UNIVERSIDADE FEDERAL DE MINAS GERAIS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
PROJETO E ANÁLISE DE ALGORITMOS

Última alteração: 5 de abril de 2006

Professor: Nivio Ziviani

Monitor: Fabiano C. Botelho

2º Trabalho Prático - 04/04/06 - 10 pontos

Data de Entrega: 24/04/06

Penalização por Atraso: 1 ponto até 28/04/06 mais 1 ponto por dia útil a seguir

Observação: Toda a documentação deverá ser apresentada como uma página acessível via Web (apresente o link para acesso à documentação).

---

1. O algoritmo a seguir é útil para ser usado em corretores ortográficos. Sejam dadas duas cadeias de caracteres: $X = x_1, x_2, \cdots, x_m$ e $Y = y_1, y_2, \cdots, y_n$. O número k de operações de substituição, inserção e retirada de caracteres necessário para transformar X em Y é conhecido como distância de edição. Assim, a distância de edição ed(X,Y) corresponde ao número k de operações necessárias para converter X em Y.

   Por exemplo, se $X =$ `matranda` e $Y =$ `saturadas` então ed(X,Y)=4. A sequência de operações é: (i) substitui x_1 por y_1 ('`s`'), (ii) insere y_4 ('`u`') após x_3, (iii) retirax_6 ('`n`') e (iv) insere y_9 ('`s`') após x_8.

   (a) Apresente um algoritmo que use programação dinãmica para determinar a distância de edição entre X e Y.
   (b) Forneça a complexidade de tempo e de espaço do seu algoritmo.
   (c) Implemente o seu algoritmo. Registre a sequência de operações realizadas e implemente um procedimento para imprimir a sequência final de operações realizadas.
   (d) Mostre o funcionamento do algoritmo para o exemplo "`matranda`" e "`saturadas`".

2. Dado um grafo G = (V, A), um *conjunto independente de vértices* é um subconjunto $V' \subseteq V$ tal que todo par de vértices de V' não é adjacente (isto é, se $x, y \in V'$ então a aresta $x, y \notin A$).

   Um *conjunto independente maximal*[1] é máximo se todos os outros conjuntos independentes têm cardinalidade menor ou igual. O conjunto $\{2, 3\}$ na Figura 1 não é um conjunto independente maximal enquanto os conjuntos $\{0, 1\}$ e $\{2, 3, 4\}$ são conjuntos independentes maximais, sendo que o conjunto $\{2, 3, 4\}$ é um conjunto independente máximo.

[1] Um conjunto independente é *maximal* quando não existe nenhum outro conjunto independente que o contenha, isto é, um conjunto que não pode ser completado.

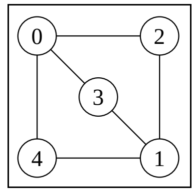

Figura 1: Exemplo de conjunto independente maximal

Este problema tem várias aplicações práticas. Exemplos:

(a) Suponhamos que você queira realizar uma reunião envolvendo o maior número possível de pessoas do seu círculo de amizades que não se conhecem. Dentre as pessoas que poderiam ser convidadas, você traça um grafo contendo uma aresta ligando duas pessoas que se conhecem. O conjunto independente máximo representa o maior conjunto de pessoas que não se conhecem.

(b) Seja um grafo cujos vértices representam projetos que podem ser executados em uma unidade de tempo. Todo projeto que utiliza recursos comuns a um outro projeto são interligados por uma aresta. O conjunto independente máximo representa o conjunto maximal de projetos que podem ser executados em paralelo (simultaneamente) em um único período de tempo.

(c) Problema das oito rainhas: Oito rainhas são colocadas em um tabuleiro de xadrez de tal forma que nenhuma rainha possa atacar diretamente outra rainha. Este problema foi investigado por C. F. Gauss em 1850, que não conseguiu resolvê-lo inteiramente. O problema pode ser generalizado para um tabuleiro qualquer de tamanho $n \times n$. Seja um grafo cujos vértices representam as posições de um tabuleiro. Para cada posição do tabuleiro, interligar por uma aresta todas as posições que possam ser atingidas pela rainha a partir dela. O conjunto independente máximo representa a solução para o problema das n rainhas.

O que fazer:

(a) O problema de encontrar o conjunto independente máximo de um grafo é $\mathcal{NP}$-Completo (redução a partir do clique). Implemente um algoritmo capaz de obter a solução ótima para este problema. Informe o tamanho do maior problema que você conseguiu obter a solução ótima. Comente o resultado indicando o motivo da limitação e faça uma estimativa do tempo necessário no caso de termos uma entrada dez vezes maior que a do maior problema que você resolveu.

(b) Implemente um algoritmo polinomial para obter uma redução tal que você consiga resolver o problema do clique utilizando o algoritmo implementado no item anterior. Mostre o funcionamento do seu algoritmo para o exemplo acima.

(c) Implemente um algoritmo aproximado que resolva este problema eficientemente e produza "boas" soluções sob o ponto de vista prático. Apresente uma análise de complexidade de tempo do seu algoritmo aproximado.

(d) Apresente uma análise indicando o quanto a solução aproximada fornecida se aproxima do resultado ótimo. (Você pode explicar resultados encontrados na literatura ou ainda apresentar sua própria demonstração.)

(e) Realize testes com entradas escolhidas aleatoriamente e compare os resultados com a análise de eficiência apresentada no item anterior. Comente os resultados obtidos e discuta o quão próxima a análise teórica está da realidade.

As referências principais para este exercício são:

Christofides, N. (1975) *Graph Theory An Algorithm Approach*, Academic Press, capítulo 3, pp. 30–35.

Bron, C. e Kerbosch, J. (1973) “Finding All Cliques of an Undirected Graph”, CACM 16 (9), 575–579.

Brélaz, D. “New Methods to Color the Vertices of a Graph”, CACM 22 (4), 251–256.