo de execução no pior caso; tempo de execução no melhor caso; se ele é estável ou não; qual o número de registros movimentados no pior caso; qual o número de registros movimentados no melhor caso; e faça um ranking de todos eles (qual é, na sua opinião, o melhor, o segundo melhor, assim por diante).
6. O algoritmo de ordenação por inserção pode ser expresso recursivamente da seguinte forma: para ordenar $A[1 \dots n]$, recursivamente ordene $A[1 \dots n-1]$ e então insira $A[n]$ na seqüência já ordenada $A[1 \dots n-1]$.
 - (a) Escreva o algoritmo recursivo de ordenação por inserção;
 - (b) Calcule o seu custo, no pior caso.