

PUC-Rio
Departamento de Informática
Profs. Marcus Vinicius S. Poggi de Aragão
Período: 2005.1
Horário: 3as-feiras e 5as-feiras de 15-17
31 de março de 2005
Data da Entrega: 29 de junho de 2005

ESTRUTURAS DISCRETAS (INF 1631)

2º Trabalho de Implementação

SEGURANÇA ESTATÍSTICA DE DADOS - Descrição

O problema de Segurança Estatística de Dados está presente quando um órgão público de levantamento de dados, como o IBGE por exemplo, divulga relatórios onde além de dados agregados divulga também alguns dados individuais, mas sem revelar outros dados.

De forma mais precisa, considere que uma tabela $T(n, m)$ de n colunas e m linhas armazena as informações em questão, que no caso são valores inteiros positivos. As somas dos valores nas linhas e nas colunas tem que ser divulgados e além deles o máximo de valores individuais, desde que estes valores divulgados não permitam inferir outros valores da tabela.

Por exemplo, cada coluna pode se referir a uma cidade e cada linha a um setor de atividade do governo. Os valores podem ser os gastos do governo com a atividade associada à linha: saúde, educação, infra-estrutura, etc. Pode ser que os valores destes gastos sejam "confidenciais", mas que alguns tenham que ser revelados, além dos totais por cidade e por atividade para um conjunto de cidades, do mesmo estado por exemplo. O órgão deve divulgar os valores totais por cidade e por atividade e alguns exemplos de gastos com certas atividades em certas cidades (mas só os que agradam!). De forma alguma, os dados divulgados devem permitir que sejam inferidos outros valores (o que permitiria deduzir que o gasto com saúde em uma dada cidade foi exageradamente baixo, por exemplo).

Assim, para uma tabela $n \times m$ de elementos t_{ij} positivos ou nulos, são dadas as somas das linhas r_i para $i = 1, \dots, m$ e as somas das colunas c_j para $j = 1, \dots, n$; e uma lista de p posições da tabela $\{(i_1, j_1), (i_2, j_2), \dots, (i_p, j_p)\}$ que tem seus valores associados t_{i_k, j_k} , $k = 1, \dots, p$, divulgados, deseja-se determinar se alguma outra posição na tabela também fica determinada.

ALGORITMO PARA RESOLUÇÃO

Para determinar se uma dada posição está com o seu valor determinado construa um grafo bi-partido $G = (L, C, E)$ onde L é o conjunto de vértices associado às linhas da matriz, C é

o conjunto de vértices associado às colunas da matriz, e E o conjunto de arcos composto dos arcos que saem de cada vértice de L para todos os vértices de C .

Associe a cada vértice em L um fluxo fixo de valor r_i que entra no vértice e a cada vértice em C um fluxo fixo de valor c_j que sai do vértice. Observe que a conservação de fluxo (tudo que entra tem que ser igual a tudo que sai) nos vértices do grafo correspondem exatamente às somas dos valores das linhas (vértices em L) e às somas dos valores das colunas (vértices em C).

Em seguida, para cada valor divulgado $t_{i_k j_k}$, $k = 1, \dots, p$, retire o arco (i_k, j_k) do grafo e subtraia $t_{i_k j_k}$ do fluxo que entra no vértice i_k em L e do fluxo que sai de j_k em C . Represente por r'_i e c'_j os valores dos fluxos que entram e que saem dos vértices de L e C , respectivamente, após a retirada dos valores conhecidos.

Obtenha para o grafo resultante $G' = (L, C, E')$ com os fluxos fixos de entrada em L de valores r' e fluxos fixos de saída em C de valores c' um **fluxo viável**. Isto é, valores positivos ou nulos x_{ij} para cada arco em E' que satisfaçam a conservação de fluxo em cada vértice.

