MAC 5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2022

Lista 3

1. Qual é o consumo de espaço do QUICKSORT no pior caso?

2. Quando um algoritmo recursivo tem como último comando executado, em algum de seus casos, uma chamada recursiva, tal chamada é denominada *recursão de cauda* (*tail recursion*). Um exemplo de recursão de cauda acontece no QUICKSORT.

   Toda recursão de cauda pode ser substituída por uma repetição. No caso do QUICKSORT, obtemos o seguinte:

   QUICKSORT (A, p, r)
   enquanto $p < r$
       $q \leftarrow$ PARTICIONE (A, p, r)
       QUICKSORT $(A, p, q - 1)$
       $p \leftarrow q + 1$

   Mostre como essa idéia pode ser usada (de uma maneira mais esperta) para melhorar significativamente o consumo de espaço no pior caso do QUICKSORT.

3. Considere o seguinte algoritmo que determina o segundo maior elemento de um vetor $v[1 \,..\, n]$ com $n \geq 2$ números positivos distintos.

   Algoritmo Máximo (v, n)
   1. $maior \leftarrow 0$
   2. *segundo_maior* $\leftarrow 0$
   3. **para** $i \leftarrow 1$ **até** n **faça**
   4.   **se** $v[i] > maior$
   5.   **então** *segundo_maior* $\leftarrow maior$
   6. $maior \leftarrow v[i]$
   7.   **senão se** $v[i] >$ *segundo_maior*
   8.   **então** *segundo_maior* $\leftarrow v[i]$
   9. **devolva** *segundo_maior*

   Suponha que v é uma permutação de 1 a n escolhida ao acaso dentre todas as permutações de 1 a n, de acordo com a distribuição uniforme de probabilidade. Seja X o número de vezes que a variável *segundo_maior* é alterada (ou seja, o número de execuções das linhas 5 e 8 do algoritmo) numa chamada de Máximo(v, n). Note que X é uma variável aleatória. Calcule o valor esperado de X.

4. Considere o seguinte algoritmo que calcula o maior e o menor elemento de um vetor $v[1 \,..\, n]$ com elementos distintos.

   Algoritmo MaiorMenor (v, n)
   1. *maior* $\leftarrow v[1]$
   2. *menor* $\leftarrow v[1]$
   3. **para** $i \leftarrow 2$ **até** n **faça**
   4.   **se** $v[i] >$ *maior*
   5.   **então** *maior* $\leftarrow v[i]$
   6.   **senão se** $v[i] <$ *menor*
   7.   **então** *menor* $\leftarrow v[i]$
   8. **devolva** *maior*, *menor*

   Suponha que a entrada do algoritmo é uma permutação de 1 a n escolhida uniformemente dentre todas as permutações de 1 a n.

   Qual é o número esperado de comparações executadas na linha 6 do algoritmo? Qual é o número esperado de atribuições efetuadas na linha 7 do algoritmo?

5. **(CLRS 8.4-3)** Seja X uma variável aleatória que é igual ao número de caras em duas jogadas de uma moeda justa. Quanto vale $E[X^2]$? Quanto vale $E[X]^2$?

6. Escreva uma função que recebe um vetor com n letras A's e B's e, por meio de trocas, move todos os A's para o início do vetor. Sua função deve consumir tempo $O(n)$.

7. Escreva uma função que rearranje um vetor $v[p \,..\, r]$ de inteiros de modo que tenhamos $v[p \,..\, j-1] \leq 0$ e $v[j \ldots r] > 0$ para algum j em $p \,..\, r+1$. Faz sentido exigir que j esteja em $p \,..\, r$? Procure fazer uma função rápida que não use vetor auxiliar. Repita o exercício depois de trocar $v[j \,..\, r] > 0$ por $v[j \,..\, r] \geq 0$. Faz sentido exigir que $v[j]$ seja 0?

8. Sejam $X[1 \,..\, n]$ e $Y[1 \,..\, n]$ dois vetores, cada um contendo n números ordenados. Escreva um algoritmo $O(\lg n)$ para encontrar uma das medianas de todos os $2n$ elementos nos vetores X e Y. Simule o seu algoritmo detalhadamente para os vetores $X = (1, 2, 3, 4)$ e $Y = (5, 6, 7, 8)$.

9. Para esta questão, vamos dizer que a mediana de um vetor $A[p \,..\, r]$ com número inteiros é o valor que ficaria na posição $A[\lfloor (p+r)/2 \rfloor]$ depois que o vetor $A[p \,..\, r]$ fosse ordenado.

   Dado um algoritmo linear "caixa-preta" que devolve a mediana de um vetor, descreva um algoritmo simples, linear, que, dado um vetor $A[p \,..\, r]$ de inteiros distintos e um inteiro k, devolve o k-ésimo mínimo do vetor. (O k-ésimo mínimo de um vetor de inteiros distintos é o elemento que estaria na k-ésima posição do vetor se ele fosse ordenado.)