# MC658 - Análise de Algoritmos III

Prof. Cid C. de Souza – IC-UNICAMP – 1º semestre de 2019

Lista de Exercícios de PLI e matrizes TU[1]

## Programação linear inteira (PLI)

1. Considere o problema de se achar um conjunto independente máximo em um grafo. Formule este problema como um programa linear inteiro.

2. Considere o problema da cobertura por conjuntos com pesos: Temos um conjunto de elementos $S$ e subconjuntos $S_1, \ldots, S_k$ de $S$, onde cada subconjunto $S_i$ possui um peso $p_i$. Deseja-se achar uma coleção $C = \{S_p, \ldots, S_q\}$ dos subconjuntos satisfazendo $S = \cup_{S_i \in C} S_i$ e tal que a soma dos pesos dos subconjuntos de $C$ seja mínima. Formule este problema como um programa linear inteiro.

3. Suponha que você está modelando um PLI em que deve-se escolher um conjunto de investimentos $\{1, \ldots, 7\}$. Usando variáveis binárias modele as seguintes restrições:

   (a) Não se pode aplicar em todos os investimentos.
   (b) Deve-se escolher pelo menos um dos investimentos.
   (c) O investimento 1 não pode ser escolhido se o investimento 3 for escolhido.
   (d) O investimento 4 só pode ser escolhido caso o investimento 2 seja escolhido.
   (e) Ambos investimentos 1 e 5 devem ser escolhidos ou nenhum dos dois pode ser escolhido.

4. Hermes&Renato devem apresentar propagandas de seu programa de TV pelo menos 4 vezes por dia. As propagandas são apresentadas durante outros programas da emissora. Por isso, Hermes&Renato devem aparecer nas gravações destes programas para fazerem a sua propaganda. Suponha que uma gravação de um programa, seja ela qual for, tem duração de 1 hora. Além disso, são várias as gravações de cada programa em um dia. Estas gravações iniciam-se sempre nas horas cheias, começando às 10 da manhã e terminando às 7 da noite. A escolha de Hermes&Renato por qual gravação e de qual programa eles pretendem participar é influenciada pela reputação do programa, quem é a apresentadora, horário da gravação etc. Vamos denotar por $p_{ik}$ o peso da preferência de Hermes&Renato pela gravação do programa $k$ que começa na hora $i$.

   (a) Formule um PLI que gere um escalonamento de quais gravações Hermes&Renato devem participar de tal forma que se maximize a soma das preferências da dupla. Explique cada uma das restrições do seu modelo.
   (b) Modifique a formulação para que Hermes&Renato não participem de mais de duas gravações seguidas sem antes terem um intervalo.
   (c) Modifique a formulação para que Hermes&Renato tenham um escalonamento em que suas gravações comecem o mais tarde possível.

5. Seja um grafo não-orientado $G = (V, E)$ com custos não-negativos $w_e$ associados a toda aresta $e \in E$. Dado um subconjunto $U$ de vértices de $V$, o *corte* definido por $U$ em $G$ é o conjunto de arestas $(i, j)$ onde $i \in U$ e $j \notin U$, o qual é denotado por $\delta(U)$.

[1] Esta lista compila exercícios de semestres anteriores em que a disciplina foi ministrada por vários docentes do IC.

O *problema do corte* <u>*máximo*</u> em $G$ é $\mathcal{NP}$-difícil. Neste problema deve-se encontrar o subconjunto $U$ de vértices tal que a soma das arestas em $\delta(U)$ é maximizado. Formule este problema usando Programação Linear Inteira. Explique o significado de cada variável e de cada restrição do seu modelo.

*Dica*: use variáveis binárias para representar os vértices que estão em $U$ e as arestas que estão em $\delta(U)$.

