MAC–IME–USP

Carlos Eduardo Ferreira
Sala 108C Tel.: 3091 6079
e-mail cef@ime.usp.br

MAC338 - Análise de Algoritmos

Segundo semestre de 2022

Lista 2

Todos os exercícios são importantes para o aprendizado da disciplina

Vocês devem entregar os exercícios 1b, 1e e 3 até **17 de setembro**

1. Resolva as recorrências abaixo.

   (a) $T(n) = 2T(\lfloor n/2 \rfloor) + \Theta(n^2)$

   (b) $T(n) = 8T(\lfloor n/2 \rfloor) + \Theta(n^2)$

   (c) $T(n) = 2T(\lfloor n/2 \rfloor) + \Theta(n^3)$

   (d) $T(n) = 7T(\lfloor n/3 \rfloor) + \Theta(n^2)$

   (e) $T(n) = T(\lfloor 9n/10 \rfloor) + \Theta(n)$

2. Escreva um algoritmo que ordena uma lista de n itens dividindo-a em três sublistas de aproximadamente $n/3$ itens, ordenando cada sublista recursivamente e intercalando as três sublistas ordenadas. Analise seu algoritmo concluindo qual é o seu consumo de tempo.

3. Seja $X[1\,..\,n]$ um vetor de inteiros e i e j dois índices distintos de X, ou seja, i e j são inteiros entre 1 e n. Dizemos que o par (i, j) é uma *inversão* de X se $i < j$ e $X[i] > X[j]$. Escreva um algoritmo $\mathrm{O}(n \lg n)$ que devolva o número de inversões em um vetor X, onde n é o número de elementos em X.

4. Descreva um algoritmo que, dados inteiros n e k, juntamente com k listas ordenadas que em conjunto tenham n registros, produza uma única lista ordenada contendo todos os registros dessas listas (isto é, faça uma *intercalação*). O seu algoritmo deve ter complexidade $\mathrm{O}(n \lg k)$. Note que isto se transforma em $\mathrm{O}(n \lg n)$ no caso de n listas de 1 elemento, e em $\mathrm{O}(n)$ se só houver duas listas (no total com n elementos).

5. Considere a seqüência de vetores $A_k[1\,..\,2^k], A_{k-1}[1\,..\,2^{k-1}], \ldots, A_1[1\,..\,2^1]$, e $A_0[1\,..\,2^0]$. Suponha que cada um dos vetores é crescente. Queremos reunir, por meio de sucessivas operações de intercalação (= *merge*), o conteúdo dos vetores $A_0, \ldots, A_k$ em um único vetor crescente $B[1\,..\,n]$, onde $n = 2^{k+1} - 1$. Escreva um algoritmo que faça isso em $\mathrm{O}(n)$ unidades de tempo. Use como subrotina o Intercale visto em aula.

6. Considere a recorrência abaixo

$$T(1) = 1; T(n) = \max_{0 \leq k \leq n-1} \{T(k) + T(n-k-1)\} + n, n \geq 2.$$

Prove que $T(n) \geq \frac{n^2}{2}, n \geq 1$.

7. Considere a recorrência

$$T(1) = 1; T(n) = T(\lceil \frac{n}{10} \rceil) + T(\lfloor \frac{9n}{10} \rfloor) + n, n \geq 2.$$

Prove que $T(n)$ é $\mathrm{O}(n \log n)$.