PROJETO E ANÁLISE DE ALGORITMOS

Segunda lista
Data de entrega: 17 de junho de 1994

---

1. (a) Escrever um algoritmo polinomial para o problema 2-SAT. Estudar sua complexidade (referência: consultar o problema SATISFIABILITY em [GaJ]).

2. (a) [CLR] Problema 10-2, página 194.

3. (a) Demonstrar que o problema 0-1 KNAPSACK é NP-completo.

4. (b) Dado um problema de otimização P, definir: "algoritmo ϵ-aproximativo", "esquema de aproximação", "esquema de aproximação polinomial" e "esquema de aproximação totalmente polinomial". Propor um esquema de aproximação totalmente polinomial para o problema da mochila em variáveis 0-1, ilustrando ainda com um exemplo numérico.

5. (b) Sejam $(x_1, x_2, \ldots, x_n)$ e $(y_1, y_2, \ldots, y_n)$ dois vetores de números inteiros ordenados em ordem não-decrescente. Escreva um algoritmo para obter a mediana dos $2n$ elementos combinados. Justifique cuidadosamente o algoritmo proposto (referência: J.L. Bentley e M.I. Shamos, "Divide-and-Conquer in Multidimensional Space", *Proceedings of the 8th ACM Symposium on Theory of Computing*, 220–230, 1976).

6. (b) Demonstrar que o problema PARTITION é NP-completo.

7. (c) Seja $G = [V, E]$ um grafo orientado sem ciclos. Descrever um algoritmo de complexidade linear para determinar se o grafo G contem um caminho hamiltoniano.

8. (c) *(Preemptive scheduling)* Dadas m máquinas $M_i, i = 1, \ldots, m$, e n tarefas J_j, cada uma com um tempo de processamento $p_j, j = 1, \ldots, n$, deseja-se obter um escalonamento destas n tarefas nas m máquinas com possibilidade de superposição cuja duração seja mínima. Em dado instante, cada máquina só pode estar executando uma única tarefa e cada tarefa só pode estar sendo executada em uma única máquina. Entretanto, uma tarefa poderá ser decomposta em uma ou mais frações que são executadas em instantes diferentes, desde que, num mesmo instante, no máximo uma única fração esteja sendo executada.

9. (c) Demonstrar que o problema MAX-CUT é NP-completo.

10. (d-e) Seis vagões chegam pela via A na ordem $V_2, V_6, V_3, V_1, V_4, V_5$. Deseja-se reordená-los na saída pela via B pela ordem crescente de seus índices. Sabendo-se que nenhum vagão pode dar marcha-a-ré, descrever um modelo de otimização em grafos que permita resolver este problema, determinando o número mínimo de vias de triagem necessárias. Aplique-o a esta instância.

11. (d-e) Sejam n cidades que devem comunicar-se entre si. Cada cidade i tem no máximo uma mensagem a enviar para a cidade j. Todas as mensagens têm a mesma duração e são transmitidas através de um satélite, cuja capacidade (número de mensagens sendo transmitidas simultaneamente) é suficientemente grande. Uma cidade pode receber uma mensagem e enviar outra simultaneamente, mas jamais receber ou enviar mais de uma mensagem simultaneamente. Formular este problema como um modelo de otimização em grafos, ilustrando através do exemplo correspondente à matriz de tráfego $T = (t_{ij})_{i=1,\ldots,n;j=1,\ldots,n}$ abaixo ($n = 5$), onde $t_{ij} = 1$ se existe uma mensagem a ser enviada da cidade i para a cidade j, $t_{ij} = 0$ caso contrário:

$$T = (t_{ij})_{i=1,\ldots,5;j=1,\ldots,5} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

12. (d) Demonstrar que o problema VERTEX COVER é NP-completo.

13. (e) Demonstrar que o problema DOMINATING SET é NP-completo.