Lista 2 — Algoritmos Probabilísticos **[JD 24-3]**

Exercício 1 (variáveis geométricas não têm memória). Sejam $X \sim \text{Geom}(p)$ uma variável aleatória, $t \geqslant 0$ e $n \geqslant 1$ dois números inteiros. Prove que $\mathbb{P}[\, X = n+t \mid X > t\,] = \mathbb{P}[\, X = n\,]$ e que $\mathbb{P}[\, X \geqslant n+t \mid X > t\,] = \mathbb{P}[\, X \geqslant n\,]$.

Exercício 2. Prove as seguintes propriedades para variáveis aleatórias discretas.

1. Se $\mathbb{P}[\, X \leqslant Y\,] = 1$ então $\mathbb{E}\, X \leqslant \mathbb{E}\, Y$.
2. Se $\mathbb{P}[\, a \leqslant X \leqslant b\,] = 1$ então $a \leqslant \mathbb{E}\, X \leqslant b$.
3. Se $X \geqslant 0$ e $\mathbb{E}\, X = 0$ então $\mathbb{P}[\, X = 0\,] = 1$.

Exercício 3. Seja π uma permutação de $\{1, 2, \dots, n\}$. Para todos $1 \leqslant i < j \leqslant n$ o par (i, j) determina um inversão de π se $\pi(i) > \pi(j)$. Determine o número esperado de inversões numa permutação escolhida ao acaso.

Exercício 4. Sejam U e M conjuntos finitos. Uma família de funções $\mathcal{H} \subseteq M^U$ que quando munida da medida uniforme satisfaz

$$\mathbb{P}_{h \in_R \mathcal{H}}[h(u) = i] = \frac{1}{|M|}, \text{ para todo } u \in U \text{ e para todo } i \in M \tag{1}$$

não é suficiente para garantir um bom comportamento da família de funções de *hash* numa tabela de espalhamento. Verifique que a família $\mathcal{H}$ formada pelas $|M|$ funções constantes satisfaz (1).

Exercício 5. Projete um algoritmo aleatorizado que recebe uma lista de n números e devolve o k-ésimo maior elemento da lista. O número esperado de comparações deve ser $\leqslant cn$ para alguma constante positiva c (*dica:* use o particionamento do *quicksort*).

Exercício 6. Dado um conjunto de n números e um real positivo ε, uma ε-aproximação da mediana é um elemento cujo posto (posição considerando os mesmo elementos em ordem crescente) está no intervalo $[(1-\varepsilon)n/2, (1+\varepsilon)n/2]$. Projete um algoritmo aleatorizado para computar uma 1/2-aproximação da mediana de um vetor com n elementos com tempo de execução $O(\log n \log\log n)$ e probabilidade de erro $2n^{-2}$ (*dica*: amostre e ordene).

Exercício 7. Na seção 3.1.2 foi dito que se uma tabela de espalhamento usa uma função aleatória escolhida numa família universal, então o número esperado de colisões para S fixo é $\binom{n}{2}(1/m) = n(n-1)/(2m)$ de modo que se $m = n^2$ então $\mathbb{E}\, C < 1/2$ e não há colisão com probabilidade pelo menos 1/2. Portanto, se S é um conjunto fixo (estático) podemos sortear uma função até que encontremos uma que não ocasiona colisões para elementos de S. Escreva um algoritmo aleatorizado que dado S encontra uma função de *hash* h *perfeita*. Determine a complexidade desse algoritmo. É possível usar o método das esperanças condicionais nesse caso?

Exercício 8. Sejam X e Y variáveis aleatórias discretas e reais. Prove que para qualquer função $h\colon Y(\Omega) \to \mathbb{R}$

$$\mathbb{E}\,(X - \mathbb{E}[\, X \mid Y\,])^2 \leqslant \mathbb{E}\,(X - h(Y))^2$$

ou seja, $\mathbb{E}[\, X \mid Y\,]$ é a função de Y que melhor aproxima X no sentido de ter o menor erro quadrático médio. Agora, prove que vale a igualdade se, e só se, existe uma função $g\colon Y(\Omega) \to \mathbb{R}$ tal que $g(Y) = \mathbb{E}[\, X \mid Y\,]$ com probabilidade 1.

Exercício 9. Suponha que temos uma fonte de bits aleatórias que responde 0 com probabilidade $p \in (0,1)$ e responde 1 com probabilidade $1-p$, independentemente. Escreva um algoritmo que usa essa fonte responda 0 ou 1 uniforme e independentemente. Calcule o tempo esperado de execução em função de p.

Exercício 10. Prove que se distribuímos ao acaso n bolas em m caixas com a condição que $n < (2/e)m\ln(m)$, então com alta probabilidade o maior número de bolas em uma caixa é

$$\frac{4\ln(n)}{\ln\left(\frac{2n}{me}\ln(n)\right)}.$$