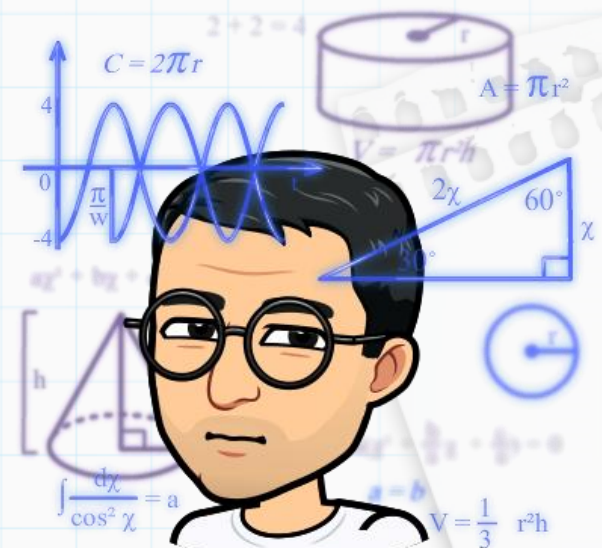
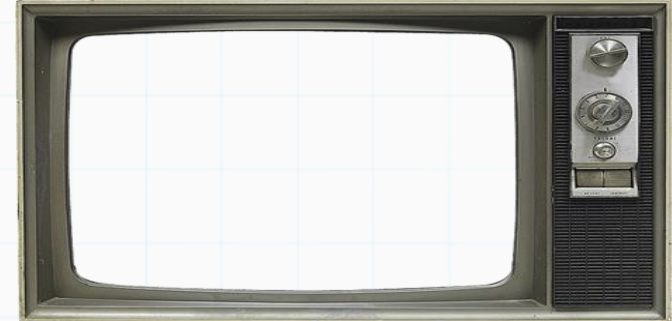


Programação Inteira

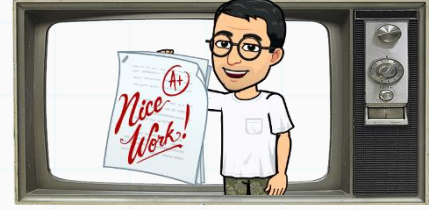
Professor : Yuri Frota

www.ic.uff.br/~yuri/pi.html

yuri@ic.uff.br



Trabalho

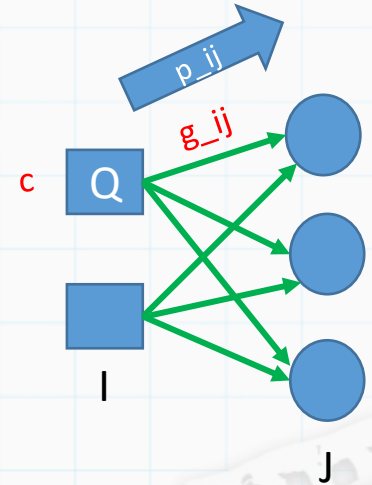


Seja J o conjunto de clientes que precisam ser atendidos, e I o conjunto de facilidades que podem atender os clientes (caso sejam construídas). Os conjuntos I e J definem um grafo bipartite não completo com conjunto de arestas E que definem que facilidades (se construídas) podem atender que clientes.

Para cada par $(i, j) \in E$ onde $i \in I$ e $j \in J$, denotamos por g_{ij} o custo de atendimento de i para j , e c o custo de construção da facilidade i (mesmo custo para a construção de todas as facilidades). Além disso, cada facilidade $i \in I$ possui uma quantidade máxima Q de recursos para atendimento, porém, se a facilidade i atender o cliente j , este atendimento irá consumir p_{ij} recursos. O Problema de localização de facilidades capacitado (PLFC) é definido como o problema de estabelecer que facilidades serão construídas, como será os atendimentos, de modo que todos os clientes sejam atendidos e o custo total seja mínimo.

Definimos então as variáveis binárias y_i , para toda facilidade $i \in I$ onde $y_i = 1$ se a facilidade é construída e $y_i = 0$, caso contrário. Além disso, definimos as variáveis binárias x_{ij} , para todo $i \in I$, $j \in J$ e $(i, j) \in E$, onde $x_{ij} = 1$ se i atende j , e $x_{ij} = 0$ caso contrário.

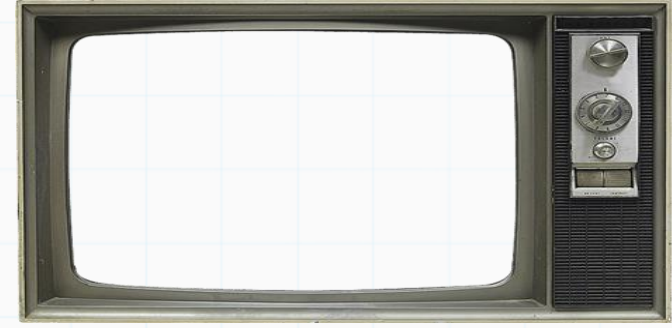
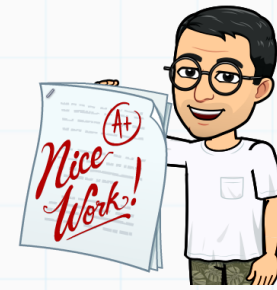
O modelo é definido a seguir:



Trabalho

$$\min \sum_{i \in I} cy_i + \sum_{(i,j) \in E} g_{ij} x_{ij}$$

(1)



$$\sum_{i \in I} x_{ij} = 1, \quad \forall j \in J$$

(2)

$$x_{ij} \leq y_i, \quad \forall (i, j) \in E$$

(3)

$$\sum_{(i,j) \in E} p_{ij} x_{ij} \leq Q y_i, \quad \forall i \in I$$

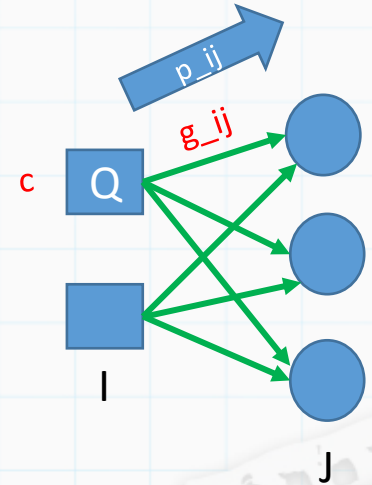
(4)

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in E$$

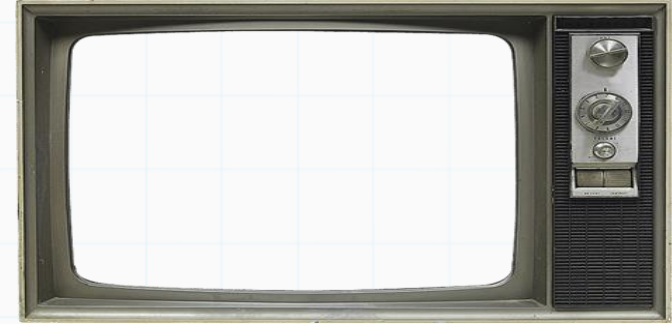
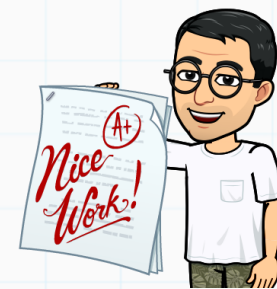
(5)

$$y_i \in \{0, 1\}, \quad \forall i \in I$$

(6)



Trabalho



- resolver as 3 instancias fornecidas:

TPI_F_0.txt (ótimo é 165)

TPI_F_1.txt

TPI_F_2.txt

Formato do arquivo:

ni (numero de facilidades) nj (numero de clientes) c (custo de abrir facilidade) Q (quantidade de recursos em cada facilidade)

NL (numero de linhas de facilidades/clientes a seguir)

facilidade(i) cliente(j) g_{ij} p_{ij}

EX:

2 3 50 5 5

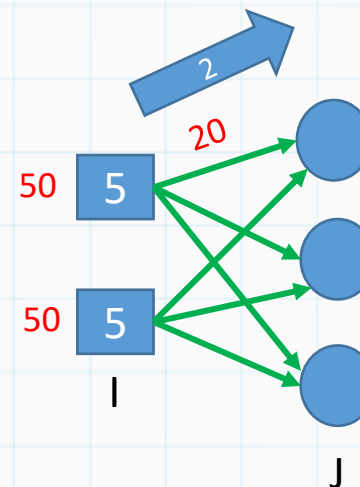
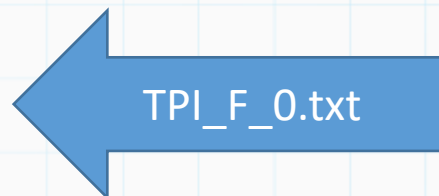
1 1 20 2

1 2 15 2

1 3 10 2

2 1 40 2

2 3 30 2



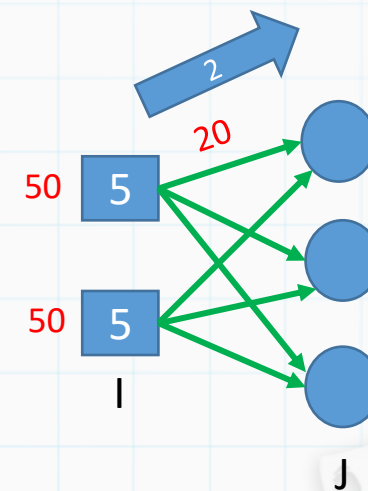
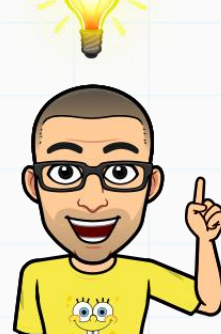
Trabalho

- Dica de Implementação:

- Use a instancia pequena para verificar a implementação
- A cada restrição implementada imprima o modelo e cheque a construção da restrição com a que você escreveu no papel, faça isso com as variáveis também

- Prazo:

- 2 semanas
- Atraso de 1 dia até 1 semana -> - 1 ponto
- Atraso de 1 semana e um dia até 2 semanas -> - 2 ponto
-



Até a próxima