6. Uma firma encontra-se em fase de expansão e está planejando a sua mudança para um novo prédio onde ocupará $Z$ andares. Cada andar dispõe de $W$ m$^2$ de área útil. A firma é subdividida em $n$ setores. Cada setor $i \in \{1, \ldots, n\}$ necessita de $w_i$ m$^2$ de área para se instalar. Supõe-se que esta área é inferior à área de um andar e que um mesmo setor não será distribuído em mais de um andar. Além disso, a necessidade de interação entre o pessoal de dois setores $i$ e $j$ foi medida pelos administradores da firma. O problema é encontrar uma alocação dos setores aos andares que minimize o *custo* total de manter separados os diversos pares de setores da firma. Para um par de setores $i$ e $j$ o custo positivo da mudança será $c_{ij} > 0$ se $i$ e $j$ ficarem em andares distintos (independentemente de quantos andares os separam) e *zero* caso eles fiquem no mesmo andar. Formule este problema usando Programação Linear Inteira. Explique o significado de cada variável e de cada restrição do seu modelo.

7. Uma biblioteca recebeu $n$ livros novos. Para cada livro $k$ são conhecidos a sua altura $a_k$ e a sua largura $\ell_k$. Foi decidido que os livros serão armazenados em estantes de altura $H$ e de largura $L$ fixas. Existe uma flexibilidade de se instalar uma quantidade variável de prateleiras nas estantes. Assim, se forem instaladas $i$ prateleiras em uma estante, haverá $i + 1$ vãos na estante disponíveis para colocar os livros naquela estante. Note que cada uma das prateleiras terá largura igual à largura da estante, ou seja, $L$.

   Para instalar uma prateleira tem-se um custo $c_p$ enquanto que cada estante custa $c_e$. O objetivo é determinar quantas estantes comprar e quantas prateleiras instalar em cada uma delas de modo a minimizar o custo da biblioteca. Para tanto, pede-se que o problema seja equacionado usando Programação Linear Inteira. Explique o significado de cada variável e de cada restrição do seu modelo.

   Supõe-se que: (1) todos livros devem ser guardados em pé, (2) a profundidade das prateleiras é suficiente para acomodar qualquer livro, (3) a espessura de uma prateleira é desprezível e (4) existe uma quantidade conhecida $j_{\max}$ de estantes que já se sabe ser capaz de armazenar todos os livros.

   *Dicas*: a altura mínima de um vão entre duas prateleiras (ou entre o fundo ou o topo da estante e uma prateleira) não deve ser inferior à altura de um livro. Logo, é possível estabelecer um limite máximo $N$ sobre o número de prateleiras em uma estante.

8. Formule como um PLI o problema de dispor 8 rainhas em um tabuleiro de xadrez de tal forma que as rainhas não se ataquem, ou seja, não compartilhem uma mesma linha, nem uma mesma coluna e nem uma mesma diagonal. Explique o significado de cada variável, da função objetivo e de cada restrição do seu modelo.

9. Formule como um PLI o seguinte problema. Um sistema operacional para um computador paralelo tem $P$ processadores idênticos. Neste computador devem ser escalonadas $n$ tarefas $1, \ldots, n$, cada tarefa $i$ gasta tempo de execução $t_i$. O sistema operacional tem o problema de distribuir as tarefas para os $n$ processadores de tal maneira que o tempo para executar todas as tarefas *(makespan)* nestes computadores seja mínimo.

10. Formule como um PLI o seguinte problema. Dado um conjunto $F$ de potenciais facilidades a serem abertas com um custo $c_i$ para a abertura da facilidade $i \in F$ e um conjunto de clientes $C$ e arestas $ij$, ligando a facilidade $i$ ao cliente $j$ com custo $c_{ij}$. Adicionalmente, cada facilidade $i \in F$ não pode se ligar a mais que $b_i$ clientes *(capacitated facility location).* O objetivo é encontrar um subconjunto das facilidades $O \subseteq F$ que devem ser abertas de maneira a minimizar a soma do custo para abrir estas facilidades juntamente com o custo da atribuição de cada cliente com uma das facilidades aberta.

11. Formule como um PLI o problema do *bin-packing*, ou problema do empacotamento unidimensional, o qual consiste em: dados lista de itens $L$, cada item $i \in L$ com tamanho $s_i$, onde $0 < s_i \leq C$, encontrar uma partição de $L$ em conjuntos $L_1, \ldots, L_k$ tal que $s(L_i) \leq C$ e $k$ é mínimo.

