

INSTITUTO TECNOLÓGICO DE AERONÁUTICA



Disciplina: Introdução à Pesquisa Operacional

Professor: Antônio Augusto Chaves

Aluna: Ana Paula dos Santos Gularte

Resolução de Listas de Exercícios em aula

Exercício 1 - Problema



EXERCÍCIO 1. Em uma fazenda deseja-se fazer 10.000 Kg de ração com o menor custo possível. De acordo com as recomendações do veterinário dos animais da fazenda, a mesma deve conter 15% de proteína, um mínimo de 8% de fibra, e no mínimo 1.100 calorias por quilo de ração e no máximo 2.250 calorias por quilo. Para se fazer a ração, estão disponíveis quatro ingredientes cujas características técnico-econômicas estão mostrados na Tabela abaixo (dados são dados em %, exceto calorias e custo).

	Proteína	Fibra	Calorias/Kg	Custo/Kg
Cevada	6,9	6	1.760	30
Aveia	8,5	11	1.700	48
Soja	9	11	1.056	44
Milho	27,1	14	1.400	56

A ração deve ser feita contendo no mínimo 20% de milho e no máximo 12% de soja. Formule um modelo de Programação Linear para o problema.

Exercício 1 - Modelagem



VARIÁVEIS:

c: massa de cevada

a: massa de aveia

s: massa de soja

m: massa de milho

Obs.: Todas as massas estão em kg.

OBJETIVO: Minimizar o custo, dado por

$$30*c + 48*a + 44*s + 56*m$$

Exercício 1 - Modelagem



RESTRIÇÕES:

$$c + a + s + m = 10000 \text{ (Conservação de massa)}$$

$$6.9*c + 8.5*a + 9*s + 27.1*m = 15*10000 \text{ (15\% de proteína)}$$

$$6*c + 11*a + 11*s + 14*m \geq 8*10000 \text{ (Mínimo de 8\% de fibra)}$$

$$1760*c + 1700*a + 1056*s + 1400*m \geq 1100*10000 \text{ (Mínimo 1.100 calorias por quilo)}$$

$$1760*c + 1700*a + 1056*s + 1400*m \leq 2250*10000 \text{ (Máximo 2.250 calorias por quilo)}$$

$$m \geq 0.2*10000 \text{ (Mínimo 20\% de milho)}$$

$$s \leq 0.12*10000 \text{ (Máximo 12\% de soja)}$$

$$c, a, s, m \geq 0 \text{ (Massas não negativas)}$$

Exercício 1 - Solução Python (PuLP)



```
E1 = pulp.LpProblem("E1",pulp.LpMinimize)

## Declara Variáveis
c = pulp.LpVariable('c', lowBound=0, cat='Continuous')
a = pulp.LpVariable('a', lowBound=0, cat='Continuous')
s = pulp.LpVariable('s', lowBound=0, cat='Continuous')
m = pulp.LpVariable('m', lowBound=0, cat='Continuous')

## Função Objetivo
E1+= 30*c + 48*a + 44*s + 56*m

## Constraints
E1+= c + a + s + m == 10000
E1+= 6.9*c + 8.5*a + 9*s + 27.1*m == 15*10000
E1+= 6*c + 11*a + 11*s +14*m >= 8*10000
E1+= 1760*c + 1700*a + 1056*s + 1400*m >= 1100*10000
E1+= 1760*c + 1700*a + 1056*s + 1400*m <= 2250*10000
E1+= m >= 0.2*10000
E1+= s <= 0.12*10000

E1
```

```
## Solução
E1.solve()
pulp.LpStatus[E1.status]

'Optimal'
```

```
## Resultado das Variáveis
for variable in E1.variables():
    print ("{} = {}".format(variable.name, variable.varValue))

## Resultado da Função Objetivo
print (pulp.value(E1.objective))

a = 0.0
c = 5990.099
m = 4009.901
s = 0.0
404257.426
```


Exercício 2 - Problema



EXERCÍCIO 2. Uma empresa responsável pelo abastecimento semanal de um certo produto ao Rio de Janeiro e a São Paulo, pretende estabelecer um plano de distribuição do produto a partir dos centros produtores situados em Belo Horizonte, Ribeirão Preto e São José dos Campos. As quantidades semanalmente disponíveis em Belo Horizonte, Ribeirão Preto e São José dos Campos são 70, 130 e 120 toneladas respectivamente. O consumo semanal previsto deste produto é de 180 toneladas no Rio de Janeiro e 140 toneladas em São Paulo. Os custos de transporte, em \$/tonelada, de cada centro produtor para cada centro consumidor está dado abaixo:

	Rio de Janeiro	São Paulo
Belo Horizonte	13	25
Ribeirão Preto	25	16
São José dos Campos	15	40

Considerando que o objetivo da empresa é minimizar seu custo total de transporte, formule um modelo de Programação Linear para o problema.

Exercício 2 - Modelagem



ÍNDICES DAS CIDADES:

- 1 - Belo Horizonte
- 2 - Ribeirão Preto
- 3 - São José dos Campos
- 4 - Rio de Janeiro
- 5 - São Paulo

VARIÁVEIS:

X_{ij} : quantidade de toneladas que saem da cidade i e vão para cidade j

OBJETIVO: Minimizar o custo, dado por

$$13 * X_{14} + 25 * X_{15} + 25 * X_{24} + 16 * X_{25} + 15 * X_{34} + 40 * X_{35}$$

Exercício 2 - Modelagem



RESTRIÇÕES:

$X_{14} + X_{15} \leq 70$ (Quantidade disponível semanalmente em Belo Horizonte)

$X_{24} + X_{25} \leq 130$ (Quantidade disponível semanalmente em Ribeirão Preto)

$X_{34} + X_{35} \leq 120$ (Quantidade disponível semanalmente em São José dos Campos)

$X_{14} + X_{24} + X_{34} = 180$ (Consumo semanal do Rio de Janeiro)

$X_{15} + X_{25} + X_{35} = 140$ (Consumo semanal de São Paulo)

$X_{14}, X_{15}, X_{24}, X_{25}, X_{34}, X_{35} \geq 0$ (Massas não negativas)

Exercício 2 - Solução Python (PuLP)



```
E2 = pulp.LpProblem("E2",pulp.LpMinimize)

## Declara Variáveis
X14 = pulp.LpVariable('X14', lowBound=0, cat='Continuous')
X15 = pulp.LpVariable('X15', lowBound=0, cat='Continuous')
X24 = pulp.LpVariable('X24', lowBound=0, cat='Continuous')
X25 = pulp.LpVariable('X25', lowBound=0, cat='Continuous')
X34 = pulp.LpVariable('X34', lowBound=0, cat='Continuous')
X35 = pulp.LpVariable('X35', lowBound=0, cat='Continuous')

## Função Objetivo
E2+= 13*X14 + 25*X15 + 25*X24 + 16*X25 + 15*X34 + 40*X35

## Constraints
E2+= X14 + X15 <= 70
E2+= X24 + X25 <= 130
E2+= X34 + X35 <= 120
E2+= X14 + X24 + X34 == 180
E2+= X15 + X25 + X35 == 140

E2
```

```
## Solução
E2.solve()
pulp.LpStatus[E2.status]
```

'Optimal'

```
## Resultado das Variáveis
for variable in E2.variables():
    print ("{} = {}".format(variable.name, variable.varValue))
```

```
## Resultado da Função Objetivo
print (pulp.value(E2.objective))
```

```
X14 = 60.0
X15 = 10.0
X24 = 0.0
X25 = 130.0
X34 = 120.0
X35 = 0.0
4910.0
```

Exercício 3 - Problema



EXERCÍCIO 3. Na produção de unidades de quatro tipos de produtos, são utilizadas duas máquinas. O tempo utilizado na fabricação de cada unidade, de cada tipo de produto, em cada uma das duas máquinas, está dado da tabela abaixo.

Máquina	Produto 1	Produto 2	Produto 3	Produto 4
1	2	3	4	2
2	3	2	1	2

O custo total de produção de uma unidade de cada produto é diretamente proporcional ao tempo de uso da máquina. Considere que o custo por hora para as máquinas 1 e 2 são \$10 e \$15 respectivamente. O total de horas disponíveis para todos os produtos nas máquinas 1 e 2 são 500 e 380 respectivamente. Se o preço de venda, por unidade, dos produtos 1, 2, 3 e 4 é de \$65, \$70, \$55 e \$45, formule o problema como um modelo de Programação Linear com o objetivo de maximizar o lucro líquido total.

Exercício 3 - Modelagem



VARIÁVEIS:

X_{ij} : quantidade de produtos do tipo j que são produzidos pela máquina do tipo i

OBJETIVO: Maximizar o lucro, que é a receita menos o custo, dado por

SOMA em i, j de $(P_j - T_{ij} * C_i) * X_{ij}$, sendo

P_j - preço da unidade do produto j

T_{ij} - tempo que a máquina i leva pra produzir o produto j

C_i - custo da hora da máquina i

$$\begin{aligned} & (65 - 2 * 10 - 3 * 15) * X_1 + (70 - 3 * 10 - 2 * 15) * X_2 + (55 - 4 * 10 - 1 * 15) * X_3 + (45 - 2 * 10 - 2 * 15) * X_4 \\ & = \\ & 10X_2 - 5X_4 \end{aligned}$$

Exercício 3 - Modelagem



RESTRIÇÕES:

$2 \cdot X1 + 3 \cdot X2 + 4 \cdot X3 + 2 \cdot X4 \leq 500$ (Tempo total disponível para máquina 1)

$3 \cdot X1 + 2 \cdot X2 + 1 \cdot X3 + 2 \cdot X4 \leq 380$ (Tempo total disponível para máquina 2)

$X1, X2, X3, X4 \geq 0$ e inteiras

Exercício 3 - Solução Python (PuLP)



```
E3 = pulp.LpProblem("E3",pulp.LpMaximize)

## Declara Variáveis
X1 = pulp.LpVariable('X1', lowBound=0, cat='Integer')
X2 = pulp.LpVariable('X2', lowBound=0, cat='Integer')
X3 = pulp.LpVariable('X3', lowBound=0, cat='Integer')
X4 = pulp.LpVariable('X4', lowBound=0, cat='Integer')

## Função Objetivo
E3+=(65-2*10-3*15)*X1 + (70-3*10-2*15)*X2 + (55-4*10-1*15)*X3 + (45-2*10-2*15)*X4

## Constraints
E3+= 2*X1 + 3*X2 + 4*X3 + 2*X4 <= 500
E3+= 3*X1 + 2*X2 + 1*X3 + 2*X4 <= 380

E3
```

```
1 ## Solução
2 E3.solve()
3 pulp.LpStatus[E3.status]
```

'Optimal'

```
1 ## Resultado das Variáveis
2 for variable in E3.variables():
3     print ("{} = {}".format(variable.name, variable.varValue))
4
5 ## Resultado da Função Objetivo
6 print (pulp.value(E3.objective))
```

```
X1 = 0.0
X2 = 166.0
X3 = 0.0
X4 = 0.0
1660.0
```


Exercício 4 - Problema



EXERCÍCIO 4. Uma companhia de aviação está considerando a compra de aviões de passageiros de 3 tipos: de pequeno curso, de curso médio e de longo curso. O preço de compra seria de $\$6,7M$ para cada avião de longo curso, $\$5M$ para aviões de médio curso e $\$3,5M$ para aviões de pequeno curso. A diretoria autorizou um gasto máximo de $\$150M$ para estas compras, independentemente de quais aviões serão comprados. As viagens aéreas em todos os tipos de aviões, fazem prever que os aviões andarão sempre lotados. Estima-se que o lucro anual líquido seria de $\$0,42M$ para cada avião de longo curso, $\$0,30M$ para avião de médio curso e $\$0,23M$ para avião de pequeno curso. A companhia terá pilotos treinados para pilotar 30 novos aviões. Se somente aviões de pequeno curso forem comprados, a divisão de manutenção estaria apta a manter 40 novos aviões. Cada avião de médio curso gasta $1/3$ a mais de manutenção do que o dispendido por um avião de pequeno curso e o de longo curso $2/3$ a mais. As informações acima foram obtidas por uma análise preliminar do problema. Uma análise mais detalhada será feita posteriormente. No entanto, usando os dados acima como uma primeira aproximação, a diretoria da empresa deseja conhecer quantos aviões de cada tipo deveriam ser comprados se o objetivo é maximizar o lucro. Formule um modelo de Programação Linear para este problema. ($M = 1.000.000$)

Exercício 4 - Modelagem



VARIÁVEIS:

