EXERCÍCIOS

MAC430 ALGORITMOS E COMPLEXIDADE DE COMPUTAÇÃO

2o. SEMESTRE DE 2012

Instruções:

1. Os problemas e exercícios com data de entrega devem ser resolvidos e entregues. Estes exercícios farão parte de sua avaliação.
2. A entrega dos exercícios deve ser feita no começo da aula (as datas de entrega serão datas de aula).
3. O monitor levará em conta não apenas a correção de sua solução, mas também a apresentação (soluções mal escritas valerão *consideravelmente menos*).
4. Você deve fazer os demais problemas e exercícios, mas não é necessário entregar suas soluções.
5. Não deixe de atacar os desafios, mas eles podem resistir às suas investidas mais seriamente que os problemas.
6. A entrega de soluções de problemas e desafios sem data de entrega pode contribuir em sua avaliação.

Exercícios, problemas e desafios:

1. Seja X um conjunto e denotemos por $\mathcal{P}^{<\infty}(X)$ o conjunto dos subconjuntos finitos de X. Seja $f\colon \omega = \{0, 1, 2, \dots\} \to \mathcal{P}^{<\infty}(X)$ uma função e suponha que, para todo $n \in \omega$, existe uma função $g_n\colon n = \{0, \dots, n-1\} \to X$ tal que
   (a) $g_n(i) \in f(i)$ para todo $i \in n$
   e
   (b) os $g_n(i)$ $(i \in n)$ são todos distintos.
   Prove que existe uma função $g\colon \omega \to X$ tal que (a) $g(i) \in f(i)$ para todo $i \in \omega$ e (b) os $g(i)$ $(i \in \omega)$ são todos distintos. Se removemos a hipótese de que $f(i)$ é finito para todo i, o resultado ainda vale? $\{$*Data de entrega*: 7/8/2012$\}$
2. (Problema) Prove a seguinte versão mais geral do Exercício 1. Seja $f\colon Y \to \mathcal{P}^{<\infty}(X)$ uma função e suponha que, para todo $Z \subset Y$ com Z finito, existe uma função $g_Z\colon Z \to X$ tal que
   (a) $g_Z(i) \in f(i)$ para todo $i \in Z$
   e
   (b) os $g_Z(i)$ $(i \in Z)$ são todos distintos.
   Prove que existe uma função $g\colon Y \to X$ tal que (a) $g(i) \in f(i)$ para todo $y \in Y$ e (b) os $g(i)$ $(i \in Y)$ são todos distintos.
3. Suponha que X e Y sejam conjuntos e suponha que existam funções injetoras $f\colon X \to Y$ e $g\colon Y \to X$. Prove que existe uma bijeção $h\colon X \to Y$. $\{$*Data de entrega*: 7/8/2012$\}$
4. Sejam G e H dois grafos. Suponha que G seja isomorfo a um subgrafo de H e H seja isomorfo a um subgrafo de G. Mostre que, se os grafos G e H são finitos, então G é isomorfo a H. $\{$*Data de entrega*: 9/8/2012$\}$

Date: Versão de 12 de dezembro de 2012.

5. Mostre que, se omitimos a hipótese de que G e H são grafos finitos no Exercício 4, então a conclusão daquele exercício é falsa. {*Data de entrega*: 9/8/2012}
6. Mostre que a linguagem $L = \{0^k 1^{2k} \colon k \in \mathbb{N}\}$ é decidível. {*Data de entrega*: 14/8/2012}
7. Um conjunto $A \subset \mathbb{Z}$ é *diofantino* se existe um natural n e um polinômio com coeficientes inteiros $p(t, x_1, \dots, x_n)$ de $n+1$ variáveis tal que $A = \{a \in \mathbb{Z} \colon \text{existe } (r_1, \dots, r_n) \in \mathbb{Z}^n \text{ tal que } p(a, r_1, \dots, r_n) = 0\}$.
   - (i) Prove que todo conjunto diofantino é Turing-reconhecível.
   - (ii) Matijasevič provou que todo conjunto Turing-reconhecível $A \subset \mathbb{Z}$ é diofantino. Use o fato que existem conjuntos Turing-reconhecíveis $A \subset \mathbb{Z}$ que não são decidíveis para deduzir que a resposta ao 10o. Problema de Hilbert é negativa, isto é, que não existe um algoritmo que, dado um natural n e um polinômio com coeficientes inteiros $q(x_1, \dots, x_n)$ com n variáveis, decide se existe $(r_1, \dots, r_n) \in \mathbb{Z}^n$ tal que $q(r_1, \dots, r_n) = 0$.

