MAC–IME–USP

Carlos Eduardo Ferreira
Sala 108C Tel.: 3091 6079
e-mail cef@ime.usp.br

MAC338 - Análise de Algoritmos

Segundo semestre de 2021

Lista3

Todos os exercícios são importantes para o aprendizado da disciplina

Vocês devem entregar os exercícios 1 e 7 até **24 de setembro**

1. Seja $M(n)$ definida pela recorrência

$$
\begin{aligned}
M(0) &= 1 \\
M(1) &= 1 \\
M(n) &= \min_{0 \leq k \leq n-1} \{M(k) + M(n-k-1)\} + n \quad \text{para } n = 2, 3, 4, \ldots
\end{aligned}
$$

   Mostre que $M(n) \geq \frac{1}{2}(n+1)\lg(n+1)$ para todo $n \geq 0$.

2. Qual é o consumo de espaço do Quicksort no pior caso?

3. 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:

   Algoritmo Quicksort (A, p, r)

```
enquanto p < r
    q ← Particione (A, p, r)
    Quicksort (A, p, q − 1)
    p ← q + 1
```

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

4. 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 ← 0
2. segundo_maior ← 0
3. para i ← 1 até n faça
4.     se v[i] > maior
5.         então segundo_maior ← maior
6.               maior ← v[i]
7.         senão se v[i] > segundo_maior
8.                  então segundo_maior ← 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.

5. 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 ← v[1]
2. menor ← v[1]
3. para i ← 2 até n faça
4.     se v[i] > maior
5.         então maior ← v[i]
6.         senão se v[i] < menor
7.                  então menor ← 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?

6. **(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 $E[X]^2$?

7. Considere o programa abaixo que calcula a soma dos pares de um vetor v com n elementos sorteados ao acaso em $[1, 10]$.

Algoritmo SomaPares (v, n)

```
1. soma ← 0
2. para i ← 1 até n faça
```

    se $v[i]$ é par
        então *soma* $\leftarrow$ *soma* $+ v[i]$
devolva *soma*

Suponha que a probabilidade de que o elemento seja qualquer elemento em $[1, 10]$ é a mesma. Observe que $soma$ é uma variável aleatória. Qual o valor esperado da variável $soma$ no fim da execução do programa? Escreva um programa na sua linguagem de programação predileta que verifica empiricamente o resultado que você obteve.