

# Introdução à Otimização

## Aula 3

A Modeling Language for

Mathematical Programming: **AMPL**

Ambiente de desenvolvimento: **GUSEK**

Solver: **GLPK**

# Modelo de Programação Linear

## Análise de Atividades

### Forma compacta

$$\text{Max } Z = \sum_{j=1}^n c_j x_j$$

*Sujeito a*

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \forall i = 1, \dots, m$$

$$x_j \geq 0, \forall j = 1, \dots, n$$

Temos uma matriz  $A_{m \times n}$  retangular.

Normalmente  $n > m$ , ou seja, A tem mais colunas do que linhas.

## Modelo Compacto Melhorado

|             | Liga tipo A | Liga tipo B | Disp. max |
|-------------|-------------|-------------|-----------|
| Cobre       | 2           | 1           | 16        |
| Zinco       | 1           | 2           | 11        |
| Chumbo      | 1           | 3           | 15        |
| P. de venda | \$30,00     | \$50,00     |           |

**Conjuntos de índices:** `Ligas := {A, B}`, `MP := {Cobre, Zinco, Chumbo}`

**Declaração dos parâmetros de entrada:**

`PV[j], j ∈ Ligas; // preço de venda/unid. da liga j;`

`Matriz[i,j], i ∈ MP, j ∈ Ligas; // cons. da MP i por unid. da liga j`

`Disp[i], i ∈ MP; //disponibilidade do material i;`

**Conjuntos:**  $Ligas := \{A, B\}$ ,  $MP := \{\text{Cobre}, \text{Zinco}, \text{Chumbo}\}$

**Parâmetros:**

$PV[j]$ ,  $j \in Ligas$ ;

$Matriz[i, j]$ ,  $i \in MP$ ,  $j \in Ligas$ ;

$Disp[i]$ ,  $i \in MP$ ;

**Variáveis de decisão:**

$X[j]$ ,  $j \in Ligas$ ,  $\geq 0$ ;

**Função objetivo:**

Maximize  $Z$ :  $\sum\{j \in Ligas\} PV[j] * X[j]$ ;

**Restrições:**

$Restr\{i \in MP\}$ :  $\sum\{j \in Ligas\} Matriz[i, j] * X[j] \leq Disp[i]$ ;

**Conjuntos:**  $Ligas := \{A, B\}$ ,  $MP := \{\text{Cobre}, \text{Zinco}, \text{Chumbo}\}$

**Parâmetros:**

$PV[j]$ ,  $j \in Ligas$ ;

$Matriz[i, j]$ ,  $i \in MP$ ,  $j \in Ligas$ ;

$Disp[i]$ ,  $i \in MP$ ;

**Variáveis de decisão:**

$X[j]$ ,  $j \in Ligas$ ,  $\geq 0$ ;

**Função objetivo:**

$\text{Max } Z = \sum\{j \in Ligas\} PV[j] * X[j]$ ;

$\text{Max } Z = PV[A] * X[A] + PV[B] * X[B]$

**Restrições:**

$\text{Restr}\{i \in MP\}: \sum\{j \in Ligas\} Matriz[i, j] * X[j] \leq Disp[i]$ ;

**Conjuntos:** Ligas := {A, B}, MP := {Cobre, Zinco, Chumbo}

**Parâmetros:**

PV[j], j ∈ Ligas;

Matriz[i,j], i ∈ MP, j ∈ Ligas;

Disp[i], i ∈ MP;

**Variáveis de decisão:**

X[j], j ∈ Ligas, ≥ 0;

**Função objetivo:**

Max Z = sum{j ∈ Ligas} PV[j]\*X[j];

Max Z = PV[A]\*X[A] + PV[B]\*X[B]

**Restrições:**

Restr{i ∈ MP}: sum{j ∈ Ligas} Matriz[i,j]\*X[j] ≤ Disp[i];

i = Cobre: sum{j ∈ {A, B}}

Matriz[Cobre, A]\*X[A] + Matriz[Cobre, B]\*X[B] ≤ Disp[Cobre]

**Conjuntos:** Ligas := {A, B}, MP := {Cobre, Zinco, Chumbo}

**Parâmetros:**

PV[j], j ∈ Ligas;

Matriz[i,j], i ∈ MP, j ∈ Ligas;

Disp[i], i ∈ MP;

**Variáveis de decisão:**

X[j], j ∈ Ligas, ≥ 0;

**Função objetivo:**

Max Z = sum{j ∈ Ligas} PV[j]\*X[j];

Max Z = PV[A]\*X[A] + PV[B]\*X[B]

**Restrições:**

Restr{i ∈ MP}: sum{j ∈ Ligas} Matriz[i,j]\*X[j] ≤ Disp[i];

i = Cobre: sum{j ∈ {A, B}}

Matriz[Cobre, A]\*X[A] + Matriz[Cobre, B]\*X[B] ≤ Disp[Cobre]

i = Zinco: sum{j ∈ {A, B}}

Matriz[Zinco, A]\*X[A] + Matriz[Zinco, B]\*X[B] ≤ Disp[Zinco]

**Conjuntos:** Ligas := {A, B}, MP := {Cobre, Zinco, Chumbo}

**Parâmetros:**

PV[j], j ∈ Ligas;