   {*Data de entrega*: 16/8/2012}
8. (Continuação da Questão 7) Seja $P = \{2, 3, 5, 7, \dots\}$ o conjunto dos números primos. Prove que existe um natural n e um polinômio de coeficientes inteiros $q(x_1, \dots, x_n)$ tal que
$$P = \{q(r_1, \dots, r_n) > 0 \colon (r_1, \dots, r_n) \in \mathbb{N}^n\}. \tag{1}$$
Isto é, os valores positivos que q assume, quando avaliado em argumentos naturais, é exatamente o conjunto dos números primos. [*Sugestão.* Use o Teorema de Matijasevič, enunciado na Questão 7. Considere o polinômio $q(t, x_1, \dots, x_n) = (t+1)(1-p^2(t, x_1, \dots, x_n)) - 1$, onde $p(t, x_1, \dots, x_n)$ é o polinômio na definição de conjuntos diofantinos.] {*Data de entrega*: 21/8/2012}
9. (i) Elabore o conceito de *máquina de Turing com infinitas fitas.* A função de transição deve ser da forma
$$\delta \colon Q' \times \Gamma^* \to Q \times \Gamma^* \times \{L, R\}^*, \tag{2}$$
onde, como de usual, $Q' = Q \setminus \{q_{\text{ac}}, q_{\text{rej}}\}$. Seja $s = \gamma_1 \dots \gamma_k \in \Gamma^*$. O valor de $\delta(q, s)$ diz qual deve ser a transição quando a máquina está no estado q e tem, sob a cabeça da i-ésima fita, a letra γ_i $(1 \le i \le k)$ e a j-ésima fita está vazia para todo $j > k$. Se $\delta(q, s) = (q', s', m)$, então $s' = \gamma'_1 \dots \gamma'_{k'}$ e $m = m_1 \dots m_{k'} \in \{L, R\}^{k'}$ para algum $k' \ge k$, a letra γ'_i deve ser escrita na i-ésima fita, e a cabeça da i-ésima fita deve ser movida de acordo com m_i $(1 \le i \le k')$. Para responder essa questão, basta dizer sucintamente os detalhes que faltam ser especificados.
   - (ii) Suas máquinas com infinitas fitas são equivalentes às MTs usuais? (Naturalmente, prove sua afirmação.)

   {*Data de entrega*: 23/8/2012}
10. Considere o seguinte algoritmo não-determinístico (que aceita toda entrada).

$A =$ "Dada entrada $v \in \Sigma^*$:

1. Mova a cabeça até encontrar um branco.

2. Escolha, não-deterministicamente, uma palavra $w \in \Sigma^*$ e escreva w na fita, imediatamente após v.

3. Aceite v."

(i) O algoritmo A sempre pára?

(ii) Defina uma máquina de Turing não-determinística N que executa A.

(iii) Sua máquina N sempre pára? Isto é, é verdade que, para qualquer entrada v, toda computação de N é finita?

(iv) Sua resposta para (iii) depende de N? Ou todas as máquinas N que executam A têm o mesmo comportamento? (Como sempre, justifique sua resposta.)

