EP2 Otimização Linear

Bruno Hideki Akamine

NUSP: 11796322

1 Sobre o Exercício-Programa

Neste exercício-programa estaremos implementando as duas fases do método Simplex, dessa vez nosso programa pode receber qualquer tipo de problema, ou melhor, não precisamos supor que o problema é viável e contém somente soluções não degeneradas. Implementamos duas versões do método simplex : o método simplex revisado e o método *full tableau*.

2 Implementações

Como no relatório do primeiro exercício-programa já explicamos o funcionamento da segunda fase do método Simplex tanto do revisado quanto do *full tableau*, iremos nos focar em explicar o pivoteamento e a primeira fase. No programa feito usamos a mesma estrutura de loop em ambos os métodos para a segunda fase.

2.1 Método Simplex Revisado

2.1.1 Principais partes do programa

Vamos começar falando um pouco sobre a regra de pivoteamento aplicada e como se deu sua implementação nessa versão do método Simplex. Neste exercício-programa foi pedido para implementar a regra de pivoteamento do menor índice. No nosso programa aplicamos essa regra em duas ocasiões: para a escolha da variável de entrada e para a escolha da variável de saída. No caso da escolha da variável de entrada basta checarmos qual dos índices é menor, como imprimimos todos os índices não básicos com seus custos reduzidos para a melhor visualização do método e do seu funcionamento não mantemos o vetor de variáveis não basicas (`variaveis_Nbasicas`) ordenado e portanto a velocidade do método pode ser um pouco comprometida para alguns casos específicos já que não economizamos cálculos. Também é aplicado o pivoteamento para a escolha de variável de saída, porém como precisamos escolher o menor θ possível, só aplicamos essa regra quando temos θ's pequenos que se coincidem e então escolhemos a variável com menor índice(A regra de pivoteamento é usada várias vezes durante o método, vamos deixar aqui as linhas correspondentes à primeira utilização para a escolha da variável de entrada e de saída. Variável de entrada - Linhas 39 a 45; Variável de saída: Linhas 58 a 71).

Tendo ja falado sobre a regra de pivoteamento utilizado, podemos começar a falar sobre a implementação da primeira fase do método Simplex revisado. O objetivo da primeira fase é encontrar uma solução básica viável inical, decidir se o problema é viável e retirar restrições redundantes. Para encontrar a primeira solução básica viável, resolvemos um problema auxiliar da seguinte forma:

$$\begin{aligned} min\ & y \\ sujeito\, a\ & Ax + y = b \\ & x \geq 0 \\ & y \geq 0 \end{aligned}$$

Sabemos que uma solução inicial para esse problema auxiliar é (x,y) onde x é um vetor nulo e y = b. Para fazer essa implementação concatenamos à matriz A uma matriz identidade de dimensão $m \times m$ onde m é o número de restrições, ou melhor, temos a matriz auxiliar $A' = [A|I_{m\times m}]$. Assim, realizamos a iteração do método simplex usual para esse problema auxiliar. Ao fim da iteração, se a solução ótima tiver custo maior que zero, concluimos que o problema é inviável. Se o custo for igual a zero, porém ainda tivermos variáveis artificiais na base, retiramos elas da seguinte forma: Olhamos para a linha correspondente a variável artificial da matriz $B^{-1}A$ se todas as entradas da linha forem iguais a zero, conclui-se que temos uma restrição redundante e portanto podemos reduzir o número de restrições para $m-1$(Linhas 151 a 157). Para isso, retiramos a linha e coluna correspondente a variável artificial na inversa B^{-1}, e com isso conseguimos reduzir em uma dimensão a matriz quadradada $m \times m$ para $m - 1 \times m - 1$. Caso haja alguma entrada diferente de zero, realizamos as operações elementares de linhas usuais para que a variável correspondente a essa entrada entre na base e a variável artifical saia da base. Assim a primeira fase termina com uma solução basica viável inicial para o problema original, ou conclui que o problema é inviável. Para inciar a segunda fase basta diminuir a dimensão da solução (x, y) de $1 \times m + n$ para $1 \times n$ retirando a parte correspondente ao y que sabemos que é igual ao vetor nulo (pois pela contrução do nosso algoritmo ao fim da primeira fase todas as variáveis artificiais não são variáveis básicas). Assim começamos a iteração da segunda fase do método Simplex que foi o foco do primeira exercício-programa.

