MAC 310 MATEMÁTICA CONCRETA

PROVA 2

1. Instruções

A duração desta prova é de até 2 horas.

2. Questões

1. (valor 1.5 pontos) Resolva as seguintes relações de recorrência:
   (a) $a_n = 5a_{n-1} + 6a_{n-2}, n \geq 2, a_0 = 1, a_1 = 3$;
   (b) $a_{n+1} = 4a_n - 4a_{n-1}, n \geq 1, a_0 = 7, a_1 = 4$;
2. (valor 1.5 pontos) Resolva a recorrência abaixo onde $n = 2^k$ para algum inteiro positivo k.
$$\begin{aligned} T(1) &= 4 \\ T(n) &= 2T(n/2) + 3n + 2 \quad \text{para } n > 1. \end{aligned}$$
3. (valor 1.5 pontos) Resolva a recorrência abaixo usando o método das funções geradoras:
$$a_{n+1} + a_n = 2n + 3, n \geq 0, a_0 = 1;$$
4. (valor 2.0 pontos) Considere a função `random`(a, b) que devolve um inteiro k tal que $a \leq k \leq b$. Assumindo que cada chamada da função `random` gasta tempo $\Theta(1)$, descreva um algoritmo aleatório de complexidade de tempo $\Theta(n)$ que dado um vetor $A[1..n]$ permuta os elementos do vetor A aleatoriamente (i.e. cada permutação do vetor A ocorre com probabilidade $1/n!$ como resposta do algoritmo.)
5. (**Teste de igualdade**) Suponha que existam dois bancos de dados em computadores distintos. Os computadores podem se comunicar via modem a uma velocidade de 1.200 bps, ou seja, a transmissão de dados é bastante lenta e portanto cara. O problema é saber se os bancos de dados são iguais. Uma solução é transferir um banco de dados de um computador para outro e depois verificar se são os mesmos. Outra solução, que iremos analisar aqui, é de se transferir uma seqüência de caracteres muito mais curta que o banco de dados e baseado nessa seqüência se decidir com grande margem de segurança se os arquivos são iguais.

   O problema pode ser formulado de uma maneira mais geral. Duas pessoas, *Alice* e *Bob* querem computar o valor de uma função booleana $f(x, y)$. Alice e Bob têm poder computacional ilimitado, conhecem a

Date: 26 de junho de 1997.

função f e concordam em colaborar da melhor maneira possível para calucular a função f. O problema é que Alice e Bob ficarão separados e um valor de x será fornecido somente à Alice enquanto um valor de y será fornecido somente ao Bob. Separados, eles podem se comunicar através de um canal de comunicação transmitindo bits. Antes da separação eles têm toda a liberdade para combinar qual a maneira mais eficiente de se computar f. Quantos bits de comunicação entre Alice e Bob são necessários (ou suficientes) para se computar a função numa dada entrada (x, y)? É claro que a resposta vai depender da função f. Esse tipo de problema é estudado num ramo da computação conhecido como *Complexidade de Comunicação.*

No nosso caso, considerando-se x e y como codificação binária dos bancos de dados, o valor da função f é SIM se $x = y$ e NÃO caso contrário. Seja $n := |x|$ (i.e. n é o número de bits de x). Vamos examinar um algoritmo aleatório que, usando $O(\log n)$ bits de comunicação, verifica se $x = y$ com grande margem de segurança.[1]

Algoritmo Teste_Igualdade
Entrada: Alice conhece uma seqüência x de n bits e Bob conhece uma seqüência y de n bits. As seqüência x e y são vistas como números.
Saída: SIM se $x = y$ ou NÃO se $x \neq y$.
início

- **Passo 1:** Alice escolhe aleatoriamente um número primo p com no máximo $c \log n$ bits, ou seja, $p < n^c$; (c é uma constante a ser escolhida de acordo com a margem de segurança desejada);
- **Passo 2:** Alice envia para Bob os valores de p e $x \pmod p$;
- **Passo 3:** Bob verifica se $x \equiv y \pmod p$;
  em caso afirmativo a resposta é SIM; caso contrário a resposta é NÃO.

fim.

PERGUNTAS

(a) (valor 0.5 ponto) Qual é o número de operações bit-a-bit feitas no Passo 2?
(b) (valor 1.5 pontos) Como você faria o Passo 3? Qual é o número de operações aritméticas e operações bit-a-bit do seu algoritmo para executar este passo?
(c) (valor 1.0 pontos) No caso em que $x = y$ o algoritmo nunca erra. Estime a probabilidade de que $x \neq y$ e o algoritmo responda SIM. Em outras palavras, estime

$$\Pr(\text{ERRO}) = \Pr_{p<n^c}(x \neq y \text{ e } x \equiv y \pmod p).$$

[1]Comunicando apenas 64 bits podemos decidir, com probabilidade de erro menor que 10^{-4}, sobre a igualdade ou não de duas sequências de 100.000 bits.

(d) (valor 0.5 ponto) Qual é o número de bits que Alice deve enviar para Bob para que tenhamos

$$\Pr(\textsc{erro}) = O(\frac{1}{n^{10}}).$$

Para fazer o item (c) acima você pode usar os seguintes resultados da Teoria dos Números.

Teorema 1 (Teorema dos Números Primos)**.** *Seja $\pi(n)$ o número de números primos menores do que n ($n = 0, 1, 2, \dots$). (Por exemplo, $\pi(10) = 4$.) Para $n \geq 2$ temos que*

$$\pi(n) \approx \frac{n}{\ln n}.$$

[A aproximação $n/\log n$ dá uma estimativa muito boa de $\pi(n)$. Por exemplo, para $n = 10^9$ o erro é menor do que 6%, pois $\pi(n) = 50.847.478$ e $n/\ln n \approx 48.254.942$.]

Teorema 2. *Seja $\vartheta(n)$ o número de divisores primos de n. Temos que*

$$\vartheta(n) = O(\frac{\ln n}{\ln \ln n}).$$