L - Quantidade de aviões de longo curso

M - Quantidade de aviões de médio curso

P - Quantidade de aviões de pequeno curso

OBJETIVO: Maximizar o lucro, dado por

$$0.42*L + 0.30*M + 0.23*P$$

RESTRIÇÕES:

Se X é o dinheiro disponível para manutenção, então as manutenções dos aviões de pequeno, médio e longo cursam custam, respectivamente, $X/40$, $X/30$ e $X/24$.

$$L*X/24 + M*X/30 + P*X/40 \leq X \rightarrow (1/24)*L + (1/30)*M + (1/40)*P \leq 1 \text{ (Manutenção)}$$

$$L + M + P \leq 30 \text{ (Quantidade de pilotos)}$$

$$L, M, P \geq \text{e inteiros}$$

Exercício 4 - Solução Python (PuLP)



```
E4 = pulp.LpProblem("E4",pulp.LpMaximize)

## Declara Variáveis
L = pulp.LpVariable('L', lowBound=0, cat='Integer')
M = pulp.LpVariable('M', lowBound=0, cat='Integer')
P = pulp.LpVariable('P', lowBound=0, cat='Integer')

## Função Objetivo
E4+= 0.42*L + 0.30*M + 0.23*P

## Constraints
E4+= (1/24)*L + (1/30)*M + (1/40)*P <= 1
E4+= L + M + P <= 30
E4+= 6.7*L + 5*M + 3.5*P <= 150

E4
```

```
## Solução
E4.solve()
pulp.LpStatus[E4.status]
```

'Optimal'

```
## Resultado das Variáveis
for variable in E4.variables():
    print ("{} = {}".format(variable.name, variable.varValue))
```

```
## Resultado da Função Objetivo
print (pulp.value(E4.objective))
```

```
L = 14.0
M = 0.0
P = 16.0
9.56
```


Exercício 5 - Problema



EXERCÍCIO 5. Uma empresa tem 3 fábricas com ociosidade na produção. Todas as 3 fábricas tem capacidade de produzir um certo produto e a gerência decidiu usar uma parte da ociosidade na produção deste produto. O produto pode ser feito em 3 tamanhos: grande, médio e pequeno, que dão um lucro líquido de \$12, \$10 e \$9 respectivamente. As fábricas 1, 2 e 3 tem capacidade de fabricar 500, 600 e 300 unidades do produto respectivamente, independentemente do tamanho a ser produzido. Há, no entanto, limitação do espaço para estocagem. As fábricas 1, 2 e 3 tem 9000, 8000 e 3500 m^2 de área para estocagem respectivamente. Cada unidade de tamanho grande, médio e pequeno necessita de 20, 15 e 12 m^2 respectivamente. O Departamento de Vendas indicou que 600, 800 e 500 unidades dos tamanhos grande, médio e pequeno, respectivamente, podem ser vendidas por dia. De maneira a manter uma certa uniformidade, a gerência decidiu que a percentagem do uso das capacidades ociosas das 3 fábricas devem ser iguais. A gerência deseja saber quanto de cada tamanho deve ser produzido em cada fábrica de maneira que o lucro seja máximo. Formule um modelo de Programação Linear para este problema.

Exercício 5 - Modelagem



O QUE QUEREMOS? Quanto de cada tamanho deve ser produzido em cada fábrica de maneira que o lucro seja máximo.

VARIÁVEIS:

X_{ij} : quantidade a ser produzida do produto no tamanho j , (j = Grande, Médio, Pequeno) na fábrica i , (i = 1, 2, 3)

GF1 = Produto Grande na Fábrica 1

GF2 = Produto Grande na Fábrica 2

GF3 = Produto Grande na Fábrica 3

MF1 = Produto Médio na Fábrica 1

MF2 = Produto Médio na Fábrica 2

MF3 = Produto Médio na Fábrica 3

PF1 = Produto Pequeno na Fábrica 1

PF2 = Produto Pequeno na Fábrica 2

PF3 = Produto Pequeno na Fábrica 3

Exercício 5 - Modelagem



FUNÇÃO OBJETIVO: Maximizar o lucro, dado por

$$12*(GF1+GF2+GF3) + 10*(MF1+MF2+MF3) + 9*(PF1+PF2+PF3)$$

RESTRIÇÕES:

Estocagem

$$20*GF1 + 15*MF1 + 12*PF1 \leq 9000m^2$$

$$20*GF2 + 15*MF2 + 12*PF2 \leq 8000m^2$$

$$20*GF3 + 15*MF3 + 12*PF3 \leq 3500m^2$$

Demanda

$$GF1 + GF2 + GF3 \leq 600 \text{ unidades}$$

$$MF1 + MF2 + MF3 \leq 800 \text{ unidades}$$

$$PF1 + PF2 + PF3 \leq 500 \text{ unidades}$$

Porcentagem iguais das Capacidades Ociosas

$$(GF1 + MF1 + PF1)/500 == (GF2 + MF2 + PF2)/600$$

$$(GF1 + MF1 + PF1)/500 == (GF3 + MF3 + PF3)/300$$

$$(GF2 + MF2 + PF2)/600 == (GF3 + MF3 + PF3)/300$$

Não negatividade

$GF1, GF2, GF3, MF1, MF2, MF3, PF1, PF2, PF3 \geq 0$ e inteiros

Exercício 5 - Solução Excel (OpenSolver)



	G	M	P	TOTAL				
FAB1	363	112	5	480	<=	500	96,00%	96,0000%
FAB2	1	360	215	576	<=	600	96,00%	96,0000%
FAB3	4	4	280	288	<=	300	96,00%	96,0000%
TOTAL	368	476	500					
					>=	0		
FO MAX	R\$12,00	R\$10,00	R\$ 9,00	max R\$ 13.676,00				

Restrito a:

ESTOCAGEM	FAB1	20	15	12	9000	<=	9000	m^2
	FAB2	20	15	12	8000	<=	8000	
	FAB3	20	15	12	3500	<=	3500	
DEMANDA	G	1	0	0	368	<=	600	unidades
	M	0	1	0	476	<=	800	
	P	0	0	1	500	<=	500	

Exercício 5 - Solução Python (PuLP)



```
E5 = pulp.LpProblem("E5",pulp.LpMaximize)

## Declara Variáveis
GF1 = pulp.LpVariable('GF1', lowBound=0, cat='Integer')
GF2 = pulp.LpVariable('GF2', lowBound=0, cat='Integer')
GF3 = pulp.LpVariable('GF3', lowBound=0, cat='Integer')
MF1 = pulp.LpVariable('MF1', lowBound=0, cat='Integer')
MF2 = pulp.LpVariable('MF2', lowBound=0, cat='Integer')
MF3 = pulp.LpVariable('MF3', lowBound=0, cat='Integer')
PF1 = pulp.LpVariable('PF1', lowBound=0, cat='Integer')
PF2 = pulp.LpVariable('PF2', lowBound=0, cat='Integer')
PF3 = pulp.LpVariable('PF3', lowBound=0, cat='Integer')

## Função Objetivo
E5+= 12*(GF1+GF2+GF3) + 10*(MF1+MF2+MF3) + 9*(PF1+PF2+PF3)

## Constraints

# Estocagem
E5+= 20*GF1 + 15*MF1 + 12*PF1 <= 9000
E5+= 20*GF2 + 15*MF2 + 12*PF2 <= 8000
E5+= 20*GF3 + 15*MF3 + 12*PF3 <= 3500

# Demanda
E5+= GF1 + GF2 + GF3 <= 600
E5+= MF1 + MF2 + MF3 <= 800
E5+= PF1 + PF2 + PF3 <= 500

# Percentagem das Capacidades Ociosas
E5+= (GF1 + MF1 + PF1)/500 == (GF2 + MF2 + PF2)/600
E5+= (GF1 + MF1 + PF1)/500 == (GF3 + MF3 + PF3)/300
E5+= (GF2 + MF2 + PF2)/600 == (GF3 + MF3 + PF3)/300
```

E5

```
## Solução
E5.solve()
pulp.LpStatus[E5.status]
```

'Optimal'

```
## Resultado das Variáveis
for variable in E5.variables():
    print ("{} = {}".format(variable.name, variable.varValue))
```

```
## Resultado da Função Objetivo
print (pulp.value(E5.objective))
```

```
GF1 = 363.0
GF2 = 1.0
GF3 = 4.0
MF1 = 112.0
MF2 = 360.0
MF3 = 4.0
PF1 = 5.0
PF2 = 215.0
PF3 = 280.0
13676.0
```

Exercício 5 - Solução para o Cliente



Fabricação de produtos com tamanhos variados utilizando a capacidade ociosa das fábricas de maneira que o lucro seja máximo, conforme tabela abaixo:

	Produto Grande	Produto Médio	Produto Pequeno
Fábrica 1	363	112	5
Fábrica 2	1	360	215
Fábrica 3	4	4	280

*Quantidade de produtos fabricados.

Além disso, a fabricação dos produtos atende a área para estocagem de cada fábrica de **9.000m²** da fábrica1, **8.000m²** da fábrica2 e **3.500m²** da fábrica3.

A demanda diária prevista pela área de vendas foi atendida em **61%** para o **tamanho de produto G**, **60%** para o **tamanho M** e **100%** para o **tamanho P**.

Também foi atendida a uniformidade atendendo a solicitação da gerência em manter a igualdade da percentagem do uso das capacidades ociosas das fábricas, ficando em: **96% de ocupação de cada uma das fábricas**.

Lucro total da produção diária: \$ 13.676,00.

Exercício 6 - Problema



EXERCÍCIO 6. Um investidor pode investir dinheiro em duas atividades A e B disponíveis no início dos próximos 5 anos. Cada \$1 investido em A no começo de um ano retorna \$1,40 (um lucro de \$0,40) dois anos mais tarde (a tempo de imediato reinvestimento). Cada \$1 investido em B no início de um ano retorna \$1,70, três anos mais tarde. Existem ainda 2 atividades C e D que estarão disponíveis no futuro. Cada \$1 investido em C no início do segundo ano retorna \$2,00, quatro anos mais tarde. Cada \$1 investido em D no começo do quinto ano, retorna \$1,30 um ano mais tarde. O investidor tem \$10.000. Ele deseja conhecer como investir de maneira a maximizar a quantidade de dinheiro acumulado no início do sexto ano.

Formule um modelo de Programação Linear para este problema. Considere que não há inflação.

Exercício 6 - Modelagem



VARIÁVEIS

SI = Saldo Inicial

SI_to_A1 = Saldo Inicial para A no ano 1

SI_to_A2 = Saldo Inicial para A no ano 2

SI_to_A3 = Saldo Inicial para A no ano 3

SI_to_A4 = Saldo Inicial para A no ano 4

SI_to_B1 = Saldo Inicial para B no ano 1

SI_to_B2 = Saldo Inicial para B no ano 2

SI_to_B3 = Saldo Inicial para B no ano 3

SI_to_C2 = Saldo Inicial para C no ano 2

SI_to_D5 = Saldo Inicial para D no ano 5

Exercício 6 - Modelagem



VARIÁVEIS

A1_to_A3 = De A no ano 1 para A no ano 3

A2_to_A4 = De A no ano 2 para A no ano 4

A3_to_A5 = De A no ano 3 para A no ano 5

A4_to_SF = De A no ano 4 para o Saldo Final

A5_to_D5 = De A no ano 5 para D no ano 5

A5_to_SF = De A no ano 5 para o Saldo Final

Exercício 6 - Modelagem



VARIÁVEIS

B1_to_B4 = De B no ano 1 para B no ano 4

B2_to_B5 = De B no ano 2 para B no ano 5

B3_to_SF = De B no ano 3 para o Saldo Final

B4_to_SF = De B no ano 4 para o Saldo Final

B5_to_D5 = De B no ano 5 para D no ano 5

B5_to_SF = De B no ano 5 para o Saldo Final

C2_to_SF = De C no ano 2 para o Saldo Final

D5_to_SF = De D no ano 5 para o Saldo Final

SF = Saldo Final

FUNÇÃO OBJETIVO: Maximizar SF

Exercício 6 - Modelagem



RESTRIÇÕES

$$SI == 10000$$

#NÓ SALDO INICIAL

$$SI == SI_to_A1 + SI_to_A2 + SI_to_A3 + SI_to_A4 + \\ SI_to_B1 + SI_to_B2 + SI_to_B3 + SI_to_C2 + \\ SI_to_D5$$