Matriz[i,j], i ∈ MP, j ∈ Ligas;

Disp[i], i ∈ MP;

**Variáveis de decisão:**

X[j], j ∈ Ligas, ≥ 0;

**Função objetivo:**

Max Z = sum{j ∈ Ligas} PV[j]\*X[j];

Max Z = PV[A]\*X[A] + PV[B]\*X[B]

**Restrições:**

Restr{i ∈ MP}: sum{j ∈ Ligas} Matriz[i,j]\*X[j] ≤ Disp[i];

i = Cobre: sum{j ∈ {A, B}}

Matriz[Cobre, A]\*X[A] + Matriz[Cobre, B]\*X[B] ≤ Disp[Cobre]

i = Zinco: sum{j ∈ {A, B}}

Matriz[Zinco, A]\*X[A] + Matriz[Zinco, B]\*X[B] ≤ Disp[Zinco]

i = Chumbo: sum{j ∈ {A, B}}

Matriz[Chumbo, A]\*X[A] + Matriz[Chumbo, B]\*X[B] ≤ Disp[Chumbo]



**Conjuntos:**  $Ligas := \{A, B\}$ ,  $MP := \{\text{Cobre}, \text{Zinco}, \text{Chumbo}\}$

**Parâmetros:**

$PV[j]$ ,  $j \in Ligas$ ;

$Matriz[i,j]$ ,  $i \in MP$ ,  $j \in Ligas$ ;

$Disp[i]$ ,  $i \in MP$ ;

**Variáveis de decisão:**

$X[j]$ ,  $j \in Ligas$ ,  $\geq 0$ ;

**Função objetivo:**

$\text{Max } Z = \sum\{j \in Ligas\} PV[j] * X[j]$ ;

$\text{Max } Z = PV[A] * X[A] + PV[B] * X[B]$

**Restrições:**

$\text{Restr}\{i \in MP\}: \sum\{j \in Ligas\} Matriz[i,j] * X[j] \leq Disp[i]$ ;

$Matriz[\text{Cobre}, A] * X[A] + Matriz[\text{Cobre}, B] * X[B] \leq Disp[\text{Cobre}]$

$Matriz[\text{Zinco}, A] * X[A] + Matriz[\text{Zinco}, B] * X[B] \leq Disp[\text{Zinco}]$

$Matriz[\text{Chumbo}, A] * X[A] + Matriz[\text{Chumbo}, B] * X[B] \leq Disp[\text{Chumbo}]$

## Modelo Compacto do GUSEK

```
set MP := {1..3};      # conjunto das matérias primas
set Ligas := {1..2};   # conjunto das ligas
param PV{Ligas};       # preço de venda de cada liga
param disp{i in MP};  # disponibilidade de cada matéria prima
param mat{i in MP, j in Ligas}; # matriz de consumo
var x{j in Ligas}, >=0; # quantidade de liga a ser produzida

maximize Lucro: sum{j in Ligas} x[j] * PV[j];
Disp_MP{i in MP}: sum{j in Ligas} mat[i, j] * x[j] <= disp[i];
solve;
```

## Modelo Compacto do **GUSEK** (cont.)

**data**;

**param** PV :=

1 30

2 50;

Ativar a opção de geração do  
arquivo com o resumo da  
solução

**param** disp :=

1 16

3 15

2 11;

