Tópicos Avançados em Algoritmos - Folha de exercícios (revisão para o teste II)

(i) Um ou mais destes exercícios (ou alíneas) serão incluídos na prova. (ii) Este conjunto de exercícios é bastante mais extenso que o teste. (iii) Será colocada uma resolução na página da disciplina.

1. Verdadeiro ou falso?
   Notas. Elemento de ordem i de um vector $v[1 \cdots n]$: aquele que ficaria na posição i se o vector estivesse ordenado. Mediana: elemento de ordem $\lfloor i/2 \rfloor$. Os tempos considerados são tempos no pior caso; $n = |x|$.

   (a) No algoritmo aleatorizado do quick-sort, quaisquer que sejam os dados x (vector a ordenar com n valores), o tempo médio é de ordem $O(n \log n)$.

   (b) Qualquer linguagem na classe BPP é recursiva.

   (c) O menor elemento de um vector já ordenado pode ser determinado em tempo $O(1)$

   (d) O menor elemento de um vector (não necessariamente ordenado) pode ser determinado em tempo $O(1)$

   (e) Para qualquer i, o elemento de ordem i de um vector (não necessariamente ordenado) pode ser determinado em tempo $O(\log n)$

   (f) A mediana de um vector (não necessariamente ordenado) não pode ser determinada em tempo $O(n)$

2. Suponha que uma linguagem L pertence à classe RP, existindo um algoritmo aleatorizado polinomial A tal que $x \in L \Rightarrow$ [prob $\{A(x) = 1\} \geq 1/2$] e $x \notin L \Rightarrow$ [prob $A(x) = 1 = 0$]. Mostre como transformar este algoritmo num algoritmo B que é análogo a A excepto no facto de a probabilidade de errar ser $\leq 10^{-6}$. Justifique.

3. Na demonstração (por indução) de que o número de comparações $t(n)$ efectuadas pelo algoritmo clássico "quicksort" satisfaz $t(n) \leq cn \ln n$ para determinado $c > 0$ a definir, usa-se o seguinte argumento indutivo

$$
\begin{aligned}
t(n) &= (n-1) + \frac{1}{n}\sum_{i=0}^{n-1}[t(i) + t(n-1-i)] && (1)\\
&= (n-1) + \frac{2}{n}\sum_{i=1}^{n-1} t(i) && (2)\\
&\leq (n-1) + \frac{2c}{n}\sum_{i=1}^{n-1} i \ln i && (3)\\
&\leq (n-1) + \frac{2c}{n}\int_1^n x \ln x \, dx && (4)\\
&\leq (n-1) + \frac{2c}{n}\left(\frac{1}{2}n^2 \ln n - \frac{n^2}{4} + \frac{1}{4}\right) && (5)\\
&\leq cn \ln n \quad \text{para} \quad c \geq 2 && (6)\\
&\in O(n \ln n) && (7)
\end{aligned}
$$

   Justifique sumariamente cada passagem deste argumento. A sua justificação deve ter estar dividida do seguinte modo: (1):... / de (1) para (2):... / de (2) para (3):... / de (3) para (4):... / de (4) para (5):... / de (5) para (6):... / de (6) para (7):...

4. Pretende-se ordenar um vector com n inteiros, eventualmente repetidos, compreendidos entre 1 e u, sendo $u \leq 1000$. Descreva numa linguagem de programação informal (mas de forma exacta) um algoritmo para esse fim com eficiência $O(n + u)$.

5. Mostre que a ordenação efectuada em cada etapa (excepto na última) do "radix sort" tem que ser estável.

6. Nem sempre é fácil mostrar que uma rede de comparadores é uma rede de ordenação. Por exemplo, para a rede **A** em baixo, tal prova não é trivial.

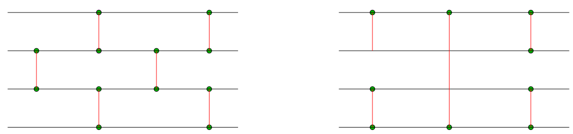
   (a) Indique um resultado que permite poupar bastante tempo na demonstração referida.

   (b) Suponha que as entradas na rede **A** são, de cima para baixo, 4, 3, 2, e 1. Indique os valores nos diversos pontos da rede.

   (c) Mostre que a rede **B** não é de ordenação.

   (d) Determine a profundidade da rede **A**.

   (e) Determine a profundidade da rede **B**.



A **B**

7. Quais das seguintes sequências são bitónicas: ε, `000`, `010`, `000111000`, `0101`?

8. Como sabe, a rede "ordenador bitónico" BS(n) que foi estudada é constituída, para n da forma $n = 2^p$ com $p \geq 1$, por (i) uma rede HC(n) seguida de (ii) duas redes BS($n/2$). Sabendo que o

   – o número de componentes de uma rede HC(n) é $n/2$

   – em qualquer saída a profundidade de uma rede HC(n) é 1

   determine

   (a) a profundidade $d(n)$ de BS(n)

   (b) a número de componentes $c(n)$ de BS(n)

9. Suponha que para um determinado n fixo temos uma rede de ordenação com c comparadores. Mostre que existe um um programa sequencial, sem ciclos nem chamadas a funções, que ordena qualquer sequência de n valores efectuando exactamente c comparações. Todas as instruções do programa devem ser da forma

$$\texttt{if x[i]} < \texttt{x[j]}: \ \ \texttt{x[i]} \leftrightarrow \texttt{x[j]}$$

   Exemplifique para a rede **A**.

10. "Para $n \geq 4$ toda a rede de ordenação com n linhas tem a seguinte propriedade: qualquer linha é extremidade de pelo menos 2 ordenadores".
    Demonstre a afirmação anterior ou, se ela for falsa, apresente um contra-exemplo.

11. "Se as entradas de um comparador são x_1 e x_2 e as saídas y_1 e y_2 e se f for uma função que satisfaz

$$z \leq z' \Rightarrow f(z) \leq f(z')$$

    então, se as entradas forem $f(x_1)$ e $f(x_2)$, as saídas são $f(y_1)$ e $f(y_2)$".
    Demonstre a afirmação anterior ou, se ela for falsa, apresente um contra-exemplo.