Exercício 6 - Modelagem



RESTRIÇÕES

#NÓ A1

$$SI_to_A1 == A1_to_A3$$

#NÓ A2

$$SI_to_A2 == A2_to_A4$$

#NÓ A3

$$SI_to_A3 + 1.4 * A1_to_A3 == A3_to_A5$$

#NÓ A4

$$SI_to_A4 + 1.4 * A2_to_A4 == A4_to_SF$$

#NÓ A5

$$1.4 * A3_to_A5 == A5_to_D5 + A5_to_SF$$

Exercício 6 - Modelagem



RESTRIÇÕES

#NÓ B1

$$SI_to_B1 == B1_to_B4$$

#NÓ B2

$$SI_to_B2 == B2_to_B5$$

#NÓ B3

$$SI_to_B3 == B3_to_SF$$

#NÓ B4

$$1.7 * B1_to_B4 == B4_to_SF$$

#NÓ B5

$$1.7 * B2_to_B5 == B5_to_D5 + B5_to_SF$$

Exercício 6 - Modelagem



RESTRIÇÕES

#NÓ C2

$$SI_to_C2 == C2_to_SF$$

#NÓ D5

$$SI_to_D5 + A5_to_D5 + B5_to_D5 == D5_to_SF$$

#NÓ SALDO FINAL

$$A5_to_SF + 1.4 * A4_to_SF + 2 * C2_to_SF + \\ 1.3 * D5_to_SF + 1.7 * B3_to_SF + B4_to_SF + \\ B5_to_SF == SF$$

Exercício 6 - Solução Python (PuLP)



```
E6 = pulp.LpProblem("E6",pulp.LpMaximize)

## Declara Variáveis
SI = pulp.LpVariable('SI', lowBound=0, cat='Continuous')
SI_to_A1 = pulp.LpVariable('SI_to_A1', lowBound=0, cat='Continuous')
SI_to_A2 = pulp.LpVariable('SI_to_A2', lowBound=0, cat='Continuous')
SI_to_A3 = pulp.LpVariable('SI_to_A3', lowBound=0, cat='Continuous')
SI_to_A4 = pulp.LpVariable('SI_to_A4', lowBound=0, cat='Continuous')
SI_to_B1 = pulp.LpVariable('SI_to_B1', lowBound=0, cat='Continuous')
SI_to_B2 = pulp.LpVariable('SI_to_B2', lowBound=0, cat='Continuous')
SI_to_B3 = pulp.LpVariable('SI_to_B3', lowBound=0, cat='Continuous')
SI_to_C2 = pulp.LpVariable('SI_to_C2', lowBound=0, cat='Continuous')
SI_to_D5 = pulp.LpVariable('SI_to_D5', lowBound=0, cat='Continuous')
A1_to_A3 = pulp.LpVariable('A1_to_A3', lowBound=0, cat='Continuous')
A2_to_A4 = pulp.LpVariable('A2_to_A4', lowBound=0, cat='Continuous')
A3_to_A5 = pulp.LpVariable('A3_to_A5', lowBound=0, cat='Continuous')
A4_to_SF = pulp.LpVariable('A4_to_SF', lowBound=0, cat='Continuous')
A5_to_D5 = pulp.LpVariable('A5_to_D5', lowBound=0, cat='Continuous')
A5_to_SF = pulp.LpVariable('A5_to_SF', lowBound=0, cat='Continuous')
B1_to_B4 = pulp.LpVariable('B1_to_B4', lowBound=0, cat='Continuous')
B2_to_B5 = pulp.LpVariable('B2_to_B5', lowBound=0, cat='Continuous')
B3_to_SF = pulp.LpVariable('B3_to_SF', lowBound=0, cat='Continuous')
B4_to_SF = pulp.LpVariable('B4_to_SF', lowBound=0, cat='Continuous')
B5_to_D5 = pulp.LpVariable('B5_to_D5', lowBound=0, cat='Continuous')
B5_to_SF = pulp.LpVariable('B5_to_SF', lowBound=0, cat='Continuous')
C2_to_SF = pulp.LpVariable('C2_to_SF', lowBound=0, cat='Continuous')
D5_to_SF = pulp.LpVariable('D5_to_SF', lowBound=0, cat='Continuous')
SF = pulp.LpVariable('SF', lowBound=0, cat='Continuous')

## Função Objetivo
E6+= SF
```

Exercício 6 - Solução Python (PuLP)



```
## Constraints
```

```
E6+= SI == 10000
```

```
#NÓ SALDO INICIAL
```

```
E6+= SI == SI_to_A1 + SI_to_A2 + SI_to_A3 + SI_to_A4 + SI_to_B1 + SI_to_B2 + SI_to_B3 + SI_to_C2 + SI_to_D5
```

```
E6+= SI_to_A1 == A1_to_A3 #NÓ A1
```

```
E6+= SI_to_A2 == A2_to_A4 #NÓ A2
```

```
E6+= SI_to_A3 + 1.4*A1_to_A3 == A3_to_A5 #NÓ A3
```

```
E6+= SI_to_A4 + 1.4*A2_to_A4 == A4_to_SF #NÓ A4
```

```
E6+= 1.4*A3_to_A5 == A5_to_D5 + A5_to_SF #NÓ A5
```

```
E6+= SI_to_B1 == B1_to_B4 #NÓ B1
```

```
E6+= SI_to_B2 == B2_to_B5 #NÓ B2
```

```
E6+= SI_to_B3 == B3_to_SF #NÓ B3
```

```
E6+= 1.7*B1_to_B4 == B4_to_SF #NÓ B4
```

```
E6+= 1.7*B2_to_B5 == B5_to_D5 + B5_to_SF #NÓ B5
```

```
E6+= SI_to_C2 == C2_to_SF #NÓ C2
```

```
E6+= SI_to_D5 + A5_to_D5 + B5_to_D5 == D5_to_SF #NÓ D5
```

```
#NÓ SALDO FINAL
```

```
E6+= A5_to_SF + 1.4*A4_to_SF + 2*C2_to_SF + 1.3*D5_to_SF + 1.7*B3_to_SF + B4_to_SF + B5_to_SF == SF
```

```
## Solução
```

```
E6.solve()
```

```
pulp.LpStatus[E6.status]
```

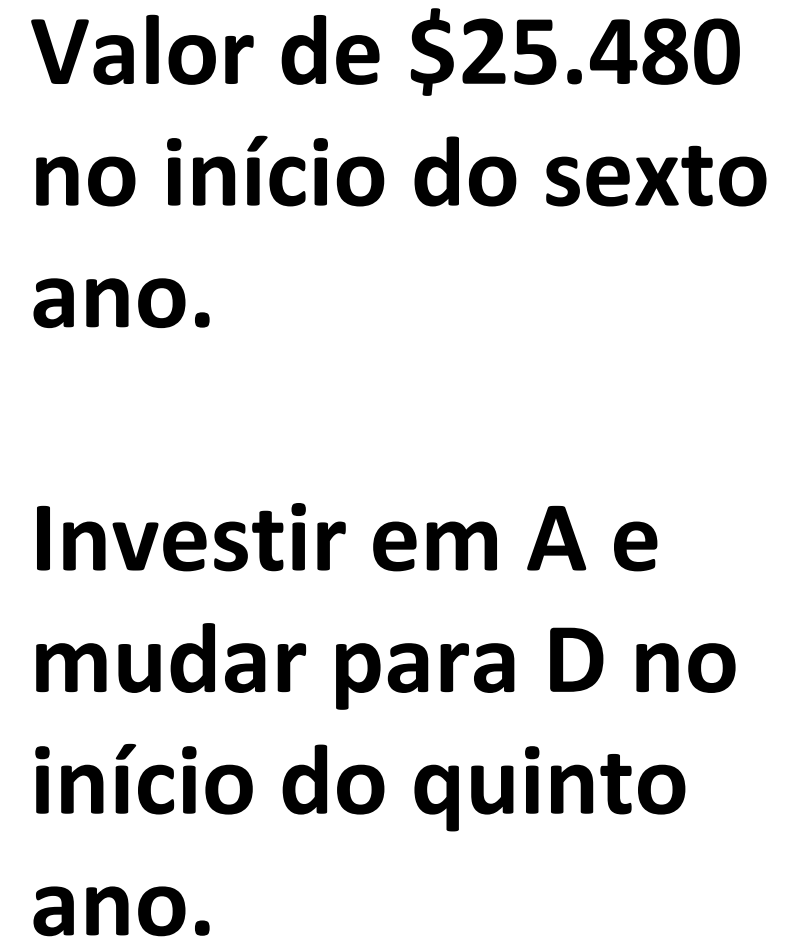
```
'Optimal'
```


Exercício 6 - Solução Python (PuLP)



```
## Resultado das Variáveis  
for variable in E6.variables():  
    print("{} = {}".format(variable.name, variable.varValue))  
  
## Resultado da Função Objetivo  
print (pulp.value(E6.objective))
```

```
A1_to_A3 = 10000.0  
A2_to_A4 = 0.0  
A3_to_A5 = 14000.0  
A4_to_SF = -0.0  
A5_to_D5 = 19600.0  
A5_to_SF = 0.0  
B1_to_B4 = 0.0  
B2_to_B5 = 0.0  
B3_to_SF = 0.0  
B4_to_SF = 0.0  
B5_to_D5 = 0.0  
B5_to_SF = 0.0  
C2_to_SF = 0.0  
D5_to_SF = 19600.0  
SF = 25480.0  
SI = 10000.0  
SI_to_A1 = 10000.0  
SI_to_A2 = 0.0  
SI_to_A3 = 0.0  
SI_to_A4 = 0.0  
SI_to_B1 = 0.0  
SI_to_B2 = 0.0  
SI_to_B3 = 0.0  
SI_to_C2 = 0.0  
SI_to_D5 = 0.0  
25480.0
```



**Investir em A e
mudar para D no
início do quinto
ano.**

Exercício 7 - Problema



EXERCÍCIO 7. Com seus conhecimentos do curso, um aluno calcula que poderia se preparar com perfeição para o exame de uma certa disciplina D_1 em 20 horas de estudo intensivo. Para uma outra disciplina D_2 ele precisa de 25 horas. Para passar, ele precisa obter no mínimo 50 pontos (num máximo de 100) em cada uma delas. Além disso, ele deseja alcançar a maior média ponderada possível, sendo 3 e 5 os pesos de D_1 e D_2 respectivamente. Ele dispõe de apenas 30 horas para estudar.

Formule o problema como um modelo de Programação Linear, a fim de obter a distribuição das horas de estudo, considerando proporcionalidade entre o esforço e o rendimento de seus estudos.

Exercício 7 - Modelagem



O QUE QUEREMOS? Qual a distribuição das horas de estudo em cada disciplina é necessário levando em consideração o esforço e o rendimento dos estudos.

VARIÁVEIS:

D1 = horas de estudos dedicados a disciplina 1

D2 = horas de estudos dedicados a disciplina 2

OBJETIVO: Maximizar o tempo, dado por

$$15*D1 + 20*D2/8$$

RESTRIÇÕES:

Quantidade total de horas disponíveis para estudo: $D1 + D2 = 30$

Mínimo 50 pontos em cada disciplina para passar: $5*D1 \geq 50$; $4*D2 \geq 5$

D1, D2 ≥ 0 e inteiras

Exercício 7 - Solução Python (PuLP)



```
1 E7 = pulp.LpProblem("E7",pulp.LpMaximize)
2
3 ## Declara Variáveis
4 D1 = pulp.LpVariable('D1', lowBound=0, cat='Integer')
5 D2 = pulp.LpVariable('D2', lowBound=0, cat='Integer')
6
7
8 ## Função Objetivo
9 E7+= (15*D1 + 20*D2)/8
10
11 ## Constraints
12 E7+= D1 + D2 == 30
13 E7+= 5*D1 >= 50
14 E7+= 4*D2 >= 50
15
16 E7
```

```
1 ## Solução
2 E7.solve()
3 pulp.LpStatus[E7.status]
```

'Optimal'

```
1 ## Resultado das Variáveis
2 for variable in E7.variables():
3     print ("{} = {}".format(variable.name, variable.varValue))
4
5 ## Resultado da Função Objetivo
6 print (pulp.value(E7.objective))
```

```
D1 = 10.0
D2 = 20.0
68.75
```

SOLUÇÃO PARA O CLIENTE CONFORME ABAIXO:

Montar para um aluno a distribuição das suas horas de estudo em duas disciplinas, levando em consideração a disponibilidade de tempo de 30 horas, os pesos e a quantidade mínima de 50 pontos para passar em cada uma nas disciplinas com a maior média ponderada.

Total da média ponderada: **68.75**, com a distribuição de 10 horas para a disciplina 1 e 20 horas para a disciplina 2.