2.1.2 Exeplos de execução

1. **Problema inviável**

   O problema que usaremos de exemplo é:

- $A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 7 & 5 & 3 & 2 \\ 3 & 5 & 10 & 15 \end{pmatrix}$
- $b = (15, 120, 100)$
- $c = (-4, -5, -9, -11)$
- $m = 3$
- $n = 4$

Ou seja, temos o problema:

$$\begin{aligned} min \;\; & -4x_1 - 5x_2 - 9x_3 - 11x_4 \\ sujeito\,a \;\; & x_1 + x_2 + x_3 + x_4 = 15 \\ & 7x_1 + 5x_2 + 3x_3 + 2x_4 = 120 \\ & 3x_1 + 5x_2 + 10x_3 + 15x_4 = 100 \end{aligned}$$

Temos portanto a saída do método:

```
----------      Fase 1      ----------
***********Iteração 0***************
Indices das variáveis básicas:(indice - valor associado)
5  15.00000
6  120.00000
7  100.00000

Valor da função objetiva: 235.00000

Matriz inversa:
Diagonal Matrix

   1   0   0
   0   1   0
   0   0   1

Vetor p:
   1
   1
   1
Custos reduzidos:
1  -11.00000
2  -11.00000
3  -14.00000
4  -18.00000
```

```
 Entra na base: 1

Direção:
5  1.00000
6  7.00000
7  3.00000

Theta*
  15.00000

Sai da base : 5

***********Iteração 1***************
Indices das variáveis básicas:(indice - valor associado)
1  15.00000
6  15.00000
7  55.00000

Valor da função objetiva: 70.00000

Matriz inversa:
   1   0   0
  -7   1   0
  -3   0   1

Vetor p:
  -10
    1
    1
Custos reduzidos:
5  11.00000
2  0.00000
3  -3.00000
4  -7.00000

 Entra na base: 3

Direção:
1  1.00000
6  -4.00000
7  7.00000

Theta*
  7.85714

Sai da base : 7
```

```
***********Iteração 2***************
Indices das variáveis básicas:(indice - valor associado)
1  7.14286
6  46.42857
3  7.85714

Valor da função objetiva: 46.42857

Matriz inversa:
   1.4286        0  -0.1429
  -8.7143   1.0000   0.5714
  -0.4286        0   0.1429

Vetor p:
  -8.7143
   1.0000
   0.5714
Custos reduzidos:
5  9.71429
2  0.85714
7  0.42857
4  -1.85714

 Entra na base: 4

Direção:
1  -0.71429
6  1.85714
3  1.71429

Theta*
  4.58333

Sai da base : 3

***********Iteração 3***************
Indices das variáveis básicas:(indice - valor associado)
1  10.41667
6  37.91667
4  4.58333

Valor da função objetiva: 37.91667

Matriz inversa:
   1.2500        0  -0.0833
```

```
  -8.2500   1.0000   0.4167
  -0.2500        0   0.0833

Vetor p:
  -8.2500
   1.0000
   0.4167
Custos reduzidos:
5  9.25000
2  1.16667
7  0.58333
3  1.08333
Problema inviável.
ind = 1
x = NaN
d = NaN
```