**param** mat :

1 2 :=

1 2 1

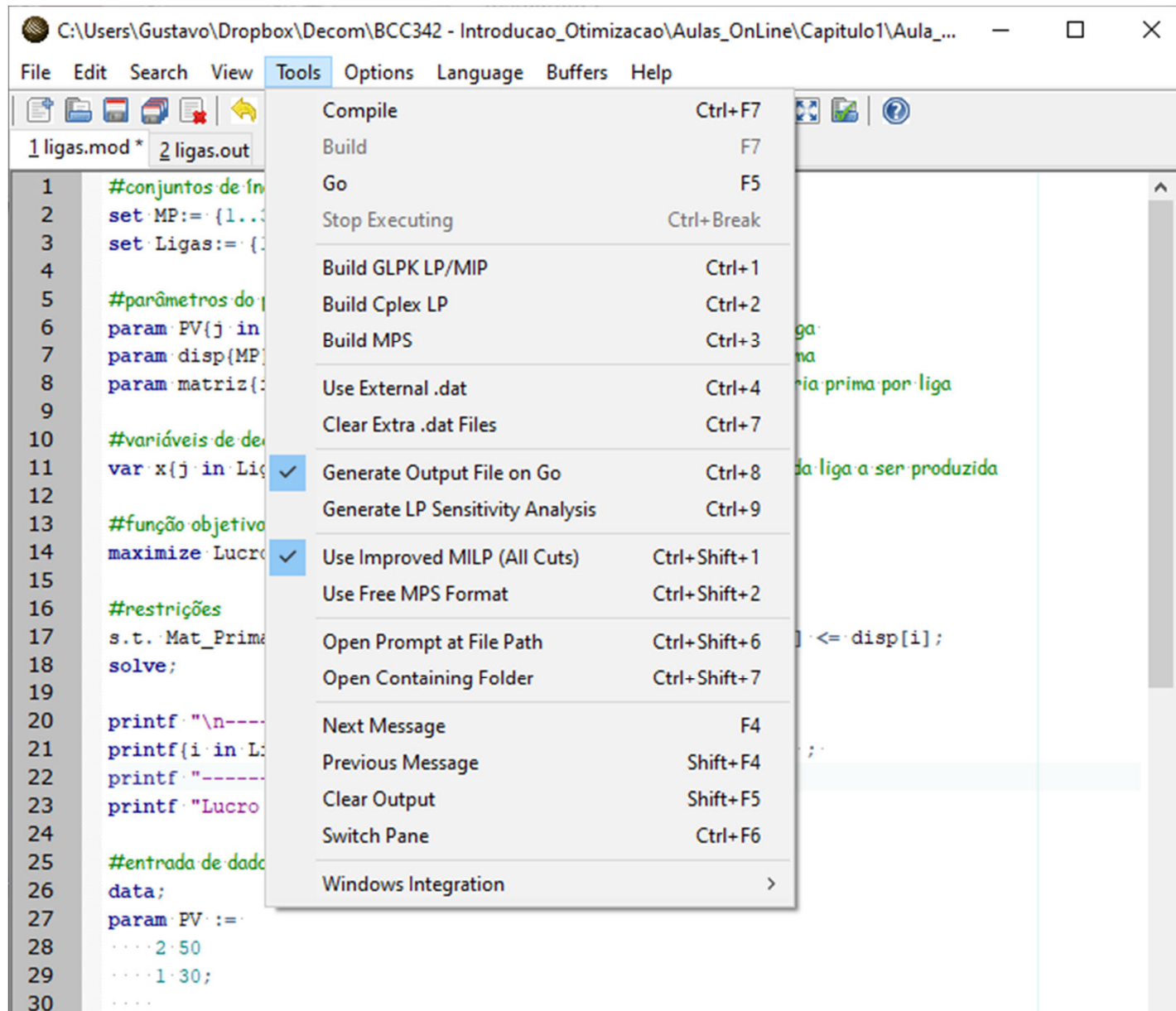
2 1 2

3 1 3;

**end**;

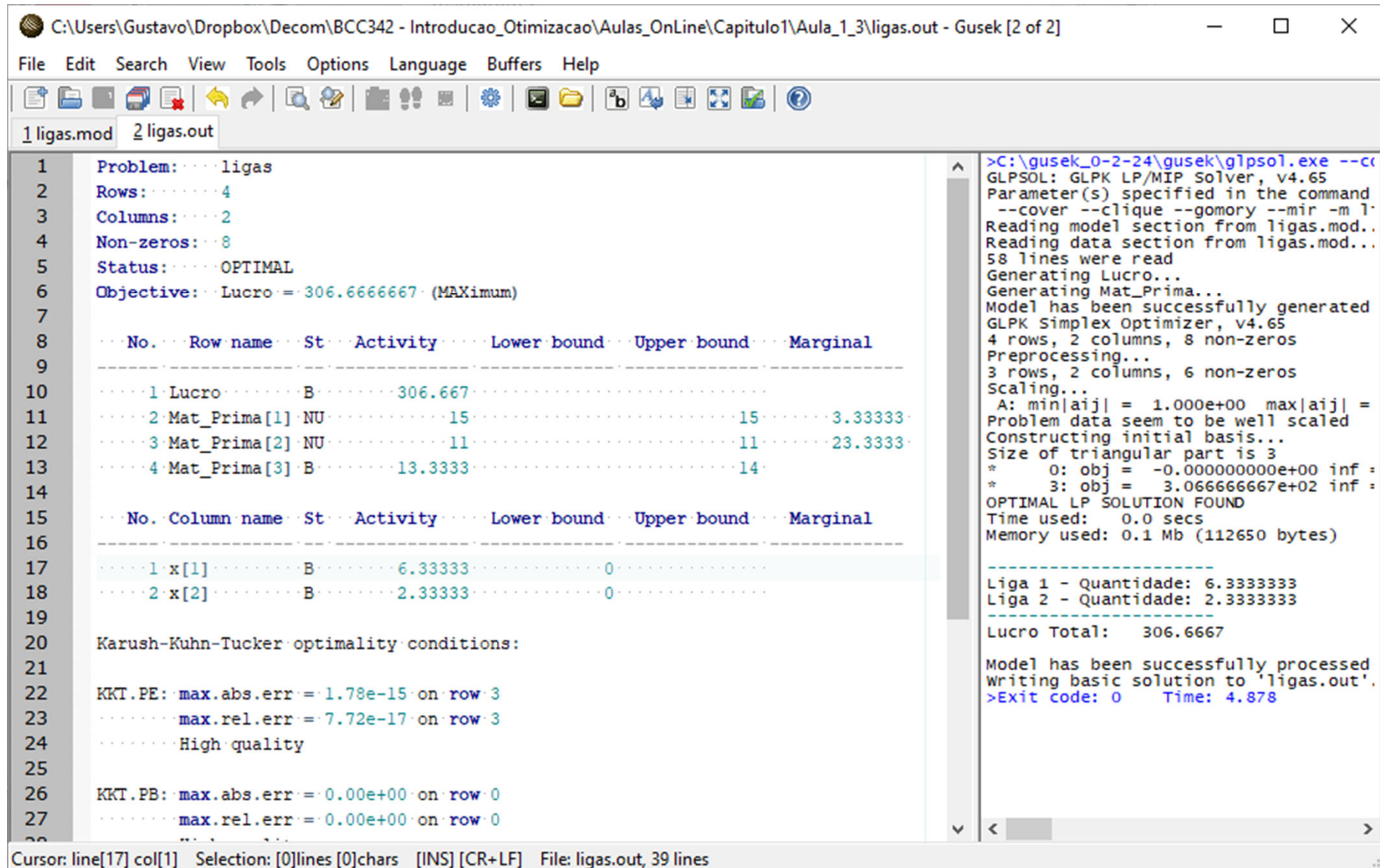
## Modelo Compacto do **GUSEK** (cont.)