Exercício 8 - Problema



EXERCÍCIO 8. Uma companhia deseja obter uma nova liga metálica com 30% de chumbo, 20% de zinco e 50% de estanho a partir de alguns minérios tendo as seguintes propriedades:

	Minérios				
Propriedades	1	2	3	4	5
Chumbo (%)	30	10	50	10	50
Zinco (%)	60	20	20	10	10
Estanho (%)	10	70	30	80	40
Custo (\$Kg)	8,5	6	8,9	5,7	8,8

O objetivo é determinar as proporções destes minérios que deveriam ser misturados para produzir a nova liga com o menor custo possível. Formule este problema como um modelo de Programação Linear.

Exercício 8 - Modelagem



O QUE QUEREMOS? Qual é a proporção de cada minério que deveriam ser misturados para produzir a nova liga com o menor custo possível.

VARIÁVEIS:

X_j = Proporção de 1Kg dos minérios j que devem ser misturados, $j = 1, 2, 3, 4, 5$.

M1 = 1Kg de Minério que deve ser utilizado em uma unidade de mistura do tipo 1

M2 = 1Kg de Minério que deve ser utilizado em uma unidade de mistura do tipo 2

M3 = 1Kg de Minério que deve ser utilizado em uma unidade de mistura do tipo 3

M4 = 1Kg de Minério que deve ser utilizado em uma unidade de mistura do tipo 4

M5 = 1Kg de Minério que deve ser utilizado em uma unidade de mistura do tipo 5

Exercício 8 - Modelagem



FUNÇÃO OBJETIVO: Minimizar o custo, dado por

$$8,5*M1 + 6*M2 + 8,9*M3 + 5,7*M4 + 8,8*M5$$

RESTRIÇÕES:

Composição (Pelo menos 30% de Chumbo, 20% de Zinco e 50% de Estanho)

$$0,3*M1 + 0,1*M2 + 0,5*M3 + 0,1*M4 + 0,5*M5 = 0,3$$

$$0,6*M1 + 0,2*M2 + 0,2*M3 + 0,1*M4 + 0,1*M5 = 0,2$$

$$0,1*M1 + 0,7*M2 + 0,3*M3 + 0,8*M4 + 0,4*M5 = 0,5$$

Soma dos minérios resulta em 1Kg de mistura

$$M1 + M2 + M3 + M4 + M5 = 1$$

Os minérios podem ser utilizados ou não

$$M1 \geq 0, M2 \geq 0, M3 \geq 0, M4 \geq 0, M5 \geq 0$$

Não negatividade

$$M1, M2, M3, M4, M5 \geq 0$$

Exercício 8 - Solução Excel (OpenSolver)



PROPRIEDADES	MINÉRIOS				
	1	2	3	4	5
Proporções da nova liga	0,11	0,00	0,44	0,44	0,00

FO MIN Custo R\$ 8,60 R\$ 6,00 R\$ 8,90 R\$ 5,70 R\$ 8,80 min R\$ 7,44

30% Chumbo	30%	10%	50%	10%	50%	30%	=	30%
20% Zinco	60%	20%	20%	10%	10%	20%	-	20%
50% Estanho	10%	70%	30%	80%	40%	50%	=	50%

Exercício 8 - Solução Python (PuLP)



```
E8 = pulp.LpProblem("E8",pulp.LpMinimize)

## Declara Variáveis
M1 = pulp.LpVariable('M1', lowBound=0, cat='Continuous')
M2 = pulp.LpVariable('M2', lowBound=0, cat='Continuous')
M3 = pulp.LpVariable('M3', lowBound=0, cat='Continuous')
M4 = pulp.LpVariable('M4', lowBound=0, cat='Continuous')
M5 = pulp.LpVariable('M5', lowBound=0, cat='Continuous')

## Função Objetivo
E8+= 8.5*M1 + 6*M2 + 8.9*M3 + 5.7*M4 + 8.8*M5

## Constraints

# Nova liga a partir dos minérios existentes
E8+= M1 + M2 + M3 + M4 + M5 == 1

# Proporções de minérios (30% Chumbo, 20% Zinco e 50% Estanho)
E8+= 0.3*M1 + 0.1*M2 + 0.5*M3 + 0.1*M4 + 0.5*M5 == 0.3
E8+= 0.6*M1 + 0.2*M2 + 0.2*M3 + 0.1*M4 + 0.1*M5 == 0.2
E8+= 0.1*M1 + 0.7*M2 + 0.3*M3 + 0.8*M4 + 0.4*M5 == 0.5

# Os minérios podem ser utilizados ou não
E8+= M1 >= 0
E8+= M2 >= 0
E8+= M3 >= 0
E8+= M4 >= 0
E8+= M5 >= 0

E8
```

```
1 ## Solução
2 E8.solve()
3 pulp.LpStatus[E8.status]
```

'Optimal'

```
1 ## Resultado das Variáveis
2 for variable in E8.variables():
3     print("{} = {}".format(variable.name, variable.varValue))
4
5 ## Resultado da Função Objetivo
6 print(pulp.value(E8.objective))
```

```
M1 = 0.11111111
M2 = 0.0
M3 = 0.44444444
M4 = 0.44444444
M5 = 0.0
7.433333258999999
```


Exercício 8 - Solução para o Cliente



Produção de uma nova liga com o menor custo conforme abaixo:

A nova liga é produzida ao menor custo possível misturando 11% do Minério 1, 44% do Minério 3 e 44% do Minério 4, além disso, a mistura contém 30% de Chumbo, 20% de Zinco e 50% de Estanho.

Custo total de produção da nova liga é: 7,44 \$/Kg.

Exercício 9 - Problema



EXERCÍCIO 9. Para um bar que funciona 24 horas por dia, a seguinte quantidade de empregados é necessária:

Hora do dia	Nº mínimo de empregados
2 - 6	4
6 - 10	8
10 - 14	10
14 - 18	7
18 - 22	12
22 - 2	4

Cada empregado trabalha 8 horas consecutivas por dia. O objetivo é achar o menor número necessário de empregados de modo que a necessidade mínima acima seja obedecida. Formule o problema como um modelo de P.Linear O objetivo é determinar as proporções destes minérios que deveriam ser misturados para produzir a nova liga com o menor custo possível.

Formule este problema como um modelo de Programação Linear.

Exercício 9 - Modelagem



VARIÁVEIS:

E_{6_14} = Escala de Trabalho 6hrs - 14hrs

E_{14_22} = Escala de Trabalho 14hrs - 22hrs

E_{22_6} = Escala de Trabalho 22hrs - 6hrs

FUNÇÃO OBJETIVO:

Minimizar empregados

$E_{6_14} + E_{14_22} + E_{22_6}$

RESTRIÇÕES:

Hora do dia 2 - 6

$E9+ = 0 * E_{6_14} + 0 * E_{14_22} + 1 * E_{22_6} \geq 4$

Hora do dia 6 - 10

$E9+ = 1 * E_{6_14} + 0 * E_{14_22} + 0 * E_{22_6} \geq 8$

Hora do dia 10 - 14

$E9+ = 1 * E_{6_14} + 0 * E_{14_22} + 0 * E_{22_6} \geq 10$

Hora do dia 14 - 18

$E9+ = 0 * E_{6_14} + 1 * E_{14_22} + 0 * E_{22_6} \geq 7$

Hora do dia 18 - 22

$E9+ = 0 * E_{6_14} + 1 * E_{14_22} + 0 * E_{22_6} \geq 12$

Hora do dia 22 - 2

$E9+ = 0 * E_{6_14} + 0 * E_{14_22} + 1 * E_{22_6} \geq 4$

Exercício 9 - Solução Excel (OpenSolver)



	E_6_14	E_14_22	E_22_6	
2_6			1	
6_10	1			
10_14	1			
14_18		1		
18_22		1		
22_2			1	
Empregados =	E_6_14	E_14_22	E_22_6	min
	10	12	4	26
	>=	>=	>=	
	≥ 0	0	0	

	E_2_10	E_10_18	E_18_2	
2_6	1			
6_10	1			
10_14		1		
14_18		1		
18_22			1	
22_2			1	
Empregados =	E_2_10	E_10_18	E_18_2	min
	8	10	12	30
	>=	>=	>=	
	≥ 0	0	0	

2_6		1	4	>=	4
6_10	1		10	>=	8
10_14	1		10	>=	10
14_18		1	12	>=	7
18_22		1	12	>=	12
22_2		1	4	>=	4

2_6	1		8	>=	4
6_10	1		8	>=	8
10_14		1	10	>=	10
14_18		1	10	>=	7
18_22			12	>=	12
22_2			12	>=	4

A forma que as escalas são construídas pode alterar a necessidade de contratação mínima de empregados. Conseguimos contratar 4 empregados a menos montando a escala certa.

Exercício 9 - Solução Python (PuLP)



```
E9 = pulp.LpProblem("E9",pulp.LpMinimize)

## Declara Variáveis
E_6_14 = pulp.LpVariable('E_6_14', lowBound=0, cat='Integer')
E_14_22 = pulp.LpVariable('E_14_22', lowBound=0, cat='Integer')
E_22_6 = pulp.LpVariable('E_22_6', lowBound=0, cat='Integer')

## Função Objetivo
E9+= E_6_14 + E_14_22 + E_22_6

## Constraints

E9+= 0*E_6_14 + 0*E_14_22 + 1*E_22_6 >= 4 #Hora do dia 2 - 6

E9+= 1*E_6_14 + 0*E_14_22 + 0*E_22_6 >= 8 # Hora do dia 6 - 10

E9+= 1*E_6_14 + 0*E_14_22 + 0*E_22_6 >= 10 # Hora do dia 10 - 14

E9+= 0*E_6_14 + 1*E_14_22 + 0*E_22_6 >= 7 # Hora do dia 14 - 18

E9+= 0*E_6_14 + 1*E_14_22 + 0*E_22_6 >= 12 # Hora do dia 18 - 22

E9+= 0*E_6_14 + 0*E_14_22 + 1*E_22_6 >= 4 # Hora do dia 22 - 2
```

```
## Solução
E9.solve()
pulp.LpStatus[E9.status]
```

'Optimal'

```
## Resultado das Variáveis
for variable in E9.variables():
    print ("{} = {}".format(variable.name, variable.varValue))
```

```
## Resultado da Função Objetivo
print (pulp.value(E9.objective))
```

```
E_14_22 = 12.0
E_22_6 = 4.0
E_6_14 = 10.0
26.0
```


Exercício 9 - Solução para o Cliente



Montar 3 escalas de trabalho conforme abaixo:

Escala de Trabalho 6hrs - 14hrs: contratar 10 empregados

Escala de Trabalho 14hrs - 22hrs: contratar 12 empregados

Escala de Trabalho 22hrs - 6hrs: contratar 4 empregados

Total de contratações: 26 empregados

Exercício 10 - Problema



EXERCÍCIO 10. Uma fábrica descontinuou a produção de um produto que não estava dando lucro. Isto criou uma considerável capacidade de produção ociosa. A gerência está considerando em usar esta capacidade ociosa em um ou mais, de 3 produtos, os quais chamaremos de produtos 1, 2 e 3. A capacidade disponível das máquinas que poderiam limitar a saída está dada na tabela abaixo:

Tipo de Máquina	Tempo disponível (em máquinas-hora por semana)
A	500
B	350
C	150

O número de máquinas-hora necessárias para cada produto é:

Tipo de Máquina	Produto 1	Produto 2	Produto 3
A	9	3	5
B	5	4	0
C	3	0	2

O Departamento de Vendas indicou que o potencial de vendas para os produtos 1 e 2 excedem a taxa máxima de produção e que o potencial de vendas para o produto 3 é de 20 unidades por semana. O lucro unitário seria de \$30, \$12 e \$15 respectivamente para os produtos 1, 2 e 3. Quanto se deve fabricar dos produtos 1, 2 e 3 de maneira que o lucro seja máximo?

Formule o problema como um modelo de Programação Linear.