{*Data de entrega*: 23/8/2012}

11. Prove que a coleção das linguagens decidíveis é fechada pelas operações de (i) união, (ii) concatenação, (iii) estrela, (iv) complementação, (v) interseção. {*Exercício indicado*; *sem data de entrega*}
12. Prove que a coleção das linguagens Turing-reconhecíveis é fechada pelas operações de (i) união, (ii) concatenação, (iii) estrela, (iv) interseção, (v) homomorfismo. {*Exercício indicado*; *sem data de entrega*}
13. Seja Σ um alfabeto fixo, com as letras totalmente ordenadas por uma relação de ordem dada (por exemplo, $\Sigma = \{0, 1\}$, com $0 < 1$). A *ordem militar* em Σ^* ordena as palavras em Σ^* primeiro por comprimento e depois pela ordem lexicográfica induzida pela ordem em Σ. Prove que $L \subset \Sigma$ é decidível se e só se existe um enumerador que enumera os membros de L em ordem militar. {*Exercício indicado*; *sem data de entrega*}
14. Prove que toda linguagem Turing-reconhecível L contém uma sublinguagem $L' \subset L$ decidível com infinitas palavras. {*Data de entrega*: 30/8/2012}
15. Seja L uma linguagem. Prove que L é Turing-reconhecível se e só se existe uma linguagem decidível D tal que $L = \{x\colon$ existe y tal que $(x, y) \in D\}$. [*Sugestão.* Use máquinas não-determinísticas (interprete y como uma computação de uma MTND com entrada x e D como uma MT que verifica a correção dessa computação).] {*Data de entrega*: 11/9/2012}
16. Seja L uma linguagem que consiste de descrições de *deciders*: $L = \{\langle M_1\rangle, \langle M_2\rangle, \dots\}$, onde todos os M_i são *deciders*. Prove que existe uma linguagem decidível D tal que $D \neq L(M_i)$ para todo i. [*Sugestão.* Considere um enumerador para L.] {*Data de entrega*: 18/9/2012}
17. Digamos que uma MT M tem a propriedade P se o comprimento de toda palavra que M aceita é um número primo. Seja $L = \{\langle M\rangle\colon M$ tem a propriedade $P\}$.

    (i) Use o teorema de Rice para provar que L é indecidível.

    (ii) Dê uma prova direta de que L é indecidível (isto é, sem usar o teorema de Rice).

    {*Data de entrega*: 20/9/2012}
18. Seja L uma linguagem Turing-reconhecível. Prove que $L \leq_{\mathrm{m}} A_{\mathrm{TM}}$. {*Exercício indicado*; *sem data de entrega*}
19. Fixe um alfabeto Σ com $|\Sigma| \geq 2$. Para todo inteiro k, seja

$$\mathrm{PCP}_k = \left\{ \left\{ \begin{bmatrix} t_1 \\ b_1 \end{bmatrix}, \dots, \begin{bmatrix} t_n \\ b_n \end{bmatrix} \right\} \in \mathrm{PCP}\colon t_i,\, b_i \in \Sigma^* \text{ e } |t_i|,\, |b_i| \leq k \text{ para todo } i \right\}. \tag{3}$$

Assim, PCP_k contém as instâncias (positivas) de PCP com todas as palavras t_i e b_i envolvidas de comprimento no máximo k.

(*i*) Prove que PCP_1 é decidível.

(*ii*) Prove que PCP_k é decidível para todo $k \geq 1$. (Como sempre, justifique sua resposta.)

(*iii*) Existe uma máquina de Turing M tal que, quando executado com k, tem como saída uma a descrição de uma máquina de Turing M_k que decide PCP_k?

{*Data de entrega*: 4/10/2012}

20. Use a teoria vista em sala para desenvolver um programa em C que imprime uma listagem de si próprio. Ademais, produza seu programa de modo que ele tenha um comprimento, em bytes, que seja ou uma potência de 2, ou uma potência de 3, ou um número primo. {*Data de entrega*: 14/12/2012}

21. Escolha um dos problemas do capítulo sobre complexidade de tempo do Sipser (Capítulo 7) e resolva. Escolha um problem 'interessante'; uma boa escolha será 'melhor apreciada'. Soluções pouco originais não valerão muito (isto é, procure escolher problemas com soluções não-rotineiras). Como sempre, não deixe de caprichar na escrita de sua solução. {*Data de entrega*: 14/12/2012}

22. Escolha um dos problemas do capítulo sobre complexidade de espaço do Sipser (Capítulo 8) e resolva. Escolha um problem 'interessante'; uma boa escolha será 'melhor apreciada'. Soluções pouco originais não valerão muito (isto é, procure escolher problemas com soluções não-rotineiras). Como sempre, não deixe de caprichar na escrita de sua solução. {*Data de entrega*: 14/12/2012}