Selecione a opção **Tools** -> **Generate Output File on Go** para que seja gerado o arquivo com o resumo do resultado, ou seja, o arquivo .out



# Modelo Compacto do **GUSEK** (cont.)

Arquivo ***ligas.out*** -> relatório da solução obtida ao resolver o modelo em ***ligas.mod***.



The screenshot shows a GUSEK application window titled "C:\Users\Gustavo\Dropbox\Decom\BCC342 - Introducao\_Otimizacao\Aulas\_OnLine\Capitulo1\Aula\_1\_3\ligas.out - Gusek [2 of 2]". The window has a menu bar (File, Edit, Search, View, Tools, Options, Language, Buffers, Help) and a toolbar. Below the toolbar, there are two tabs: "1 ligas.mod" and "2 ligas.out". The main area is split into two panes. The left pane shows the problem definition and solution results for "ligas.mod". The right pane shows the output of the GLPK solver.

**Problem: ligas**

Rows: 4  
Columns: 2  
Non-zeros: 8  
Status: OPTIMAL  
Objective: Lucro = 306.666667 (MAXimum)

| No. | Row name        | St | Activity | Lower bound | Upper bound | Marginal |
|-----|-----------------|----|----------|-------------|-------------|----------|
| 1   | Lucro           | B  | 306.667  |             |             |          |
| 2   | Mat_Prime[1] NU |    | 15       | 15          |             | 3.33333  |
| 3   | Mat_Prime[2] NU |    | 11       | 11          |             | 23.3333  |
| 4   | Mat_Prime[3] B  |    | 13.3333  | 14          |             |          |

| No. | Column name | St | Activity | Lower bound | Upper bound | Marginal |
|-----|-------------|----|----------|-------------|-------------|----------|
| 1   | x[1]        | B  | 6.33333  | 0           |             |          |
| 2   | x[2]        | B  | 2.33333  | 0           |             |          |

Karush-Kuhn-Tucker optimality conditions:

KKT.PE: max.abs.err = 1.78e-15 on row 3  
max.rel.err = 7.72e-17 on row 3  
High quality

KKT.PB: max.abs.err = 0.00e+00 on row 0  
max.rel.err = 0.00e+00 on row 0

**GLPK Solver Output:**

```
>C:\gusek_0-2-24\gusek\glpsol.exe --c
GLPSOL: GLPK LP/MIP Solver, v4.65
Parameter(s) specified in the command
--cover --clique --gomory --mir -m 1
Reading model section from ligas.mod..
Reading data section from ligas.mod...
58 lines were read
Generating Lucro...
Generating Mat_Prime...
Model has been successfully generated
GLPK Simplex Optimizer, v4.65
4 rows, 2 columns, 8 non-zeros
Preprocessing...
3 rows, 2 columns, 6 non-zeros
Scaling...
A: min|aij| = 1.000e+00 max|aij| =
Problem data seem to be well scaled
Constructing initial basis...
Size of triangular part is 3
* 0: obj = -0.000000000e+00 inf =
* 3: obj = 3.066666667e+02 inf =
OPTIMAL LP SOLUTION FOUND
Time used: 0.0 secs
Memory used: 0.1 Mb (112650 bytes)

-----
Liga 1 - Quantidade: 6.333333
Liga 2 - Quantidade: 2.333333
-----
Lucro Total: 306.6667

Model has been successfully processed
Writing basic solution to 'ligas.out'.
>Exit code: 0 Time: 4.878
```

Cursor: line[17] col[1] Selection: [0]lines [0]chars [INS] [CR+LF] File: ligas.out, 39 lines

No arquivo **ligas.out** temos o tamanho do problema, o status da resolução e o valor da Função Objetivo.

**Linha 10: pode-se observar a Função Objetivo -> Lucro = 306.6666667 (novamente)**

Restrições:

**Linha 11: matéria prima 1 -> foram utilizadas 15 (Activity) das 15 tons disponíveis (Upper bound)**

**Linha 12: matéria prima 2 -> foram utilizadas 11 (Activity) das 11 tons disponíveis (Upper bound)**

Estas duas restrições são ativas, ou seja, elas restringem a melhora na FO.

**Linha 13: matéria prima 3 -> foram utilizadas 13.333 (Activity) das 14 tons disponíveis (Upper bound)**

Esta última restrição não é ativa pois ainda há uma sobra deste recurso.

Solução ótima.