Exercício 10 - Modelagem



VARIÁVEIS:

P1 = Produto 1

P2 = Produto 2

P3 = Produto 3

FUNÇÃO OBJETIVO:

Maximizar lucro semanal

$$30 * P1 + 12 * P2 + 15 * P3$$

RESTRIÇÕES:

Disponibilidade Máquina A

$$9 * P1 + 3 * P2 + 5 * P3 \leq 500$$

Disponibilidade Máquina B

$$5 * P1 + 4 * P2 + 0 * P3 \leq 350$$

Disponibilidade Máquina C

$$3 * P1 + 0 * P2 + 2 * P3 \leq 150$$

Demanda P3

$$1 * P3 \leq 20$$

Exercício 10 - Solução Excel (OpenSolver)



	P1	P2	P3
	26	55	20
	\geq	\geq	\geq
\geq	0	0	0

FO MAX \$30,00 \$12,00 \$15,00 ^{max} **\$1.740,00**

Máquina A	9	3	5	499	\leq	500
Máquina B	5	4	0	350	\leq	350
Máquina C	3	0	2	118	\leq	150
Demanda P3			1	20	\leq	20

Exercício 10 - Solução Python (PulP)



```
E10 = pulp.LpProblem("E10",pulp.LpMaximize)

## Declara Variáveis
P1 = pulp.LpVariable('P1', lowBound=0, cat='Integer')
P2 = pulp.LpVariable('P2', lowBound=0, cat='Integer')
P3 = pulp.LpVariable('P3', lowBound=0, cat='Integer')

## Função Objetivo

E10+= 30*P1 + 12*P2 + 15*P3

## Constraints

E10+= 9*P1 + 3*P2 + 5*P3 <= 500 # Disponibilidade Máquina A
E10+= 5*P1 + 4*P2 + 0*P3 <= 350 # Disponibilidade Máquina B
E10+= 3*P1 + 0*P2 + 2*P3 <= 150 # Disponibilidade Máquina C
E10+= 1*P3 <= 20 # Demanda P3
```

```
## Solução
E10.solve()
pulp.LpStatus[E10.status]
```

'Optimal'

```
## Resultado das Variáveis
for variable in E10.variables():
    print ("{} = {}".format(variable.name, variable.varValue))

## Resultado da Função Objetivo
print (pulp.value(E10.objective))
```

```
P1 = 26.0
P2 = 55.0
P3 = 20.0
1740.0
```


Exercício 10 - Solução para o Cliente



Produzir conforme abaixo:

Produto 1: 26 unidades

Produto 2: 55 unidades

Produto 3: 20 unidades

Lucro máximo semanal: \$1.740,00

Exercício 11 - Problema



EXERCÍCIO 11. O departamento de marketing de uma empresa estuda a forma mais econômica de aumentar em 30% as vendas de seus dois produtos P1 e P2. As alternativas são:

- Investir em um programa institucional com outras empresas do mesmo ramo. Esse programa deve proporcionar um aumento de 3% nas vendas de cada produto, para cada \$1.000,00 investidos.
- Investir diretamente na divulgação dos produtos. Cada \$1.000.00 investidos em P1 retornam um aumento de 4% nas vendas, enquanto que para P2 o retorno é de 10%.

A empresa dispõe de \$10.000,00 para esse empreendimento. Quanto deverá destinar a cada atividade? Formule o problema como um modelo de Programação Linear.

Exercício 11 - Modelagem



VARIÁVEIS:

Unidade: \$mil

PRO_INST_P1 = Prog. Inst. no Prod. 1

PRO_INST_P2 = Prog. Inst. no Prod. 2

DIVULGA_P1 = Divulgação do Prod. 1

DIVULGA_P2 = Divulgação do Prod. 2

FUNÇÃO OBJETIVO:

Valor alvo = 60

$$3 * \text{PRO_INST_P1} + 3 * \text{PRO_INST_P2} + \\ 4 * \text{DIVULGA_P1} + 10 * \text{DIVULGA_P2} \\ == 60$$

RESTRIÇÕES:

Orçamento mil \$

$$\text{PRO_INST_P1} + \text{PRO_INST_P2} + \text{DIVULGA_P1} + \\ \text{DIVULGA_P2} \leq 10$$

Crescimento Produto 1

$$3 * \text{PRO_INST_P1} + 4 * \text{DIVULGA_P1} == 30$$

Crescimento Produto 1

$$3 * \text{PRO_INST_P2} + 10 * \text{DIVULGA_P2} == 30$$

Exercício 11 - Solução Excel (OpenSolver)



Gasto \$mil **PRO_INST** **DIVULGA**

P1	<i>i</i> \$0,0	<i>i</i> \$7,5
P2	<i>i</i> \$0,0	<i>i</i> \$3,0

inteiro

NO FEASIBLE SOLUTION

\geq \leq 0

Valor Total Gasto \$mil = **\$10,5** \leq **\$10,0**

Aumento % **PRO_INST** **DIVULGA**

P1	3	4	30	=	= 30
P2	3	10	30	=	= 30

seek 60 **60**

FUNÇÃO OBJETIVO: = **60**

Exercício 11 - Solução Python (PulP)



```
E11 = pulp.LpProblem("E11",pulp.LpMaximize)

## Declara Variáveis
PRO_INST_P1 = pulp.LpVariable('PRO_INST_P1', lowBound=0, cat='Integer')
PRO_INST_P2 = pulp.LpVariable('PRO_INST_P2', lowBound=0, cat='Integer')
DIVULGA_P1 = pulp.LpVariable('DIVULGA_P1', lowBound=0, cat='Integer')
DIVULGA_P2 = pulp.LpVariable('DIVULGA_P2', lowBound=0, cat='Integer')

## Função Objetivo
E11+= 3*PRO_INST_P1 + 3*PRO_INST_P2 + 4*DIVULGA_P1 + 10*DIVULGA_P2 == 60

## Constraints
E11+= PRO_INST_P1 + PRO_INST_P2 + DIVULGA_P1 + DIVULGA_P2 <= 10 # Orçamento mil $
E11+= 3*PRO_INST_P1 + 4*DIVULGA_P1 == 30
E11+= 3*PRO_INST_P2 + 10*DIVULGA_P2 == 30
```

```
## Solução
E11.solve()
pulp.LpStatus[E11.status]
```

'Infeasible'

Exercício 11 - Solução para o Cliente



Para aumentar em 30% as vendas de P1 e P2, o orçamento terá que crescer de \$10.000 para **\$10.500** e ele deverá ser **investido integralmente na divulgação dos produtos.**

Exercício 12 - Problema



EXERCÍCIO 12. Um moinho fabrica produtos agrícolas para gado, ovelhas e galinhas. Esses produtos são fabricados a partir de quatro ingredientes: milho, calcário, soja e farinha de peixe, os quais contêm os seguintes nutrientes: vitamina, proteína, cálcio e gordura. As quantidades dos nutrientes em cada quilo dos ingredientes é dado na Tabela abaixo:

Ingredientes	Nutrientes			
	Vitamina	Proteína	Cálcio	Gordura
Milho	8	10	6	8
Calcário	6	5	10	6
Soja	10	12	6	6
Farinha de peixe	4	8	6	9

O moinho foi contratada para produzir 10, 6 e 8 toneladas métricas de alimentos para gado, ovelhas e galinhas, respectivamente. Por causa de escassez, uma quantidade limitada de ingredientes está disponível - 6 toneladas de milho, 10 toneladas de calcário, 4 toneladas de soja e 5 toneladas de farinha de peixe. O preço por quilo destes ingredientes é respectivamente \$0,20, \$0,12, \$0,24 e \$0,12. As quantidades máximas e mínimas por quilo dos vários nutrientes que são permitidas estão resumidas na Tabela abaixo.

Produtos	Nutrientes							
	Vitamina		Proteína		Cálcio		Gordura	
	Min	Max	Min	Max	Min	Max	Min	Max
Gado	6	∞	6	∞	7	∞	4	8
Ovelhas	6	∞	6	∞	6	∞	4	6
Galinhas	4	6	6	∞	6	∞	4	6



Considerar restrição MÁXIMA de Gordura 8 para os três produtos.

Formule este problema de minimizar os custos totais por um modelo de Programação Linear.

Exercício 12 - Modelagem



O QUE QUEREMOS? Quanto de massa de produto deve ser produzido para cada animal de maneira que o custo seja mínimo.

VARIÁVEIS:

Mij: quantidade a ser produzida dos ingredientes i , (i = Milho, Calcário, Soja, Farinha de Peixe) para os animais j , (j = Gado, Ovelha, Galinha))

MGado = Ingrediente Milho para o Gado

CGado = Ingrediente Calcário para o Gado

SGado = Ingrediente soja para o Gado

FGado = Ingrediente Farinha de Peixe para o Gado

MOvelha = Ingrediente Milho para a Ovelha

COvelha = Ingrediente Calcário para a Ovelha

SOvelha = Ingrediente Soja para a Ovelha

FOvelha = Ingrediente Farinha de Peixe para a Ovelha

MGalinha = Ingrediente Milho para a Galinha

CGalinha = Ingrediente Calcário para a Galinha

SGalinha = Ingrediente Soja para a Galinha

FGalinha = Ingrediente Farinha de Peixe para a Galinha

Exercício 12 - Modelagem



FUNÇÃO OBJETIVO: Minimizar o custo, dado por

$$0,2*(MGado + MOvelha + MGalinha) + 0,12*(CGado + COvelha + CGalinha) \backslash \\ + 0,24*(SOvelha + SGalinha + SGado) + 0.12*(FGado + FOvelha + FGalinha)$$

RESTRIÇÕES:

Demanda dos produtos em toneladas métricas

$$MGado + CGado + SGado + FGado \leq 10000$$

$$MOvelha + COvelha + SOvelha + FOvelha \leq 6000$$

$$MGalinha + CGalinha + SGalinha + FGalinha \leq 8000$$

Restrição de ingredientes em toneladas métricas

$$MGado + MOvelha + MGalinha \leq 6000$$

$$CGado + COvelha + CGalinha \leq 10000$$

$$SOvelha + SGalinha + SGado \leq 4000$$

$$FGado + FOvelha + FGalinha \leq 5000$$

Exercício 12 - Modelagem



RESTRIÇÕES:

VITAMINA permitida conforme tabela nutricional do animal

$$8 * \text{MGado} + 6 * \text{CGado} + 10 * \text{SGado} + 4 * \text{FGado} \geq 6 * 10 * 1000$$

$$8 * \text{MOvelha} + 6 * \text{COvelha} + 10 * \text{SOvelha} + 4 * \text{FOvelha} \geq 6 * 6 * 1000$$

$$8 * \text{MGalinha} + 6 * \text{CGalinha} + 10 * \text{SGalinha} + 4 * \text{FGalinha} \geq 4 * 8 * 1000$$

$$8 * \text{MGalinha} + 6 * \text{CGalinha} + 10 * \text{SGalinha} + 4 * \text{FGalinha} \leq 6 * 8 * 1000$$

PROTEÍNA permitida conforme tabela nutricional do animal

$$10 * \text{MGado} + 5 * \text{CGado} + 12 * \text{SGado} + 8 * \text{FGado} \geq 6 * 10 * 1000$$

$$10 * \text{MOvelha} + 5 * \text{COvelha} + 12 * \text{SOvelha} + 8 * \text{FOvelha} \geq 6 * 6 * 1000$$

$$10 * \text{MGalinha} + 5 * \text{CGalinha} + 12 * \text{SGalinha} + 8 * \text{FGalinha} \geq 6 * 8 * 1000$$

CÁLCIO permitida conforme tabela nutricional do animal

$$6 * \text{MGado} + 10 * \text{CGado} + 6 * \text{SGado} + 6 * \text{FGado} \geq 7 * 10 * 1000$$

$$6 * \text{MOvelha} + 10 * \text{COvelha} + 6 * \text{SOvelha} + 6 * \text{FOvelha} \geq 6 * 6 * 1000$$

$$6 * \text{MGalinha} + 10 * \text{CGalinha} + 6 * \text{SGalinha} + 6 * \text{FGalinha} \geq 6 * 8 * 1000$$

Exercício 12 - Modelagem



RESTRIÇÕES:

GORDURA permitida conforme tabela nutricional do animal

$$8 * \text{MGado} + 6 * \text{CGado} + 6 * \text{SGado} + 9 * \text{FGado} \geq 4 * 10 * 1000$$

$$8 * \text{MOvelha} + 6 * \text{COvelha} + 6 * \text{SOvelha} + 9 * \text{FOvelha} \geq 4 * 6 * 1000$$

$$8 * \text{MGalinha} + 6 * \text{CGalinha} + 6 * \text{SGalinha} + 9 * \text{FGalinha} \geq 4 * 8 * 1000$$

$$8 * \text{MGado} + 6 * \text{CGado} + 6 * \text{SGado} + 9 * \text{FGado} \leq 8 * 10 * 1000$$

$$8 * \text{MOvelha} + 6 * \text{COvelha} + 6 * \text{SOvelha} + 9 * \text{FOvelha} \leq 8 * 6 * 1000$$

$$8 * \text{MGalinha} + 6 * \text{CGalinha} + 6 * \text{SGalinha} + 9 * \text{FGalinha} \leq 8 * 8 * 1000$$