Para isso associe aos arcos em E' capacidades de fluxo da seguinte forma: o arco (i, j) terá capacidade de fluxo máxima u_{ij} igual ao menor valor entre r'_i e c'_j . Assim, $0 \leq x_{ij} \leq u_{ij}$ para todo arco de E' . Siga o algoritmo abaixo:

1. Faça inicialmente $x_{ij} = 0$ para todo arco em E' . Seja x esse fluxo.
2. Determine o grafo $G'(x)$, a rede residual, onde para cada arco (i, j) de E' existirá em $G'(x)$ um arco (i, j) com capacidade $u_{ij} - x_{ij}$, se $x_{ij} < u_{ij}$, e um arco (j, i) com capacidade x_{ij} , se $x_{ij} > 0$.
3. Se existir pelo menos um vértice em L e pelo menos um vértice em C onde ainda não há conservação de fluxo, vá para 4. Senão, x é um fluxo viável. FIM.
4. Encontre um caminho em $G'(x)$ de um vértice em L (ainda sem conservação de fluxo) a um vértice em C (também ainda sem conservação de fluxo). Seja u_{min} a menor capacidade entre os arcos deste caminho. Aumente de u_{min} o valor de x_{ij} de todos os arcos do caminho, se o arco do caminho for (j, i) subtraia u_{min} de x_{ij} . Obtém se assim um novo fluxo x . Volte para 2.

Para determinar se um dado valor desconhecido t_{ij} está determinado, basta verificar na rede residual do fluxo viável x , $G'(x)$, se existe caminho de i para j e de j para i , onde esse caminho não pode usar os arcos (i, j) e (j, i) . Se existir um dos dois caminhos, o valor NÃO está determinado. Se nenhum dos dois existirem, o valor estará determinado e x_{ij} será este valor.

EXPERIMENTAÇÃO

1. Programe o algoritmo obtido descrito acima.

2. Execute o algoritmo para o arquivo de dados deste 2o. Trabalho (disponível na página *web* do curso). A especificação dos arquivo de entrada e as saídas desejadas encontram-se mais abaixo.
3. Determine o tempo de CPU gasto para cada caso teste no arquivo de dados (utilize o mesmo procedimento do 1o. Trabalho).
4. BONUS: Quando o valor da posição pedida não pode ser obtido, é possível determinar um intervalo para este valor. Será concedido um bonus de 50% (ou seja o trabalho vale 15!) para os trabalhos que não somente determinarem se os valores pedidos podem ser inferidos, mas que forneçam os intervalos possíveis para todos as posições pedidas das tabelas.
5. Ainda no bonus, determine os tempos de CPU de cada caso tese.

CONCLUSÕES

Escreva suas conclusões analisando os resultados obtidos e o tempos de CPU. Em que condições os valores das tabelas podem ser estimados com uma boa precisão ? Você considera que o algoritmo implementado é eficiente ?

ENTREGA DO TRABALHO

O trabalho pode ser feito em grupos de até **3 (três)** alunos. O trabalho entregue deve conter:

- Um documento contendo uma análise do problema e do algoritmo proposto para a sua resolução. Discuta sobre possíveis alternativas. Apresente comentários e análises sobre a implementação e os testes realizados (**PAPEL**).
- A impressão do código fonte (papel).
- Um e-mail para **marcus.poggi@gmail.com** contendo o código fonte e o executável (“`.exe.txt`”, para passar no gmail) correspondente (é obrigatório o uso do ASSUNTO (ou SUBJECT) ED051T2).

Descrição do Arquivo de Entrada

A primeira linha contém um inteiro N contendo o número de casos teste. Cada caso teste possui na sua primeira linha um inteiro m . A linha seguinte possui m inteiros correspondendo às somas r_i das linhas da matriz (desconhecida). Na próxima linha está um inteiro n que é o número de colunas da matriz. Segue uma linha com os n inteiros correspondentes às somas dos valores nas colunas. Logo abaixo está um número p que é o número de posições reveladas da matriz. Cada uma das p linhas que seguem possuem 3 inteiros: i , j e t . Onde t é o valor

da posição (i, j) que é revelada, sendo $1 \leq i \leq m$ e $1 \leq j \leq n$. Finalmente, a linha seguinte contém um inteiro q que define o número de posições em que deverá ser verificado se o seu valor pode ser obtido a partir das informações anteriores. Estas posições aparecem nas q linhas que seguem onde cada uma possui dois inteiros i e j indicando a posição. O arquivo termina com um 0(zero). Um exemplo com 1 caso teste segue.

```
1
4
29 31 28 13
4
26 22 14 39
5
2 2 5
2 4 21
3 1 3
3 4 12
4 3 1
3
1 3
2 1
4 4
0
```

Descrição do Arquivo de Saída

O arquivo de saída contém o número do caso teste e as respostas sim ou não para cada posição que deseja-se saber se pode ser obtida. No caso afirmativo, apresente o valor.

No caso da versão BONUS, apresente para o caso negativo os intervalos menores possíveis que podem ser inferidos.