12. Considere o Problema da *Satisfatibilidade Máxima* no qual são dados uma fórmula booleana $\phi(x_1, \ldots, x_n)$ em forma normal conjuntiva, com cláusulas $C_1, \ldots, C_m$, e peso de uma cláusula $C_i$ é um valor positivo $w_i$, para $i = 1, \ldots, m$. Uma atribuição $A$ para $\phi$ é uma atribuição lógica para as variáveis $x_1, \ldots, x_n$. O peso de uma atribuição $A$ é o peso total das cláusulas satisfeitas por $A$. Apresente uma formulação em programação linear inteira para este problema.

13. **Escalonamento de Tarefas em Máquinas Relacionadas:** Dados uma lista de $n$ tarefas $T = \{1, \ldots, n\}$, cada tarefa $i \in T$ com um valor $t_i$ (referente a tempo) e $m$ máquinas $M = \{1, \ldots, m\}$, de maneira que a máquina $j \in M$ possui velocidade $s_j$. Cada tarefa deve ser alocada (escalonada) em exatamente uma das máquinas. Caso a tarefa $i \in T$ seja alocada na máquina $j \in M$, seu tempo de execução será de $t_i/s_j$ segundos. Após a alocação das tarefas, o tempo total de execução de uma máquina é a soma dos tempos de execução das tarefas alocadas a ela. Nestecontexto, o *makespan* de um escalonamento é definido como o maior tempo de execução de uma máquina, para aquele escalonamento. Neste problema, buscamos por um escalonamento das tarefas com o menor makespan possível. Faça uma formulação em programação linear inteira que resolve este problema. Explique o significado de cada variável e justifique porque a sua formulação de fato modela este problema.

14. Um restaurante abre de segunda a domingo (todos os dias). De acordo com a freqüência dos fregueses, o dono do restaurante sabe a quantidade de funcionários que ele precisa em determinado dia da semana. Esta informação é dada na seguinte tabela:

| Dia da Semana | Domingo | Segunda | Terça | Quarta | Quinta | Sexta | Sábado |
|---|---|---|---|---|---|---|---|
| Número de funcionários | 30 | 32 | 36 | 42 | 45 | 57 | 51 |

Cada funcionário deve trabalhar 5 dias seguidos e descansar os outros dias. O problema consiste em minimizar o número de funcionários contratados.

- Formule o problema através de um modelo de programação linear inteira.
  Sugestão: Use uma variável $x_i$ para representar a quantidade de funcionários começando a trabalhar no dia da semana $i$, onde $i = 1$ corresponde a domingo, $i = 2$ a segunda, assim por diante. Explique com detalhes porque sua formulação modela o problema.
- Suponha agora que o funcionário que começa a trabalhar no dia da semana $i$, ganha um salário $s_i$, $i = 1, \ldots, 7$. O que deve mudar no seu modelo se agora quisermos minimizar o total pago na folha de pagamento dos empregados. Explique seu modelo.

**Matrizes totalmente unimodulares (TU)**

1. Resolva os exercícios a seguir.

   (a) Se $A$ é matriz formada por elementos em $\{-1, 0, 1\}$, e cada coluna tem no máximo uma ocorrência de -1 e no máximo uma ocorrência de 1, então $A$ é TU.

   (b) Mostre que se $A$ é a matriz de incidência de um grafo orientado e $B$ uma matriz obtida a partir de $A$ transformando alguns elementos não nulos em 0, então $B$ também é TU.

   (c) Considere o seguinte problema de seleção de itens. São dados um conjunto de itens $N = \{1, \ldots, n\}$, uma família $\{S_1, \ldots, S_m\}$ contendo $m$ subconjuntos de $N$. Seja $M = \{1, \ldots, m\}$. Se o item $i \in N$ for selecionado, então deve-se pagar o CUSTO $c_i$. Porém, caso todos os itens de um subconjunto $S_j$ sejam selecionados, tem-se um benefício $b_j$. Suponha que os valores de $c_i$ e $b_j$ são positivos. O objetivo é selecionar os itens, de maneira que o valor total dos benefícios conseguidos, menos o valor total dos gastos de todos os itens, é máximo. Usando as propriedades de matrizes TU, mostre que o problema pode ser resolvido em tempo polinomial.