**Linha 17:  $x[1] = 6.333$**

**Linha 18:  $x[2] = 2.333$**

|    | 1 ligas.mod  | 2 ligas.lp | 3 ligas.out |
|----|--|------------|-------------|
| 1  | Problem: ... ligas   |            |             |
| 2  | Rows: ... 4  |            |             |
| 3  | Columns: ... 2   |            |             |
| 4  | Non-zeros: ... 8   |            |             |
| 5  | Status: ... OPTIMAL  |            |             |
| 6  | Objective: ... Lucro = 306.6666667 (MAXimum)   |            |             |
| 7  |  |            |             |
| 8  | ... No. ... Row name ... St ... Activity ... Lower bound ... Upper bound ... Marginal    |            |             |
| 9  | -----  |            |             |
| 10 | ... 1 Lucro ... B ... 306.667 ...  |            |             |
| 11 | ... 2 Mat_Prime[1] NU ... 15 ... 15 ... 3.33333  |            |             |
| 12 | ... 3 Mat_Prime[2] NU ... 11 ... 11 ... 23.3333  |            |             |
| 13 | ... 4 Mat_Prime[3] B ... 13.3333 ... 14  |            |             |
| 14 |  |            |             |
| 15 | ... No. ... Column name ... St ... Activity ... Lower bound ... Upper bound ... Marginal |            |             |
| 16 | -----  |            |             |
| 17 | ... 1 x[1] ... B ... 6.33333 ... 0   |            |             |
| 18 | ... 2 x[2] ... B ... 2.33333 ... 0   |            |             |

## Modelo Compacto do **GUSEK** (cont.)

Uma outra funcionalidade do Gusek é a geração do **Modelo Explícito** que será resolvido tendo como base o **Modelo Compacto** e os respectivos dados de entrada.

Esta opção só faz sentido para modelos implementados na **forma compacta**, embora possa ser invocado para modelos explícitos.

Ao acessar a opção **Tools -> Build Cplex LP** será gerado o modelo explícito para os dados de entrada, no <arquivo.lp>

Este modelo pode ser usado para tirar erros de lógica do modelo compacto e dos dados de entrada.



## Geração do *Modelo Explícito* a partir dos *dados entrada* + *Modelo Compacto*

The screenshot shows the GUSEK software interface. The 'Tools' menu is open, displaying various options for compiling and solving the model. The 'Build Cplex LP' option is highlighted. The terminal window on the right shows the output of the GLPSOL solver, indicating a successful solution.

**Tools Menu:**

- Compile (Ctrl+F7)
- Build (F7)
- Go (F5)
- Stop Executing (Ctrl+Break)
- Build GLPK LP/MIP (Ctrl+1)
- Build Cplex LP (Ctrl+2)**
- Build MPS (Ctrl+3)
- Use External .dat (Ctrl+4)
- Clear Extra .dat Files (Ctrl+7)
- ☒ Generate Output File on Go (Ctrl+8)
- Generate LP Sensitivity Analysis (Ctrl+9)
- ☒ Use Improved MILP (All Cuts) (Ctrl+Shift+1)
- Use Free MPS Format (Ctrl+Shift+2)
- Open Prompt at File Path (Ctrl+Shift+6)
- Open Containing Folder (Ctrl+Shift+7)
- Next Message (F4)
- Previous Message (Shift+F4)
- Clear Output (Shift+F5)
- Switch Pane (Ctrl+F6)
- Windows Integration >

**Terminal Output:**

```
>C:\gusek_0-2-24\gusek\gl  
GLPSOL: GLPK LP/MIP Solve  
Parameter(s) specified in  
--check --cover --clique  
Reading model section from  
58 lines were read  
Generating Lucro...  
Generating Mat_Prima...  
Model has been successful  
Writing problem data to '  
11 lines were written  
--- Problem Characteristi  
Number of rows  
Number of columns  
Number of non-zeros (matr  
Number of non-zeros (objr  
>Exit code: 0 Time: 0.
```