Exercício 12 - Solução Python (PulP)



```
E12 = pulp.LpProblem("E12",pulp.LpMinimize)
```

```
## Declara Variáveis
```

```
MGado = pulp.LpVariable('MGado', lowBound=0, cat='Continuous')
CGado = pulp.LpVariable('CGado', lowBound=0, cat='Continuous')
SGado = pulp.LpVariable('SGado', lowBound=0, cat='Continuous')
FGado = pulp.LpVariable('FGado', lowBound=0, cat='Continuous')
MOvelha = pulp.LpVariable('MOvelha', lowBound=0, cat='Continuous')
COvelha = pulp.LpVariable('COvelha', lowBound=0, cat='Continuous')
SOvelha = pulp.LpVariable('SOvelha', lowBound=0, cat='Continuous')
FOvelha = pulp.LpVariable('FOvelha', lowBound=0, cat='Continuous')
MGalinha = pulp.LpVariable('MGalinha', lowBound=0, cat='Continuous')
CGalinha = pulp.LpVariable('CGalinha', lowBound=0, cat='Continuous')
SGalinha = pulp.LpVariable('SGalinha', lowBound=0, cat='Continuous')
FGalinha = pulp.LpVariable('FGalinha', lowBound=0, cat='Continuous')
```

```
## Função Objetivo
```

```
E12+= 0.2*(MGado + MOvelha + MGalinha) + 0.12*(CGado + COvelha + CGalinha) \
      + 0.24*(SOvelha + SGalinha + SGado) + 0.12*(FGado + FOvelha + FGalinha)
```

```
## Constraints
```

```
# Demanda dos produtos em kilos
```

```
E12+= MGado + CGado + SGado + FGado == 10000
E12+= MOvelha + COvelha + SOvelha + FOvelha == 6000
E12+= MGalinha + CGalinha + SGalinha + FGalinha == 8000
```

```
# Restrição de ingredientes em kilos
```

```
E12+= MGado + MOvelha + MGalinha <= 6000
E12+= CGado + COvelha + CGalinha <= 10000
E12+= SOvelha + SGalinha + SGado <= 4000
E12+= FGado + FOvelha + FGalinha <= 5000
```

```
# VITAMINA permitida conforme tabela nutricional do animal - Por Kilo
```

```
E12+= 8*MGado + 6*CGado + 10*SGado + 4*FGado >= 6*10*1000
E12+= 8*MOvelha + 6*COvelha + 10*SOvelha + 4*FOvelha >= 6*6*1000
E12+= 8*MGalinha + 6*CGalinha + 10*SGalinha + 4*FGalinha >= 4*8*1000
E12+= 8*MGalinha + 6*CGalinha + 10*SGalinha + 4*FGalinha <= 6*8*1000
```

```
# PROTEÍNA permitida conforme tabela nutricional do animal
```

```
E12+= 10*MGado + 5*CGado + 12*SGado + 8*FGado >= 6*10*1000
E12+= 10*MOvelha + 5*COvelha + 12*SOvelha + 8*FOvelha >= 6*6*1000
E12+= 10*MGalinha + 5*CGalinha + 12*SGalinha + 8*FGalinha >= 6*8*1000
```

```
# CÁLCIO permitida conforme tabela nutricional do animal
```

```
E12+= 6*MGado + 10*CGado + 6*SGado + 6*FGado >= 7*10*1000
E12+= 6*MOvelha + 10*COvelha + 6*SOvelha + 6*FOvelha >= 6*6*1000
E12+= 6*MGalinha + 10*CGalinha + 6*SGalinha + 6*FGalinha >= 6*8*1000
```

```
# GORDURA permitida conforme tabela nutricional do animal
```

```
E12+= 8*MGado + 6*CGado + 6*SGado + 9*FGado >= 4*10*1000
E12+= 8*MOvelha + 6*COvelha + 6*SOvelha + 9*FOvelha >= 4*6*1000
E12+= 8*MGalinha + 6*CGalinha + 6*SGalinha + 9*FGalinha >= 4*8*1000
E12+= 8*MGado + 6*CGado + 6*SGado + 9*FGado <= 8*10*1000
E12+= 8*MOvelha + 6*COvelha + 6*SOvelha + 9*FOvelha <= 8*6*1000
E12+= 8*MGalinha + 6*CGalinha + 6*SGalinha + 9*FGalinha <= 8*8*1000
```

```
E12
```


Exercício 12 - Solução Python (PuLP)



```
1  ## Solução
2  E12.solve()
3  pulp.LpStatus[E12.status]
```

'Optimal'

```
1  ## Resultado das Variáveis
2  for variable in E12.variables():
3      print ("{} = {}".format(variable.name, variable.varValue))
4
5  ## Resultado da Função Objetivo
6  print (pulp.value(E12.objective))
```

```
CGado = 2500.0
CGalinha = 6153.8462
COvelha = 1346.1538
FGado = 1500.0
FGalinha = 1230.7692
FOvelha = 2269.2308
MGado = 6000.0
MGalinha = 0.0
MOvelha = 0.0
SGado = 0.0
SGalinha = 615.38462
SOvelha = 2384.6154
3720.0000047999997
```

Exercício 12 - Solução para o Cliente



Fabricação de produtos com quantidade de massa de ingredientes diversificado para cada animal de maneira que o custo da produção seja mínimo, conforme tabela abaixo:

	Gado	Ovelha	Galinha	Escassez
Milho	6.000	0	0	6.000
Calcário	2.500	1.346	6.154	10.000
Soja	0	2.385	615	3.000
Farinha de Peixe	1.500	2.269	1.231	5.000
Demanda	10.000	6.000	8.000	

Além disso, a fabricação dos produtos atende a restrição de escassez utilizando as quantidades limitadas de ingredientes disponíveis.

A demanda contratada de produção também foi atendida em **100%** para os três produtos contratados, Gado **10.000Kg**, Ovelha **6.000Kg** e Galinha **8.000Kg**.

Por fim, respeitando as quantidades mínimas e máximas nutricionais conforme especificado pelo cliente.

Custo total da produção: 3.720,00 \$/Kg.

Exercício 13 - Problema



5. Um avião deve ser carregado de modo a maximizar a receita total. Considere que há cinco itens que podem ser levados e que a empresa recebe 50 centavos por quilograma carregado, mais um valor extra (fixo), que depende do item. O avião pode carregar até 2000 kg e até 20 m³. Outros dados pertinentes estão na tabela a seguir.

Item	Peso (kg)	Volume (m ³)	Valor extra
1	1000	7	\$ 700
2	1100	10	\$ 800
3	700	10	\$ 1100
4	800	8	\$ 1000
5	500	5	\$ 700

Quais itens devem ser carregados?

Exercício 13 - Modelagem



O QUE QUEREMOS? Quais itens devem ser carregados no avião de modo que o lucro seja máximo.

DEFINIÇÃO DAS VARIÁVEIS:

$$X_i = \begin{cases} 1 & \text{se o item } i \text{ é carregado} \\ 0 & \text{se o item } i \text{ não é carregado, para } i = 1, 2, 3, 4, 5 \end{cases}$$

		ITENS					
		1	2	3	4	5	AVIÃO
Valor Extra \$		700	800	1100	1000	700	
Valor por Kilo \$		500	550	350	400	250	
Valor Total		1200	1350	1450	1400	950	CAPACIDADE
RESTRIÇÕES	Peso Kg	1000	1100	700	800	500	2000 Kg
	Volume m³	7	10	10	8	5	20 m³

Exercício 13 - Modelagem



FUNÇÃO OBJETIVO: Maximizar o lucro, dado por

$$1200 \cdot X_1 + 1350 \cdot X_2 + 1450 \cdot X_3 + 1400 \cdot X_4 + 950 \cdot X_5$$

RESTRIÇÕES:

Peso

$$1000 \cdot X_1 + 1100 \cdot X_2 + 700 \cdot X_3 + 800 \cdot X_4 + 500 \cdot X_5 \leq 2000 \text{Kg}$$

Volume

$$7 \cdot X_1 + 10 \cdot X_2 + 10 \cdot X_3 + 8 \cdot X_4 + 5 \cdot X_5 \leq 20 \text{m}^3$$

$$X_i \in B, \forall i$$

Exercício 13 – Solução (LPSolve)



Os itens X3 e X4 devem ser carregados no avião.

Lucro máximo: \$ 2.850,00.

LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

```
1 /* Função Objetivo */
2 max: 1200 x_1 + 1350 x_2 + 1450 x_3 + 1400 x_4 + 950 x_5;
3
4 /* Restrições */
5 1000 x_1 + 1100 x_2 + 700 x_3 + 800 x_4 + 500 x_5 <= 2000;
6 7 x_1 + 10 x_2 + 10 x_3 + 8 x_4 + 5 x_5 <= 20;
7
8 /* Tipo das variáveis */
9 bin x_1, x_2, x_3, x_4, x_5;
```

LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

Objective Constraints Sensitivity

Variables	MILP ...	MILP ...	MILP ...	MILP ...	result
	2150	2600	2650	2850	2850
x_3	0	0	1	1	1
x_4	0	1	0	1	1
x_1	1	1	1	0	0
x_2	0	0	0	0	0
x_5	1	0	0	0	0

Exercício 14 - Problema



6. Cinco locais (A, B, C, D, e E) precisam de uma peça grande para finalizar a construção de uma máquina, e as cinco peças existentes estão em cinco armazéns diferentes. As quilometragens entre os locais e os armazéns estão no seguinte quadro.

Armazém	Local				
	A	B	C	D	E
1	230	200	210	240	220
2	190	210	200	200	190
3	200	180	240	220	210
4	220	180	210	230	220
5	210	190	200	220	210

Determine o programa de expedição que minimiza a distância total no transporte das peças.

Exercício 14 - Modelagem



O QUE QUEREMOS? Transportar peças do armazém para o local de construção da máquina de modo que a distância no transporte das peças seja a menor possível.

$$x_{ij} = \begin{cases} 1 & \text{Peça transportada do armazém } i \text{ para o local de construção da máquina } j \\ 0 & \text{caso contrário, para } i = 1, 2, 3, 4, 5 \text{ e } j = A, B, C, D, E \end{cases}$$

FUNÇÃO OBJETIVO: Minimizar a distância do transporte das peças, dado por

$$\begin{aligned} &230 \cdot x_{1A} + 190 \cdot x_{2A} + 200 \cdot x_{3A} + 220 \cdot x_{4A} + 210 \cdot x_{5A} + 200 \cdot x_{1B} + 210 \cdot x_{2B} + 180 \cdot x_{3B} + 180 \cdot x_{4B} + 190 \cdot x_{5B} + \\ &210 \cdot x_{1C} + 200 \cdot x_{2C} + 240 \cdot x_{3C} + 210 \cdot x_{4C} + 200 \cdot x_{5C} + 240 \cdot x_{1D} + 200 \cdot x_{2D} + 220 \cdot x_{3D} + 230 \cdot x_{4D} + 220 \cdot x_{5D} + \\ &220 \cdot x_{1E} + 190 \cdot x_{2E} + 210 \cdot x_{3E} + 220 \cdot x_{4E} + 210 \cdot x_{5E} \end{aligned}$$

Exercício 14 - Modelagem



RESTRIÇÕES:

Cada armazém pode ter 1 peça enviada

$$X1A + X1B + X1C + X1D + X1E = 1$$

$$X2A + X2B + X2C + X2D + X2E = 1$$

$$X3A + X3B + X3C + X3D + X3E = 1$$

$$X4A + X4B + X4C + X4D + X4E = 1$$

$$X5A + X5B + X5C + X5D + X5E = 1$$

Cada local pode ter 1 peça recebida

$$X1A + X2A + X3A + X4A + X5A = 1$$

$$X1B + X2B + X3B + X4B + X5B = 1$$

$$X1C + X2C + X3C + X4C + X5C = 1$$

$$X1D + X2D + X3D + X4D + X5D = 1$$

$$X1E + X2E + X3E + X4E + X5E = 1$$

$$X_i \in B, \forall i$$

Exercício 14 – Solução Excel (OpenSolver)



ARMAZÉM	LOCAL				
	A	B	C	D	E
1	230	200	210	240	220
2	190	210	200	200	190
3	200	180	240	220	210
4	220	180	210	230	220
5	210	190	200	220	210

Armazém 1 irá transportar para o local C, o 2 para D, o 3 para A, o 4 para B e o 5 para E.

Distância total no transporte: **1000 Km**

	A	B	C	D	E
1	b 0	b 0	b 1	b 0	b 0
2	b 0	b 0	b 0	b 1	b 0
3	b 1	b 0	b 0	b 0	b 0
4	b 0	b 1	b 0	b 0	b 0
5	b 0	b 0	b 0	b 0	b 1

Restrições	1	1	1	1	1
=	1	1	1	1	1

Restrições	
1	1
1	1
1	= 1
1	1
1	1
1	1

min

1000

Função Objetivo Minimizar a distância

Exercício 15 - Problema



7. Uma empresa possui três fábricas (F1, F2 e F3), que enviam um tipo de produto a três clientes diferentes (C1, C2 e C3). A capacidade de produção de cada fábrica, a demanda de cada cliente e o custo unitário de transporte do produto estão na tabela a seguir.

