MAC 5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2002

Lista 4

1. Exercícios 7.3-1 (6.3-1), 7.4-1 (6.4-1), 7.5-5 (6.5-7) e 7.5-6 (6.5-8) do CLR (CLRS).

2. **(Exercício 6.5-3 do CLRS)** Escreva em pseudocódigo cada um dos seguintes procedimentos, que implementam uma fila de prioridade com um heap de mínimo: `heap-minimo`, `heap-extrai-minimo`, `heap-diminui-chave` e `heap-insere`.

3. **(Exercício 6.4-4 do CLRS)** Mostre que a complexidade de pior caso do heapsort é $\Omega(n \log n)$.

4. **(Exercício 9.1-1/8.1-1 do CLR/CLRS)** Qual é a menor profundidade possível de uma folha em uma árvore de decisão para um algoritmo de ordenação?

5. Exercícios 9.1-2 (8.1-2) e 9.1-3 (8.1-3) do CLR (CLRS).

6. **(Exercício 9.1-4 do CLR)** Professor Ludovico afirma que o limite inferior de $\Omega(n \log n)$ para ordenação de n números não se aplica ao seu ambiente de programação. No seu ambiente, o fluxo de controle do programa pode dividir-se em três depois de uma única comparação $a_i : a_j$, dependendo de qual dos três seguintes casos ocorre: $a_i < a_j, a_i = a_j$ ou $a_i > a_j$. Mostre que o professor está errado, provando que o número de "super" comparações requeridas para ordenar n elementos ainda é $\Omega(n \log n)$.

7. **(Exercício 9.2-5 do CLR)** Descreva um algoritmo que, dados n inteiros no intervalo 1 até k, preprocesse sua entrada e então responda em $O(1)$ qualquer consulta sobre quantos dos n inteiros dados caem em um intervalo $[a..b]$. O preprocessamento efetuado pelo seu algoritmo deve executar em tempo $O(n+k)$.

8. Seja k um inteiro positivo fixo. Mostre como ordenar n inteiros no intervalo de 1 até n^k em tempo $O(n)$.

Entregue no dia 23/9 os exercícios 3, 7.5-5 (6.5-7), 7.5-6 (6.5-8), 9.1-3 (8.1-3) e 7.