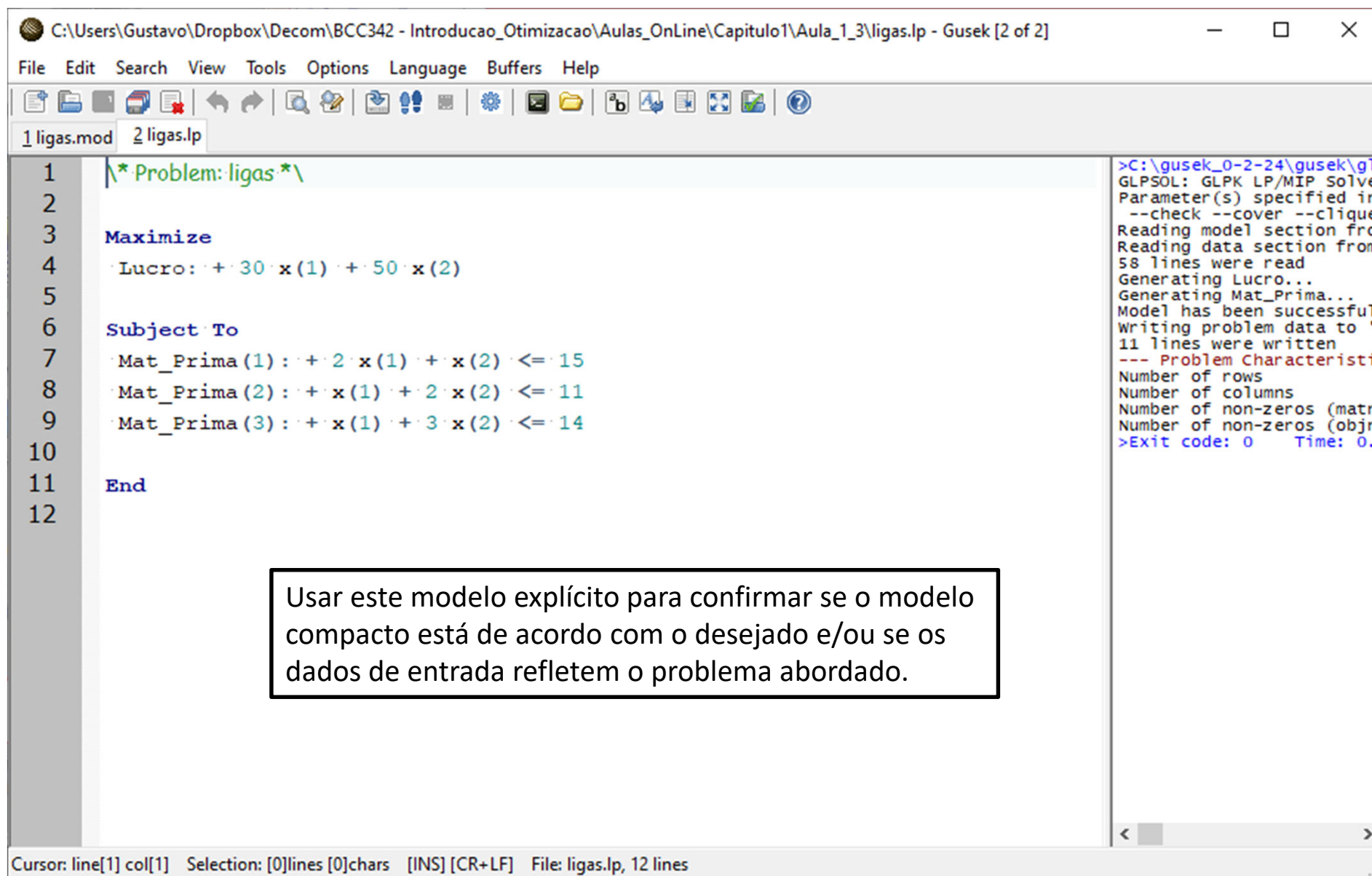
**File: ligas.mod**

```
1 #conjuntos de in  
2 set MP:= {1..3  
3 set Ligas:= {  
4  
5 #parâmetros do  
6 param PV{j in  
7 param disp{MP  
8 param matriz{  
9  
10 #variáveis de de  
11 var x{j in Lig  
12  
13 #função objetivo  
14 maximize Lucro  
15  
16 #restrições  
17 s.t. Mat_Prime  
18 solve;  
19  
20 printf "\n---  
21 printf {i in L  
22 printf "-----  
23 printf "Lucro  
24  
25 #entrada de dade  
26 data;  
27 param PV :=
```

Cursor: line[10] col[22] Selection: [0]lines [0]chars [INS] [CR+LF] File: ligas.mod, 59 lines



## Geração do *Modelo Explícito* a partir dos *dados entrada* + *Modelo Compacto*



The screenshot shows a window titled "C:\Users\Gustavo\Dropbox\Decom\BCC342 - Introducao\_Otimizacao\Aulas\_OnLine\Capitulo1\Aula\_1\_3\ligas.lp - Gusek [2 of 2]". The window contains a text editor on the left and a command prompt on the right.

**Text Editor Content (ligas.lp):**

```
1  \* Problem: ligas*\n2\n3  Maximize\n4    Lucro: + 30 x (1) + 50 x (2)\n5\n6  Subject To\n7    Mat_Prima(1) : + 2 x (1) + x (2) <= 15\n8    Mat_Prima(2) : + x (1) + 2 x (2) <= 11\n9    Mat_Prima(3) : + x (1) + 3 x (2) <= 14\n10\n11 End\n12
```

**Command Prompt Output:**

```
>C:\gusek_0-2-24\gusek\gl\nGLPSOL: GLPK LP/MIP Solve\nParameter(s) specified in\n--check --cover --clique\nReading model section fr\nReading data section fro\n58 lines were read\nGenerating Lucro...\nGenerating Mat_Prima...\nModel has been successfu\nWriting problem data to '\n11 lines were written\n--- Problem Characteristi\nNumber of rows\nNumber of columns\nNumber of non-zeros (matr\nNumber of non-zeros (objr\n>Exit code: 0    Time: 0.
```

**Text Box:**

Usar este modelo explícito para confirmar se o modelo compacto está de acordo com o desejado e/ou se os dados de entrada refletem o problema abordado.

