MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Exatas e Biológicas - ICEB
Programa de Pós-Graduação em Ciência da Computação

Disciplina: Projeto e Análise de Algoritmos
Professor: David Menotti Gomes

2ª Lista de Exercícios

Valor: 1,0 pontos (10% da nota total)

Data de Entrega: 03/08/2011

1. O problema de coloração de grafos pede para determinar o menor número de cores que podem ser atribuídas aos vértices de tal modo que vértices adjacentes têm cores diferentes. Mostre que esse problema é NP-difícil.

2. O Problema do Caixeiro Viajante (PCV) com pesos 1 e 2 consiste em encontrar um circuito hamiltoniano em um grafo completo G onde todas as arestas possuem custos 1 ou 2.
   - Transforme este problema em um problema de decisão;
   - Então, mostre uma redução polinomial de um problema conhecidamente pertence a classe NP-completo à este problema de decisão.

3. O problema da árvore de Steiner consiste no seguinte: dado um grafo G com custos nas arestas satisfazendo a desigualdade triangular e um subconjunto S de vértices de G, encontrar uma árvore em G que conecta S e que tenha custo mínimo. Note que a árvore pode ter vértices fora de S. Descreva um algoritmo de 2-aproximação para o problema. Onde você usou o fato dos custos satisfazerem a desigualdade triangular?

4. Prove que 2-SAT-monotônico é NP-completo. Dica: Pense no problema de cobrimento de vértices.

5. Para cada afirmativa a seguir, indique se é *Verdadeira* ou *Falsa* e justifique sua resposta (seja sucinto):

   (a) Se n é um inteiro tal que $a^{n-1} \equiv 1 (\bmod\ n)$ para todo a, $1 \leq a < n$ tal que $mdc(a, n) = 1$, então n é primo.

   (b) A segurança do método de RSA depende do fato de que $P \neq NP$.

   (c) O algoritmo de Euclides para calcular $mdc(a, b)$ é $O(\log^3 a)$, onde $\log a > \log b$.

   (d) O algoritmo para se calcular $a^x (\bmod\ n)$ é $O(\log n \log x)$.

(e) Se $L_1 \propto L_2$ e $L_2 \in NP$ então $L_1 \in NP$.

(f) Se 2-$SAT \propto$ 3-SAT então 2-$SAT \in NP$.

(g) O problema de isomorfismo de grafos é um exemplo de problema NP que não é nem NP-completo nem P.

(h) O algoritmo de programação dinâmica para solucionar o problema da mochila tem complexidade $O(nC)$, onde n é o número de itens e C é a capacidade da mochila, sendo assim polinomial.

Conceitos:

- ***Emparelhamento***: Seja um grafo G com conjunto de vértices V e conjunto de arestas E. Um emparelhamento é um subconjunto de E de arestas duas a duas não adjacentes. Em outras palavras, um emparelhamento é um subconjunto de arestas onde não existe duas arestas incidentes a um mesmo vértice.

- ***Desigualdade triangular***: Seja um grafo G com conjunto de vértices V de tamanho n e conjunto de arestas E. Seja $c(e)$ a função de custo associada à aresta e. Se $c(v_i v_k) \leq c(v_i v_j) + c(v_j v_k)$ para todo i, j e k sendo $1 \leq i, j, j \leq n$, $i \neq j$ e $j \neq k$, então o grafo satisfaz a desigualdade triangular.

- ***2-SAT-monotônico***: Uma fórmula booleana 2-SAT é uma fórmula na forma normal conjuntiva, ou seja, do tipo $C_1 \times C_2 \times ... \times C_l$ onde cada cláusula contém exatamente duas variáveis, como em $C_i = (x_{i1} + x_{xi2})$.

  Uma fórmula 2-SAT é monotônica se não tem variáveis na forma negadas.

  Uma fórmula é satisfatível se existe um assinalamento das variáveis $\{x_1, x_2, ..., x_n\}$ (para `verdadeiro` ou `falso`) que faça a fórmula `verdadeira`.

  Dado um número k e uma fórmula 2-SAT-monotônica conforme descrita acima, o problema 2-SAT-monotônico é o problema de se determinar se esta fórmula é satisfazível para um assinalamento onde temos, no máximo, k variáveis no estado `verdadeiro`.

  Exemplo: Seja $H = (x_1 + x_2) \times (x_2 + x_3) \times (x_3 + x_4) \times (x_4 + x_5) \times (x_1 + x_5) \times (x_3 + x_5)$ e $k = 2$. A responsta a este problema é SIM, dado que temos duas variáveis $x_1 = x_3 =$ `verdadeiro` que fazem fórmula H `verdadeira`.

O que deve ser entregue

- Solução dos exercícios propostos.

Como deve ser apresentada a solução dos exercícios

1. A lista de exercícios deve ser mandatoriamente elaborada em LaTeX. Nenhum outro formato será aceito. Veja modelo em latex: `http://www.decom.ufop.br/menotti/paa101/tps/modelo.zip`

2. Formato final: mandatoriamente em PDF (`http://www.pdf995.com/`).

Como deve ser feita a entrega

A entrega DEVE ser feita via Moodle (`www.decom.ufop.br/moodle`) na forma de um único arquivo zipado, contendo a documentação da solução das questões e arquivos que foram utilizados para elaborá-la.

Comentários Gerais

- Esses exercícios foram extraídos quase que na íntegra da 4º Prova de Projeto e Análise de Algoritmos do Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Minas Gerais, ministrada pelo professor Antônio Alfredo Ferreira Loureiro no 1º semestre de 2010;
- Comece a resolver esta lista logo, enquanto o problema está fresco na memória e o prazo para terminá-lo está tão longe quanto jamais poderá estar;
- A solução da lista é individual (grupo de UM aluno);
- Soluções copiadas (e FONTE) terão nota zero. **Ainda, como forma de coibir o *plágio acadêmico*, alunos "pegos" terão a maior nota dos testes (escrito) levada à zero**;
- Soluções entregues em atraso serão aceitas, todavia a nota atribuída a solução será zero;
- Evite discussões inócuas com o professor em tentar postergar a data de entrega.