MO417 – Complexidade de Algoritmos
Segundo Semestre de 2008
Quinta Lista de Exercícios

1 Estatísticas de Ordem

1. Mostre como QUICKSORT pode ser modificado de modo que tenha complexidade de tempo $O(n \lg n)$ no pior caso.

2. Suponha que você possui um algoritmo linear do tipo "caixa-preta" que resolve o problema de encontrar a mediana. Descreva um algoritmo linear que resolve o problema da seleção (encontrar o i-ésimo menor) para todo i.

3. O k-ésimo **quantil** de um vetor de n inteiros são os $k-1$ elementos que dividiriam o vetor, se este estivesse ordenado, em k partes de tamanho quase iguais (diferindo de no máximo 1). Por exemplo, a mediana é o segundo *quantil*. Descreva um algoritmo de complexidade $O(n \lg k)$ que determina o k-ésimo *quantil* de um vetor de n inteiros.

4. Sejam $X[1\,..\,n]$ e $Y[1\,..\,n]$ dois vetores **ordenados**. Descreva um algoritmo de complexidade $O(\lg n)$ para determinar a mediana dos $2n$ elementos de X e Y. Escreva um pseudo-código. Note que a mediana de $2k$ elementos é o elemento x que é maior ou igual a $k-1$ elementos e menor ou igual a k elementos.

5. Dado um conjunto de n números, queremos listar os i maiores elementos em ordem crescente usando um algoritmo baseado em comparações. Considere os algoritmos baseados nas seguintes idéias:

   (a) Ordene os números e liste os i maiores.

   (b) Construa uma fila de prioridade com os n números e chame EXTRACT-MAX i vezes.

   (c) Use um algoritmo para encontrar o i-ésimo maior número e particione o conjunto em torno dele. Então ordene e imprima os i maiores elementos.

   Analise a complexidade desses algoritmos em função de n e i. Determine quando um algoritmo é melhor que os outros, considerando que valores de i (em função de n).

2 Projeto de Algoritmos por Indução

6. Seja $A[1\,..\,n]$ um vetor de números reais. Uma subseqüência é um subvetor $A[i\,..\,j]$. Projete um algoritmo de complexidade de tempo $O(n)$ que determina uma subseqüência tal que o produto de todos seus elementos seja máximo (considerando todos as possíveis subseqüências). O produto de um segmento vazio é 1 por definição.

7. Suponha que são dados dois vetores $A[1\,..\,n]$ e $B[1\,..\,m]$ de inteiros. Dado um inteiro x, queremos saber se existem um elemento em A e um elemento em B tais que a soma deles é exatamente x. Seu algoritmo deve ter complexidade $O((n+m)\lg(n+m))$.

8. Seja M uma matriz $n \times m$ de números reais tal que

   (a) cada linha de M está ordenada em ordem crescente (da esquerda para a direita) e

   (b) cada coluna de M está ordenada em ordem crescente (de cima para baixo).

   Descreva um algoritmo (baseado em comparações) que recebe M e um inteiro x e determina se x está em M (ou seja, se existem índices i, j tais que $M[i, j] = x$). Seu algoritmo deve ter complexidade $O(n + m)$.

9. Considere um vetor $A[1\,..\,n]$ de números inteiros com a seguinte propriedade: para todo i, $1 \leq i \leq n - 1$, vale que $|A[i] - A[i + 1]| \leq 1$. Suponha que seja dado um elemento x tal que $A[1] \leq x \leq A[n]$. Projete um algoritmo eficiente para encontrar um índice j tal que $A[j] = x$. É verdade que sempre existe tal j? Qual é a complexidade do seu algoritmo?

10. Projete um algoritmo para determinar se dois conjuntos X e Y são disjuntos (não tem elementos em comum). Sendo $|X| = n$ e $|Y| = m$, descreva a complexidade de seu algoritmo em função de n e m.