**Status Bar:** Cursor: line[1] col[1] Selection: [0]lines [0]chars [INS] [CR+LF] File: ligas.lp, 12 lines

Fazer o Modelo no Gusek

M 3.1 - O Problema da Fábrica de Móveis

|                  | Escritivaninha | Mesa    | Armário  | Prateleira | Disponibilidade |
|------------------|----------------|---------|----------|------------|-----------------|
| Tábua            | 1              | 2       | 1        | 4          | 250             |
| Prancha          | 0              | 1       | 3        | 2          | 600             |
| Painéis          | 3              | 2       | 4        | 0          | 500             |
| Valor de revenda | \$100,00       | \$80,00 | \$120,00 | \$20,00    |                 |

**Tarefa:** Desenvolver um modelo no Gusek que maximize a receita com a venda dos móveis utilizando o modelo compacto.

## M 3.2 - Empresa de manufatura com 4 produtos

### Modelo compacto

|                                      | Produtos |      |      |      |
|--------------------------------------|----------|------|------|------|
|                                      | I        | II   | III  | IV   |
| Potencial máximo de venda (unid/mês) | 70       | 60   | 40   | 20   |
| Lucro (\$/unid)                      | 10,0     | 8,00 | 9,00 | 7,00 |

| Máq. | tempo disp.<br>h/mês |
|------|----------------------|
| M1   | 80                   |
| M2   | 20                   |
| M3   | 40                   |

| Mão de obra | tempo disp.<br>homens-h/mês |
|-------------|-----------------------------|
| MO1         | 120                         |
| MO2         | 160                         |

número de máq-hora por unidade de cada produto

| Máq | Produtos |    |     |    |
|-----|----------|----|-----|----|
|     | I        | II | III | IV |
| M1  | 5        | 4  | 8   | 9  |
| M2  | 2        | 6  | --- | 8  |
| M3  | 3        | 4  | 6   | 2  |

número de homens-hora por unidade de cada produto

| MDO | Produtos |    |     |    |
|-----|----------|----|-----|----|
|     | I        | II | III | IV |
| MO1 | 2        | 4  | 2   | 8  |
| MO2 | 7        | 3  | --- | 7  |

## MODELO "AMIGÁVEL - PORTUGOL"

### Conjuntos de índices:

Prods := {P1, P2, P3, P4}; Maq := {M1, M2, M3}; Mdo := {MO1, MO2};  
#é possível utilizar literais para os índices das estruturas

### Parâmetros:

L[j], j ∈ Prods; //lucro por unidade do produto j

L := (10, 8, 9, 7);

Vd[j], j ∈ Prods; //potencial de venda do produto j

Vd := (70, 60, 40, 20);

Disp\_Mq[i], i ∈ Maq; //disp. da maquina i

Disp\_Mq := (80, 20, 40);

Disp\_Md[i], i ∈ Mdo; //disp. da mão de obra i

Disp\_Md := (120, 160);

Matriz1[i,j], i ∈ Maq, j ∈ Prods; //consumo da maq. i por produto j

Matriz2[i,j], i ∈ Mdo, j ∈ Prods; //consumo da mdo. i por produto j

Matriz1 = (5, 4, 8, 9,  
          2, 6, 0, 8,  
          3, 4, 6, 2);  
Matriz2 = (2, 4, 2, 8,  
          7, 3, 0, 7);

### Variáveis de decisão:

X[j], j ∈ Prods, inteiro, >=0; //quantidade produzida do produto j

**Variáveis de decisão:**

$X[j]$ ,  $j \in Prods$ , inteiro,  $\geq 0$ ; //quant. produzida do prod.  $j$

**Função objetivo:**

$$\text{MAX } Z = \sum_{j \in Prods} L[j] * X[j];$$

**Restrições:**

$$Mqs: \quad \sum_{j \in Prods} Matriz1[i,j] * X[j] \leq Disp\_Mq[i], \quad \forall i \in Maq;$$

$$Mds: \quad \sum_{j \in Prods} Matriz2[i,j] * X[j] \leq Disp\_Md[i], \quad \forall i \in Mdo;$$

$$\text{Vendas:} \quad X[j] \leq Vd[j] \quad \forall j \in Prods;$$

A partir de agora, você deve fazer o modelo compacto para todos os problemas apresentados.

## M 3.2 - Empresa de manufatura com 4 produtos

### Modelo compacto

|                                      | Produtos |      |      |      |
|--------------------------------------|----------|------|------|------|
|                                      | I        | II   | III  | IV   |
| Potencial máximo de venda (unid/mês) | 70       | 60   | 40   | 20   |
| Lucro (\$/unid)                      | 10,0     | 8,00 | 9,00 | 7,00 |

| Máq. | tempo disp.<br>h/mês |
|------|----------------------|
| M1   | 80                   |
| M2   | 20                   |
| M3   | 40                   |

| Mão de obra | tempo disp.<br>homens-h/mês |
|-------------|-----------------------------|
| MO1         | 120                         |
| MO2         | 160                         |

número de máq-hora por unidade de cada produto

| Máq | Produtos |    |     |    |
|-----|----------|----|-----|----|
|     | I        | II | III | IV |
| M1  | 5        | 4  | 8   | 9  |
| M2  | 2        | 6  | --- | 8  |
| M3  | 3        | 4  | 6   | 2  |

número de homens-hora por unidade de cada produto

| MDO | Produtos |    |     |    |
|-----|----------|----|-----|----|
|     | I        | II | III | IV |
| MO1 | 2        | 4  | 2   | 8  |
| MO2 | 7        | 3  | --- | 7  |

