Exame Preliminar para o Doutorado

IME–USP, Março, 2000

Prova de Análise de Algoritmos

Instruções:

(*i*) O candidato pode resolver todas as questões.

(*ii*) Esta prova contém três questões de um ponto, quatro de dois pontos e uma de três pontos.

(*iii*) A banca considerará questões cujos valores somem até 10 pontos de modo que a soma total das notas obtidas seja máxima.

(*iv*) Mencione os teoremas e propriedades usados para justificar suas afirmações.

Questão 1 [1 ponto]
Defina árvores rubro-negras. Suponha que a raíz de uma árvore rubro-negra é rubra. Mudando a cor da raíz desta árvore, ela permanece uma árvore rubro-negra? Justifique.

Questão 2 [1 ponto]
Se $T(0) = T(1) = 1$, cada uma das seguintes recorrências define uma função T nos inteiros não negativos.

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

(b) $T(n) = 2T(n-2) + 1$

(c) $T(n) = T(n-1) + n^2$

Qual delas não pode ser limitada por uma função polinomial? Justifique a sua resposta.

Questão 3 [1 ponto]
O Sr. B. A. Vaca anunciou que desenvolveu uma nova estrutura de dados para filas de prioridades que permite realizar cada operação de *Inserção, Máximo* e *Extrai-Máximo* em tempo de pior caso $O(1)$, usando apenas comparações de chaves e movimentações. Mostre que ele se enganou.
Sugestão: Lembre do limite inferior $\Omega(n \log n)$ para complexidade da ordenação.

Questão 4 [2 pontos]
O banco de dados de uma empresa consiste de 10.000 nomes ordenados; 40% são classificados como bons clientes e eles são responsáveis por 60% dos acessos ao banco de dados. Estão em consideração duas opções para representar o banco de dados:

1. Colocar todos os nomes num único vetor e usar busca binária.
2. Colocar os bons clientes num vetor e os restantes num outro vetor. Inicialmente seria feita uma busca binária no primeiro vetor. Somente se o nome procurado não for encontrado seria feita uma busca binária no segundo vetor.

Demonstre qual das opções apresenta o melhor tempo médio de consulta. Justifique a sua resposta.

Alguns números que podem ser úteis: $\lg 10000 = 13,29$, $\lg 6000 = 12,55$, $\lg 4000 = 11,97$.

Questão 5 [2 pontos]
Dê um algoritmo que, dado um vetor de n registros, reorganiza os registros de forma a formarem um "heap", usando, no máximo, $O(n)$ comparações. Mostre que a complexidade de pior caso é realmente esta.

Questão 6 [2 pontos]
Construa 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 $O(n \lg k)$. Note que isto se transforma em $O(n \lg n)$ no caso de n listas de 1 elemento, e em $O(n)$ se só houver uma lista (de n elementos).

Questão 7 [2 pontos]
Considere o seguinte algoritmo.

```
#define n <algum valor positivo>

main() {
  int a = 0, b = n, soma = 0, p;

  scanf(p);
  while (a < n) and (b > 0) {
    if (p > 0) {
      a = a + 1;
      soma = soma + 1;
    }
    else {
      a = a - 1;
      b = b - 1;
      soma = soma + 1;
    }
    scanf(p);
  }
}
```

Calcule o valor máximo que pode estar armazenado em `soma` (em função de `n`) no fim da execução do algoritmo. Mostre uma seqüência de valores de `p` que faz `soma` atingir o valor máximo encontrado. Justifique as suas respostas.

Questão 8 [3 pontos]
Descreva um algoritmo eficiente que, dado um conjunto $\{x_1, x_2, \ldots, x_n\}$ de pontos na reta real, determine o menor conjunto de intervalos fechados de comprimento um que contém todos os pontos dados. Justifique informalmente o seu algoritmo e analise a sua complexidade.

BOA SORTE!