Período: 2014.2 **Disciplina**: D279 – Otimização Combinatória e em Redes

Parte I – Implementação de um modelo para Fixed Charge Capacitated Network Design Problem.

Formulação. Seja G = (N, A) uma rede orientada, onde N é o conjunto de nós e A é o conjunto de arcos. Seja K o conjunto de demandas, cada uma delas caracterizada por uma origem s^k , um destino t^k , e uma quantidade d^k que deve ser transportada da origem para o destino. Seja f_{ij} o custo fixo de utilização do arco ij, c_{ij} o custo variável para transportar uma unidade de fluxo através do arco ij, e u_{ij} a capacidade do arco ij. Considere a variável x_{ij}^k que indica a quantidade de fluxo referente à demanda k no arco ij, e a variável binária y_{ij} que indica se o arco ij é utilizado ou não. O objetivo é minimizar os custos fixos e os custos variáveis.

$$\min \sum_{k \in K} \sum_{ij \in A} c_{ij} x_{ij}^{k} + \sum_{ij \in A} f_{ij} y_{ij}$$
s.t.
$$\sum_{ij \in A} x_{ij}^{k} - \sum_{ji \in A} x_{ji}^{k} = \begin{cases} d^{k}, & \text{for } i = s^{k} \\ -d^{k}, & \text{for } i = t^{k} \end{cases} \quad \forall i \in N, \forall k \in K$$

$$\sum_{k \in K} x_{ij}^{k} \le u_{ij} y_{ij} \quad \forall ij \in A$$

$$x_{ij}^{k} \ge 0 \quad \forall ij \in A$$

$$x_{ij}^{k} \ge 0 \quad \forall ij \in A$$

$$y_{ij} \in \{0, 1\} \quad \forall ij \in A$$

Implementação. Implementar um modelo de programação matemática utilizando Java Concert para o problema. Uma instância deve ser lida de um arquivo de entrada (FCND.dat). Devem ser gerados 3 arquivos de saída: um arquivo com a saída padrão do CPLEX (FCND.log), um arquivo com a formulação matemática (FCND.lp), e um arquivo de solução (FCND.out) conforme especificado abaixo.

Entrada. A primeira linha contém um inteiro n, indicando a quantidade de nós (rotulados de 1 a n), e um inteiro m, indicando a quantidade de arcos. Cada uma das m linhas seguintes possui uma quintupla que caracteriza um arco: $(i, j, f_{ij}, c_{ij}, u_{ij})$. A linha seguinte contém um inteiro k indicando a quantidade de demandas. Cada uma das k linhas seguintes possui uma tripla que caracteriza uma demanda: (s^k, t^k, d^k) .

Saída. O arquivo de solução deve apresentar o custo total, o tempo e os arcos efetivamente utilizados.

Exemplo de entrada:

_	0			
6	8			
1	2	100	1	100
2	3	50	1	60
2	4	100	1	100
3	1	100	1	100
4	6	100	1	100
5	3	100	1	100
5	4	50	1	60
6	5	100	1	100
10				
1	3	10		
1 1	3 4	10 10		
1	4	10		
1 2	4 1	10 10		
1 2 2	4 1 5	10 10 10		
1 2 2 3	4 1 5 6	10 10 10 10		
1 2 2 3 4	4 1 5 6 1	10 10 10 10 10		
1 2 2 3 4 4	4 1 5 6 1 5	10 10 10 10 10 10		
1 2 2 3 4 4 5	4 1 5 6 1 5	10 10 10 10 10 10		

Exemplo de saída:

Objetive:		950,00	
Lower bound	l:	950,00	
Gap:		0,0000	
Status:		Optimal	
Time:		0,03	
	1	2	100
	2	3	50
	2	4	100
	3	1	100
	4	6	100
	5	3	100
	6	5	100