Vamos comentar um pouco sobre a iteração 0 e a iteração 3 da saída. Na iteração zero estamos comesçando a primeira fase do método Simplex com solução (0, 0, 0, 0, 15, 120, 100) para o problema auxiliar como explicado acima e vemos que de fato a matriz inicial é a matriz identidade de tamanho $m \times m$, que é igual a sua inversa. Assim, ao calcularmos os custos reduzidos notamos que todas as variáveis não básicas poderiam entrar na base, porém como estamos usando o pivoteamento de menor índice, escolhemos x_1 como pivot de entrada, e para a escolha de saída é escolhido a variável x_5. Durante as outras iterações é possível perceber que a regra de pivoteamento esta sendo feita corretamente, pois entre os índices com custo negativo, sempre é escolhido o menor índice. Agora pulamos para a última iteração, a iteração 3, nessa iteração vemos que nenhum custo reduzido é negativo, portanto conclui-se que estamos em uma solução ótima ao problema auxiliar. Porém checando-se o valor da função objetiva, que é imprimido a cada iteração, vemos que seu valor é maior que zero, portanto o programa conclui que o problema é inviável e acaba o algoritmo, devolvendo `ind = 1` como pedido no enunciado, como não é pedido quais valores a serem retornados às variáveis `x` e `d` para esse caso, devolvemos `NaN` para essas variáveis.

2. **Problema ilimitado**

   O problema que usaremos de exemplo é:

   - $A = \begin{pmatrix} -1 & 1 & 1 & 0 \\ 1 & -2 & 0 & 1 \end{pmatrix}$
   - $b = (1, 2)$
   - $c = (-2, -1, 0, 0)$
   - $m = 2$

- $n = 3$

Ou seja, temos o problema:

$$
\begin{aligned}
min \quad & -2x_1 - x_2 \\
sujeito\, a \quad & -x_1 + x_2 + x_3 = 1 \\
& x_1 - 2x_2 + x_4 = 2
\end{aligned}
$$

Temos portanto a seguinte saída do método:

```
----------      Fase 1      ----------
***********Iteração 0***************
Indices das variáveis básicas:(indice - valor associado)
5  1.00000
6  2.00000

Valor da função objetiva: 3.00000

Matriz inversa:
   1   0
   0   1

Vetor p:
   1
   1
Custos reduzidos:
1  0.00000
2  1.00000
3  -1.00000
4  -1.00000

 Entra na base: 3

Direção:
5  1.00000
6  0.00000

Theta*
  1.00000

Sai da base : 5

***********Iteração 1***************
Indices das variáveis básicas:(indice - valor associado)
```

```
3  1.00000
6  2.00000

Valor da função objetiva: 2.00000

Matriz inversa:
   1   0
   0   1

Vetor p:
   0
   1
Custos reduzidos:
1  -1.00000
2  2.00000
5  1.00000
4  -1.00000

 Entra na base: 1

Direção:
3  -1.00000
6  1.00000

Theta*
  2.00000

Sai da base : 6

***********Iteração 2***************
Indices das variáveis básicas:(indice - valor associado)
3  3.00000
1  2.00000

Valor da função objetiva: 0.00000

Matriz inversa:
   1   1
   0   1

Vetor p:
   0
   0
Custos reduzidos:
6  1.00000
2  0.00000
```

```
5  1.00000
4  0.00000
----------      Fase 2      ----------
***********Iteração 0***************
Indices das variáveis básicas:(indice - valor associado)
3  3.00000
1  2.00000

Valor da função objetiva: -4.00000

Matriz inversa:
   1   1
   0   1

Vetor p:
   0
  -2

Custos reduzidos:
2  -5.00000
4  2.00000

 Entra na base: 2

Direção:
3  -1.00000
1  -2.00000
Problema ilimitado
ind = -1
x = NaN
d =

  -1
  -2
```

Vamos comentar um pouco sobre a última iteração da primeira fase e a única iteração da segunda fase do exemplo acima. Note que na última iteração da primeira fase nenhuma das variáveis básicas corresponde a uma variável artificial, portanto não precisamos retirar nenhuma váriavel artifical da base, além disso, é possível ver que o valor da função objetiva do problema auxiliar ao final da primeira fase é igual a 0, e portanto o programa conclui que a solução ótima atual do problema auxilair também é uma solução básica viável ao problema original, sendo essa a primeira solução utilizada na segunda fase do método. Logo na primeira iteração da segunda fase, após escolher a variável deentrada (x_2) o programa percebe que todas as componentes do vetor de direção são negativas e portanto conclui que o problema é ilimitado e para a execução do programa devolvendo `ind` = `-1` e o vetor de direção em `d`. Como não foi especificado o que devolver em x nesse caso, devolvemos `NaN`.

