Trabalho Computacional de Programação Linear – 2016/1
Módulo 2: Método de 2 Fase
Data de Entrega: 30/05/2016
Profª. Maria Cristina Rangel

Importante:

- **Enviar o arquivo fonte para crangel@inf.ufes.br**
  utilizando o subject: Trabalho Computacional Módulo 2:nome1:nome2
- **O trabalho pode ser feito em dupla**

Implementar o **Método de 2 Fases** para resolver um Problema de Programação Linear (PPL):

minimize *z*= **cx**
sujeito a A**x** ≤ ou ≥ ou = **b,** $b \in R^n, b \geq 0$
x ≥ 0

onde $x \in R^n, c \in R^n, b \in R^n \, e A_{mxn}$

O programa deverá ter como dados de entrada:

1. uma chave para mostrar que precisará da primeira fase do algoritmo. Definir chave=1: método de 2 fases e chave=2: apenas o simplex.
2. os dados do PPL devem ser fornecidos em forma de matriz.
3. para construir a matriz, o PPL deve ser escrito na forma padrão. Neste caso, a matriz deve conter a linha referente à função objetivo artificial **za** e as colunas relativas às variáveis artificiais **xiª**.

Como saída de dados o programa deverá informar:

1. o valor de **za*** informando se existe ou não solução para o PPL.
2. caso não haja solução, informar que o conjunto de soluções viáveis é vazio.
3. o valor de **z***, respectivo **x*** e se é solução **única** ou **múltipla**.
4. caso não haja solução, informar se **z = -inf** (infinito).
5. imprimir o quadro tableau a cada iteração para mostrar as trocas de variáveis da base.

Exemplo 1:

min z= 2x1 - 4x2 + 3x3
sa x1 + x2 + x3 ≤ 4
x2 - 3x3 ≤ 3
6x1 - x2 + x3 ≥ 4
x1, x2, x3 ≥ 0

Escrevendo na Forma Padrão para construir a matriz de dados de entrada (necessidade de uma **x1ₐ** ≥ 0):

min z= 2x1 - 4x2 + 3x3
sa x1 + x2 + x3 + x4 = 4
x2 - 3x3 + x5 = 3
6x1 - x2 + x3 - x6 = 4
x1, x2, x3, x4, x5, x6 ≥ 0

Então, temos uma função **za** = $\mathbf{x1^a}$

$$\min za = x1_a$$
$$\min z = 2x1 - 4x2 + 3x3$$
$$\begin{array}{lrcl} \text{sa} & x1 + x2 + x3 + x4 & = & 4 \\ & x2 - 3x3 + x5 & = & 3 \\ & 6x1 - x2 + x3 - x6 + x1^a & = & 4 \end{array}$$
$$\mathbf{x1, x2, x3, x4, x5, x6, x1^a \geq 0}$$

Entrada:

dimensões da matriz **(m+1+1)** e **(n+m+1+1)**
matriz **A(m+1+1)x(n+m+1+1),** onde **(m+1+1)** = fç obj artificial + fç obj + 3 restrições
e **(n+m+1+1)** = 3 vars de naturais + 3 vars de folga + 1 var artificial + termo independente

```
1                   // utilizar a primeira fase do algoritmo
5 8                 // m+1+1 e n+m+1+1
0  0  0 0 0  0 1  0    //vetor custo da função objetivo artificial
2 -4  3 0 0  0 0  0    //vetor custo e espaços para valor da função z e za
1  1  1 1 0  0 0  4       //matriz e termo independente
0  1 -3 0 1  0 0  3
6 -1  1 0 0 -1 1 4
```

Saída: -9.14286; x1 = 1.14286, x2 = 2.85714, x3 = 0

z* = -9.143 x*=(1.143 2.857 0 0 0.143 0) solução única (aqui não imprimi o quadro ótimo!!! mas é para imprimir)

Obs.: não esquecer que restrições de igualdade possuem apenas variáveis artificiais.