   (d) No problema do conjunto fechado de peso máximo, são dados: um grafo orientado $G = (N, A)$, com conjunto de nós $N$ e conjunto de arcos $A$, e uma função de peso nos nós, $w : V \to \mathbb{Q}$, que podem dar valor positivo ou negativo. Um conjunto $F \subseteq N$ é dito ser fechado, se não há arcos saindo de $F$ para $V \setminus F$. Deseja-se encontrar um conjunto fechado de peso total máximo. Usando as propriedades de matrizes TU, mostre que o problema pode ser resolvido em tempo polinomial.

2. São dados: *(i)* um grafo orientado $G = (V, E, c)$, onde $V$ é o conjunto de nós do grafo, $E$ é o conjunto de arcos e $c$ é uma função de custo nas arestas, $c : E \to \mathbb{Q}^+$ e *(ii)* dois nós $s \in V$ e $t \in V$. Considere o problema de encontrar $k$-caminhos entre os vértices $s$ e $t$ de custo total mínimo. Ou seja, o objetivo é encontrar um conjunto de $k$ caminhos $P_1, \ldots, P_k$, cada um começando em $s$ e terminando em $t$ tal que:

   - Se $E_r$ é o conjunto de arestas em $P_r$, onde $r \in \{1, \ldots, k\}$, então $E_i \cap E_j = \emptyset$ para quaisquer $1 \leq i < j \leq k$.
   - Se $F = E_1 \cup \ldots \cup E_k$ (conjunto de todas os arcos na solução), então $\sum_{e \in F} c_e$ é mínimo.

   (a) Usando as propriedades de matrizes TU, mostre que o problema dos $k$-caminhos aresta-disjuntos de custo total mínimo pode ser resolvido em tempo polinomial.

   (b) Adapte a sua solução para o caso de grafos não-orientados.

   (c) Descreva os problemas dos dois itens anteriores, mas para caminhos disjuntos nos vértices internos (isto é, os $k$ caminhos só se interceptam nos vértices $s$ e $t$).

3. O problema do conjunto independente em grafos bipartidos é definido por: dado um grafo bipartido $G = (V, E)$ e uma função que atribui pesos aos vértices $w : V \to \mathbb{Q}$, encontrar um subconjunto de vértices $I \subseteq V$ não adjacentes entre si (conjunto independente) cuja soma total dos pesos seja máxima. Usando as propriedades de matrizes TU, mostre que o problema de encontrar um conjunto independente de peso máximo em um grafo bipartido pode ser resolvido em tempo polinomial.

4. Considere o seguite problema. Dados grafo bipartido $G = (V, E)$ e uma função que atribui pesos nas arestas $w : E \to \mathbb{Q}$, encontrar um conjunto de arestas $F \subseteq E$, de peso mínimo, tal que para todo vértice $v \in V$, haja pelo menos uma aresta de $F$ incidente em $v$ (cobertura do grafo por arestas). Usando as propriedades de matrizes TU, mostre que o problema de encontrar uma cobertura por arestas de $G$ de peso mínimo em um grafo bipartido pode ser resolvido em tempo polinomial.

5. Considere o seguinte problema, que chamaremos de VAGAS:

   > Problema VAGAS: Um certo país tem $n$ cursos de Computação em universidades públicas, cada curso $i$ (onde $i$ é um inteiro em $\{1, \ldots, n\}$) tem $v_i$ vagas. Há $m$ pessoas que são candidatas a estas vagas, cada candidato $j$ (onde $j$ é um inteiro em $\{1, \ldots, m\}$) fez várias provas dentro de um exame, que chamaremos de EN (Exame Nacional). Como cada curso usa uma ponderação diferente do EN, cada candidato pode ter pesos diferentes para cursos diferentes. Vamos denotar a nota do candidato $j$ para o curso $i$ pelo peso $p_{ij}$ (quanto maior o peso melhor).
   >
   > O governo do país deseja encontrar uma atribuição de candidatos a vagas, de maneira que: *(i)* Cada pessoa $j$ é atribuida a no máximo um curso. *(ii)* Cada curso $i$ recebe a atribuição de no máximo $v_i$ candidatos. *(iii)* A soma total dos pesos das pessoas atribuidas aos correspondentes cursos é máxima.

   Formule o problema VAGAS como um problema de programação linear inteira e, usando as propriedades de matrizes TU, mostre que existe um algoritmo determinístico polinomial que o resolve.