# Modelo Compacto do **GUSEK**

```
set P := {1..4};  
set Maq := {1..3};  
set Mdo := {1..2};
```

## Modelo Compacto do **GUSEK**

```
set P := {1..4};  
set Maq := {1..3};  
set Mdo := {1..2};  
  
param disp_maq{i in Maq};  
param disp_mdo{i in Mdo};  
param lucro{j in P};  
param pot_vendas{j in P};
```



# Modelo Compacto do **GUSEK**

```
set P := {1..4};  
set Maq := {1..3};  
set Mdo := {1..2};  
  
param disp_maq{i in Maq};  
param disp_mdo{i in Mdo};  
param lucro{j in P};  
param pot_vendas{j in P};  
  
param matriz1{i in Maq, j in P};  
param matriz2{i in Mdo, j in P};
```

# Modelo Compacto do GUSEK

```
set P := {1..4};  
set Maq := {1..3};  
set Mdo := {1..2};  
  
param disp_maq{i in Maq};  
param disp_mdo{i in Mdo};  
param lucro{j in P};  
param pot_vendas{j in P};  
  
param matriz1{i in Maq, j in P};  
param matriz2{i in Mdo, j in P};  
  
var x{p in P}, >= 0, integer; # restrição de integralidade de x
```

# Modelo Compacto do GUSEK

```
set P := {1..4};
set Maq := {1..3};
set Mdo := {1..2};

param disp_maq{i in Maq};
param disp_mdo{i in Mdo};
param lucro{j in P};
param pot_vendas{j in P};

param matriz1{i in Maq, j in P};
param matriz2{i in Mdo, j in P};

var x{p in P}, >= 0, integer; # restrição de integralidade de x

maximize lucro: sum{p in P} lucro[p] * x[p];
```

# Modelo Compacto do GUSEK

```
set P := {1..4};
set Maq := {1..3};
set Mdo := {1..2};

param disp_maq{i in Maq};
param disp_mdo{i in Mdo};
param lucro{j in P};
param pot_vendas{j in P};

param matriz1{i in Maq, j in P};
param matriz2{i in Mdo, j in P};

var x{p in P}, >= 0, integer; # restrição de integralidade de x

maximize lucro: sum{p in P} lucro[p] * x[p];

restMaq{m in Maq}: sum{p in P} matriz1[m, p] * x[p] <= disp_maq[m];
```

# Modelo Compacto do GUSEK

```
set P := {1..4};
set Maq := {1..3};
set Mdo := {1..2};

param disp_maq{i in Maq};
param disp_mdo{i in Mdo};
param lucro{j in P};
param pot_vendas{j in P};

param matriz1{i in Maq, j in P};
param matriz2{i in Mdo, j in P};

var x{p in P}, >= 0, integer; # restrição de integralidade de x

maximize lucro: sum{p in P} lucro[p] * x[p];

restMaq{m in Maq}: sum{p in P} matriz1[m, p] * x[p] <= disp_maq[m];

restMdo{i in Mdo}: sum{j in P} matriz2[i, j] * x[j] <= disp_mdo[i];
```

# Modelo Compacto do GUSEK

```
set P := {1..4};
set Maq := {1..3};
set Mdo := {1..2};

param disp_maq{i in Maq};
param disp_mdo{i in Mdo};
param lucro{j in P};
param pot_vendas{j in P};

param matriz1{i in Maq, j in P};
param matriz2{i in Mdo, j in P};

var x{p in P}, >= 0, integer; # restrição de integralidade de x

maximize lucro: sum{p in P} lucro[p] * x[p];

restMaq{m in Maq}: sum{p in P} matriz1[m, p] * x[p] <= disp_maq[m];

restMdo{i in Mdo}: sum{j in P} matriz2[i, j] * x[j] <= disp_mdo[i];

restProds{j in P}: x[j] <= pot_vendas[j];
```

data;

param disp\_maq := 1 80 3 40 2 20;

param disp\_mdo := 1 120  
2 160;

param lucro := 1 10  
2 8  
3 9  
4 7;

param pot\_vendas := 1 70  
2 60  
3 40  
4 20;

param matriz1: 1 2 3 4 :=  
1 5 4 8 9  
2 2 6 0 8  
3 3 4 6 2

param matriz2: 1 2 3 4 :=  
1 2 4 2 8  
2 7 3 0 7

Fazer o Modelo no Gusek

M 3.1 - O Problema da Fábrica de Móveis

|                  | Escritivaninha | Mesa    | Armário  | Prateleira | Disponibilidade |
|------------------|----------------|---------|----------|------------|-----------------|
| Tábua            | 1              | 2       | 1        | 4          | 250             |
| Prancha          | 0              | 1       | 3        | 2          | 600             |
| Painéis          | 3              | 2       | 4        | 0          | 500             |
| Valor de revenda | \$100,00       | \$80,00 | \$120,00 | \$20,00    |                 |

**Tarefa:** Desenvolver um modelo no Gusek que maximize a receita com a venda dos móveis utilizando o modelo compacto.