	C1	C2	C3	Capacidade
F1	10	15	12	1800
F2	17	14	20	1400
F3	15	10	11	1300
Demanda	1200	1700	1600	

Além dos custos que dependem da quantidade de itens transportados, cada carga que sai de uma fábrica em direção a um cliente diferente possui um custo fixo de \$12000, \$11000 e \$13000 para as fábricas F1, F2 e F3, respectivamente. Qual é o custo mínimo que a empresa pode ter? E como ela deve fazer o transporte do produto?

Exercício 15 - Modelagem



Variáveis Binárias:
(0 não ativa; 1 ativa)

Fábrica 1

F1_C1 =
ATIVA_F1_PARA_C1
F1_C2 =
ATIVA_F1_PARA_C2
F1_C3 =
ATIVA_F1_PARA_C3

Fábrica 2

F2_C1 =
ATIVA_F2_PARA_C1
F2_C2 =
ATIVA_F2_PARA_C2
F2_C3 =

Variáveis Inteiras:
(qtd. de itens transportados)

Fábrica 1

C1_F1 = C1_RECEBE_DE_F1
C2_F1 = C2_RECEBE_DE_F1
C3_F1 = C3_RECEBE_DE_F1

Fábrica 2

C1_F2 = C1_RECEBE_DE_F2
C2_F2 = C2_RECEBE_DE_F2
C3_F2 = C3_RECEBE_DE_F2

Fábrica 3

C1_F3 = C1_RECEBE_DE_F3
C2_F3 = C2_RECEBE_DE_F3
C3_F3 = C3_RECEBE_DE_F3

Variáveis Inteiras de Custo:
(\$ variável e \$ fixo)

Custos

CV = CUSTO_VARIÁVEL
CF = CUSTO_FIXO

Exercício 15 - Modelagem



MINIMIZAR Função Objetivo = CV + CF

Custos

CV == **10***C1_F1 + **17***C1_F2 + **15***C1_F3 + **15***C2_F1 + **14***C2_F2 + **10***C2_F3 + **12***C3_F1 + **20***C3_F2 + **11***C3_F3

CF == **12000***(F1_C1 + F1_C2 + F1_C3) + **11000***(F2_C1 + F2_C2 + F2_C3) + **13000***(F3_C1 + F3_C2 + F3_C3)

C1_F1 <= **10000***F1_C1

C2_F1 <= **10000***F1_C2

C3_F1 <= **10000***F1_C3

C1_F2 <= **10000***F2_C1

C2_F2 <= **10000***F2_C2

C3_F2 <= **10000***F2_C3

C1_F3 <= **10000***F3_C1

C2_F3 <= **10000***F3_C2

C3_F3 <= **10000***F3_C3

Capacidade de Produção

C1_F1 + C2_F1 + C3_F1 <= **1800**

C1_F2 + C2_F2 + C3_F2 <= **1400**

C1_F3 + C2_F3 + C3_F3 <= **1300**

Demanda dos Clientes

C1_F1 + C1_F2 + C1_F3 == **1200**

C2_F1 + C2_F2 + C2_F3 == **1700**

C3_F1 + C3_F2 + C3_F3 == **1600**

10000: usado para **PERMITIR** o transporte dos itens **APENAS** se a **FÁBRICA FOR ATIVADA** para o **CLIENTE**.

Exercício 15 - Solução Excel (OpenSolver)



Variáveis Binárias:
(0 não ativa; 1 ativa)

Variáveis Inteiras:
(qtd. de itens transportados)

Variáveis Inteiras de Custo:
(\$ variável e \$ fixo)

Fábrica 1

F1_C1 = 1

F1_C2 = 0

F1_C3 = 1

Fábrica 1

C1_F1 = 1.200

C2_F1 = 0

C3_F1 = 600

Fábrica 2

F2_C1 = 0

F2_C2 = 1

F2_C3 = 0

Fábrica 2

C1_F2 = 0

C2_F2 = 1.400

C3_F2 = 0

Fábrica 3

F3_C1 = 0

F3_C2 = 1

F3_C3 = 1

Fábrica 3

C1_F3 = 0

C2_F3 = 300

C3_F3 = 1.000

Custos

CV = \$52.800

CF = \$61.000

MINIMIZAR Função Objetivo

CV + CF = **\$113.800**

Exercício 16 - Problema



9. Um empresa de telefonia possui \$ 7,4 milhões destinados à construção de até seis antenas de celular para cobrir a maior população possível em 9 comunidades contíguas. As comunidades cobertas por cada uma das antenas e os custos de construção estão listados abaixo.

Antena	Comunidades cobertas	Custo (\$ milhões)
1	1, 2, 4	2,2
2	2, 3, 6	2,3
3	6, 8	2,5
4	4, 5, 6, 7	4,1
5	3, 5, 6, 8, 9	4,8
6	1, 3, 4, 7, 9	5,1

A tabela a seguir mostra a população, em milhares de habitantes, em cada uma das comunidades.

Comunidade	1	2	3	4	5	6	7	8	9
População	5	8	6	7	9	13	11	7	4

Quais antenas devem ser construídas? Quais comunidades estarão cobertas?

Exercício 16 - Modelagem



VARIÁVEIS:

X_i - Variável binária, igual a 1 se a antena i foi construída e 0 caso contrário

Y_i - Variável binária, igual a 1 se a comunidade i foi coberta e 0 caso contrário

OBJETIVO:

Maximizar o número de pessoas atendidas, dado por (em milhares)

$$5*Y_1 + 8*Y_2 + 6*Y_3 + 7*Y_4 + 9*Y_5 + 13*Y_6 + 11*Y_7 + 7*Y_8 + 4*Y_9$$

RESTRIÇÕES:

$$2.2*X_1 + 2.3*X_2 + 2.5*X_3 + 4.1*X_4 + 4.8*X_5 + 5.1*X_6 \leq 7.4 \text{ (Custo)}$$

CADA PAR SEGUINTE DE RESTRIÇÕES “AMARRA” OS Y’S COM OS X’S CORRESPONDENTES. A ORDEM É DA COMUNIDADE 1 ATÉ A COMUNIDADE 9, RESPECTIVAMENTE

$$X_1 + X_6 - 2*Y_1 \leq 0$$

$$X_1 + X_6 - Y_1 \geq 0$$

$$X_2 + X_3 + X_4 + X_5 - 4*Y_6 \leq 0 \quad X_2 + X_3 +$$

$$X_4 + X_5 - Y_6 \geq 0$$

$$X_1 + X_2 - 2*Y_2 \leq 0$$

$$X_1 + X_2 - Y_2 \geq 0$$

$$X_4 + X_6 - 2*Y_7 \leq 0$$

$$X_4 + X_6 - Y_7 \geq 0$$

$$X_2 + X_5 + X_6 - 3*Y_3 \leq 0 \quad X_2 + X_5 + X_6 - Y_3 \geq 0$$

$$X_3 + X_5 - 2*Y_8 \leq 0$$

$$X_3 + X_5 - Y_8$$

$$\geq 0$$

$$X_1 + X_4 + X_6 - 3*Y_4 \leq 0 \quad X_1 + X_4 + X_6 - Y_4 \geq 0$$

$$X_5 + X_6 - 2*Y_9 \leq 0$$

$$X_5 + X_6 - Y_9$$

$$\geq 0$$

$$X_4 + X_5 - 2*Y_5 \leq 0$$

$$X_4 + X_5 - Y_5 \geq 0$$

Exercício 16 – Solução (LPSolve)



LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

```
1 /* Função Objetivo */
2 max: 5 j_1 + 8 j_2 + 6 j_3 + 7 j_4 + 9 j_5 + 13 j_6 + 11 j_7 + 7 j_8 + 4 j_9;
3
4 /* Restrições */
5 2.2 x_1 + 2.3 x_2 + 2.5 x_3 + 4.1 x_4 + 4.8 x_5 + 5.1 x_6 <= 7.4;
6 x_1 + x_6 - 2 j_1 <= 0;
7 x_1 + x_6 - 1 j_1 >= 0;
8 x_1 + x_2 - 2 j_2 <= 0;
9 x_1 + x_6 - 1 j_2 >= 0;
10 x_2 + x_5 + x_6 - 3 j_3 <= 0;
11 x_2 + x_5 + x_6 - 1 j_3 >= 0;
12 x_1 + x_4 + x_6 - 3 j_4 <= 0;
13 x_1 + x_4 + x_6 - 1 j_4 >= 0;
14 x_4 + x_5 - 2 j_5 <= 0;
15 x_4 + x_5 - 1 j_1 >= 0;
16 x_2 + x_3 + x_4 + x_5 - 4 j_6 <= 0;
17 x_2 + x_3 + x_4 + x_5 - 1 j_6 >= 0;
18 x_4 + x_6 - 2 j_7 <= 0;
19 x_4 + x_6 - 1 j_7 >= 0;
20 x_3 + x_5 - 2 j_8 <= 0;
21 x_3 + x_5 - 1 j_8 >= 0;
22 x_5 + x_6 - 2 j_9 <= 0;
23
24 /* Tipo das Variáveis */
25 bin j_1, j_2, j_3, j_4, j_5, j_6, j_7, j_8, j_9, x_1, x_2, x_3, x_4, x_5, x_6;
```

LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

Objective Constraints Sensitivity

Variables	MILP...	re...
	59	59
L1	1	1
L2	1	1
L3	1	1
L4	1	1
L5	1	1
L6	1	1
L8	1	1
L9	1	1
x_1	1	1
x_5	1	1
L7	0	0
x_2	0	0
x_3	0	0
x_4	0	0
x_6	0	0

Portanto, devem ser construídas as antenas 1 e 5, que cobrirão as comunidades 1, 2, 3, 4, 5, 6, 8 e 9, totalizando 59 mil habitantes.

Exercício 17 - Problema



2) Considere o problema da mochila abaixo:

$$\begin{aligned} \max \quad & 9x_1 + 7x_2 + 5x_3 + 2x_4 \\ \text{s.a.} \quad & 5x_1 + 4x_2 + 3x_3 + 2x_4 \leq 7 \\ & x_1, x_2, x_3, x_4 = 0 \text{ ou } 1 \end{aligned}$$

Usando o método Branch-and-bound determina a solução ótima do problema. Elabore uma maneira sistemática e fácil de se obter a relaxação linear para este problema.

Exercício 17 - Modelagem



	ITENS					
	X1	X2	X3	X4		
	1	2	3	4		CAPACIDADE
Valor por Kilo \$	9	7	5	2		
Peso (massa) Kg	5	4	3	2	<=	7
Valor Relativo	1,80	1,75	1,67	1,00		
RELAXAÇÃO	1	2/4	0	0		12,5

Solução viável para o problema relaxado, no entanto não é uma solução ótima para o problema original, pois a solução precisa ser inteira.

Uma maneira sistemática para a relaxação é calcular o valor relativo das variáveis, ou seja, a relação entre o benefício e o peso e ordenar os valores em ordem crescente. Dessa forma, quanto maior for essa relação melhor é o candidato para entrar na mochila. Considerando essa sequência, é só ir acrescentando na mochila as variáveis até o limite da capacidade e fracionar quando necessário.

Exercício 17 - Modelagem



Sendo assim, a ordem de escolha das variáveis será: 1,2,3,4.

RAMIFICAÇÃO 1	1	0	2/3	0	X2 = 0	12,3
	3/5	1	0	0	X2 = 1	12,4
RAMIFICAÇÃO 2	0	1	1	0	X1 = 0	12,0
	1	1	0	0	X1 = 1	
RAMIFICAÇÃO 3	1	0	0	1	X3 = 0	11,0
	1	0	1	0	X3 = 1	

Não é uma solução ótima, pois nem todas as variáveis são binárias.

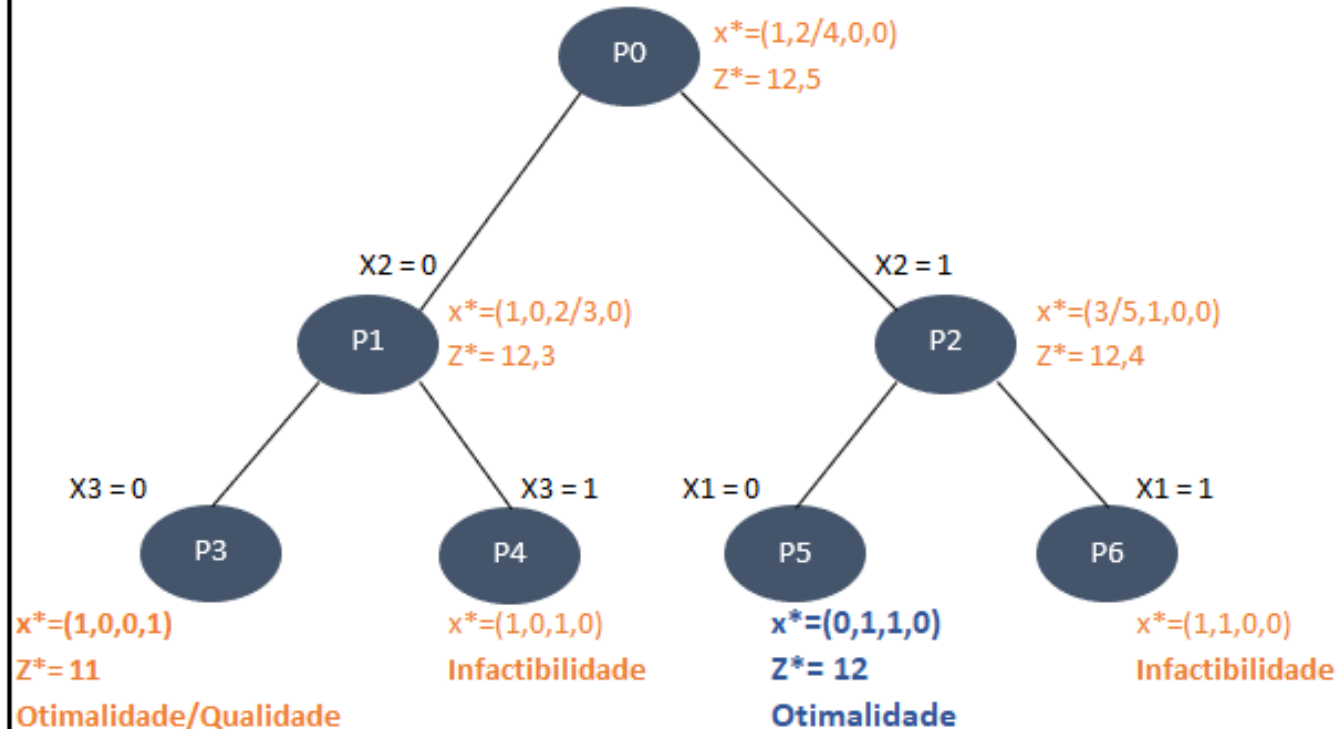
Não é uma solução ótima, pois nem todas as variáveis são binárias.

Solução ótima para o problema, pois a solução é inteira para todas as variáveis.

Infactível, pois ultrapassa a capacidade.

Solução ótima para o problema, pois a solução é inteira para todas as variáveis.

Infactível, pois ultrapassa a capacidade.



EXPLORAÇÃO DE 7 NÓS.

SOLUÇÃO ÓTIMA = 12

X1 = 0

X2 = 1

X3 = 1

X4 = 0

Exercício 17 – Solução (LPSolve)



LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

```
1 /* Objective function */
2 max: 9 * x_1 + 7 * x_2 + 5 * x_3 + 2 * x_4;
3
4 /* Variable bounds */
5
6 5 * x_1 + 4 * x_2 + 3 * x_3 + 2 * x_4 <= 7;
7
8 /* Variable bounds */
9
10 bin x_1, x_2, x_3, x_4;
```

Log Messages

Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Relaxed solution 12.5 after 3 iter is B&B base.

Feasible solution 12 after 5 iter, 3 nodes (gap 0.0%)

Optimal solution 12 after 5 iter, 3 nodes (gap 0.0%).

Relative numeric accuracy ||*|| = 0

MEMO: lp_solve version 5.5.2.5 for 32 bit OS, with 64 bit REAL variables.
In the total iteration count 5, 2 (40.0%) were bound flips.
There were 1 refactorizations, 0 triggered by time and 0 by density.
... on average 3.0 major pivots per refactorization.
The largest [LUSOL v2.2.1.0] fact(B) had 3 NZ entries, 1.0x largest basis.
The maximum B&B level was 3, 0.4x MIP order, 3 at the optimal solution.
The constraint matrix inf-norm is 5, with a dynamic range of 2.5.
Time to load data was 0.003 seconds, presolve used 0.007 seconds,
... 0.025 seconds in simplex solver, in total 0.035 seconds.

LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

Objective Constraints Sensitivity

Variables	MILP ...	result ▼
	12	12
x_2	1	1
x_3	1	1
x_1	0	0
x_4	0	0

No Solver a exploração foi de **3 NÓS**.

Na resolução manual a exploração foi de **7 NÓS**.

Exercício 18 - Problema



3) Considere o problema abaixo:

$$\begin{array}{ll}\max & 6x_1 + 9x_2 + 7x_3 \\ \text{sujeito a} & 3x_1 + 5x_2 + 4x_3 \leq 14 \\ & x_1, x_2, x_3 \geq 0 \text{ e inteiros}\end{array}$$

Usando o método Branch-and-bound determina a solução ótima do problema.

Exercício 18 – Solução (LPSolve)



LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

```
1 /* Função Objetivo */
2 max: 6 x_1 + 9 x_2 + 7 x_3;
3
4 /* Restrições */
5 3 x_1 + 5 x_2 + 4 x_3 <= 14;
6
7 /* Tipo das variáveis */
8 int x_1, x_2, x_3;
```

Exploração máxima foi de **4 NÓS**.

Solução ótima **27**.

$x_1 = 3$, $x_2 = 1$ e $x_3 = 0$

LPSolve IDE - 5.5.2.5

File Edit Search Action View Options Help



Source Matrix Options Result

Objective Constraints Sensitivity

Variables	MILP ...	result
	27	27
x_1	3	3
x_2	1	1
x_3	0	0

Log Messages

SUBMITTED

Model size: 1 constraints, 3 variables, 3 non-zeros.
Sets: 0 GUB, 0 SOS.

Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

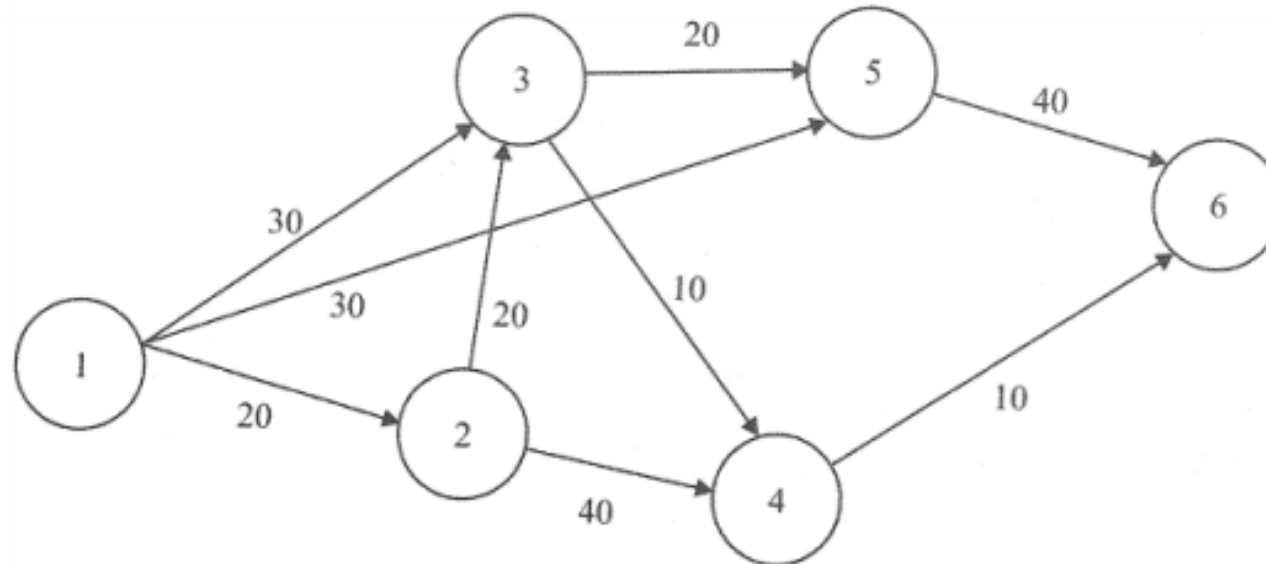
Relaxed solution 28 after 1 iter is B&B base.
Feasible solution 27 after 3 iter, 3 nodes (gap 3.4%)
Optimal solution 27 after 6 iter, 6 nodes (gap 3.4%).
Relative numeric accuracy ||*|| = 0

MEMO: lp_solve version 5.5.2.5 for 32 bit OS, with 64 bit REAL variables.
In the total iteration count 6, 0 (0.0%) were bound flips.
There were 3 refactorizations, 0 triggered by time and 0 by density.
... on average 2.0 major pivots per refactorization.
The largest [LUSOL v2.2.1.0] fact(B) had 3 NZ entries, 1.0x largest basis.
The maximum B&B level was 4, 0.7x MIP order, 3 at the optimal solution.
The constraint matrix inf-norm is 5, with a dynamic range of 1.66667.
Time to load data was 0.018 seconds, presolve used 0.008 seconds,
... 0.018 seconds in simplex solver, in total 0.044 seconds.

Exercício 19 - Problema



Exercício 4.9 Suponha que uma empresa transportadora de cargas possui três caminhões que estão atualmente nas posições 1 e 2 da rede rodoviária de uma cidade esquematizada a seguir. No nó 1 estão dois caminhões da empresa e no nó 2 está o terceiro caminhão. Um novo pedido de frete foi solicitado na localidade 6. Para este pedido, são necessários três caminhões. Além disso, a carga precisa ser retirada no máximo dentro de 50 minutos, pois depois desse prazo não é mais possível estacionar o caminhão próximo do local, por problemas de tráfego. Determine se a empresa transportadora poderá atender a este pedido. Os números ao lado de cada arco representam o número de minutos para um caminhão percorrer o arco.



Exercício 19 - Solução Excel (OpenSolver)



- 2 caminhões saindo do nó 1 até o nó 6.
TEMPO TOTAL: 30 MINUTOS

				Restrições				
Início	Fim	Caminho	Tempo (min)	Grafo (Vértices)	Fluxo		Início = 1 e Fim = -1	
1	2	b 0	20	1	1	=	1	Caminhão A e B saíndo do ponto 1
1	3	b 0	30	2	0	=	0	
1	5	b 1	30	3	0	-	0	
2	4	b 0	40	4	0	=	0	
2	3	b 0	20	5	-1	=	-1	
3	4	b 0	10					
3	5	b 0	20					
4	6	b 0	10	Tempo total	30	←	≤	50
5	6	b 0	40					
Min Tempo		30						
Tempo total para o caminhão A + B				30				

Exercício 19 - Solução Excel (OpenSolver)



- 1 caminhão saindo do nó 2 até o nó 6.
TEMPO TOTAL: 40 MINUTOS

				Restrições					
Início	Fim	Caminho	Tempo (min)	Grafo (Vértices)	Fluxo		Início = 1 e Fim = -1		
1	2	b 0	20	1	0	=	0	Caminhão C saído do ponto 2	
1	3	b 0	30	2	1	=	1		
1	5	b 0	30	3	0	-	0		
2	4	b 0	40	4	0	=	0		
2	3	b 1	20	5	-1	=	-1		
3	4	b 0	10						
3	5	b 1	20						
4	6	b 0	10	Tempo total	40	←	≤	50	
5	6	b 0	40						
Min Tempo		40							
Tempo total para o caminhão C				40					

Exercício 19 - Solução para o Cliente



- 2 caminhões saindo do nó 1 até o nó 6 + 1 caminhão saindo do nó 2 até o nó 6.

TEMPO TOTAL: 70 MINUTOS

A EMPRESA TRANSPORTADORA NÃO PODERÁ ATENDER ESSE PEDIDO, POIS VIOLA A RESTRIÇÃO DE TEMPO MÁXIMO SOLICITADA PELO CLIENTE DE 50 MINUTOS.

INSTITUTO TECNOLÓGICO DE AERONÁUTICA



Contato

Ana Paula dos Santos Gularte

[linkedin.com/in/ana-paula-santos-gularte-66b90526](https://www.linkedin.com/in/ana-paula-santos-gularte-66b90526)

Jaraguá do Sul, SC

anasantospoa@gmail.com

(47) 99753-8080

OBRIGADA!