3. **Problema com solução ótima**

   O problema que usaremos nesse exemplo é:

   - $A = \begin{pmatrix} 1 & 2 & 3 & 0 \\ -1 & 2 & 6 & 0 \\ 0 & 4 & 9 & 0 \\ 0 & 0 & 3 & 1 \end{pmatrix}$
   - $b = (3, 2, 5, 1)$
   - $c = (1110)$
   - $m = 4$
   - $n = 4$

   Ou seja, temos o problema:

$$\begin{aligned} min\ & x_1 + x_2 + x_3 \\ sujeito\,a\ & x_1 + 2x_2 + 3x_3 = 3 \\ & -x_1 + 2x_2 + 6x_3 = 2 \\ & 4x_2 + 9x_3 = 5 \\ & 3x_3 + x_4 = 1 \end{aligned}$$

   Temos portanto a seguinte saída do método:

```
----------      Fase 1      ----------
***********Iteração 0***************
Indices das variáveis básicas:(indice - valor associado)
5  3.00000
6  2.00000
7  5.00000
8  1.00000

Valor da função objetiva: 11.00000

Matriz inversa:
   1   0   0   0
   0   1   0   0
   0   0   1   0
   0   0   0   1
```

```
Vetor p:
   1
   1
   1
   1
Custos reduzidos:
1  0.00000
2  -8.00000
3  -21.00000
4  -1.00000

 Entra na base: 2

Direção:
5  2.00000
6  2.00000
7  4.00000
8  0.00000

Theta*
  1.00000

Sai da base : 6

***********Iteração 1***************
Indices das variáveis básicas:(indice - valor associado)
5  1.00000
2  1.00000
7  1.00000
8  1.00000

Valor da função objetiva: 3.00000

Matriz inversa:
   1.0000  -1.0000        0        0
        0   0.5000        0        0
        0  -2.0000   1.0000        0
        0        0        0   1.0000

Vetor p:
   1
  -3
   1
   1
Custos reduzidos:
```

```
1  -4.00000
6  4.00000
3  3.00000
4  -1.00000

 Entra na base: 1

Direção:
5  2.00000
2  -0.50000
7  2.00000
8  0.00000

Theta*
  0.50000

Sai da base : 5

***********Iteração 2***************
Indices das variáveis básicas:(indice - valor associado)
1  0.50000
2  1.25000
7  0.00000
8  1.00000

Valor da função objetiva: 1.00000

Matriz inversa:
   0.5000  -0.5000        0        0
   0.2500   0.2500        0        0
  -1.0000  -1.0000   1.0000        0
        0        0        0   1.0000

Vetor p:
  -1
  -1
   1
   1
Custos reduzidos:
5  2.00000
6  2.00000
3  -3.00000
4  -1.00000

 Entra na base: 3
```

```
Direção:
1  -1.50000
2  2.25000
7  0.00000
8  3.00000

Theta*
  0.33333

Sai da base : 8

***********Iteração 3***************
Indices das variáveis básicas:(indice - valor associado)
1  1.00000
2  0.50000
7  0.00000
3  0.33333

Valor da função objetiva: 0.00000

Matriz inversa:
   0.5000  -0.5000        0   0.5000
   0.2500   0.2500        0  -0.7500
  -1.0000  -1.0000   1.0000        0
        0        0        0   0.3333

Vetor p:
  -1
  -1
   1
   0
Custos reduzidos:
5  2.00000
6  2.00000
8  1.00000
4  0.00000

    -Retirando a variável artificial 7 da base-

Linha correspondente a variavel 7 da matriz inv * A:
0.00000 0.00000 0.00000 0.00000

Linha 3 da matriz A corresponde a uma restrição redundante, porta
nto vamos retirá-la.
Retiramos também a variável 7 das variáveis básicas.
Matriz inversa:
```

```
   0.5000  -0.5000   0.5000
   0.2500   0.2500  -0.7500
        0        0   0.3333

----------      Fase 2      ----------
***********Iteração 0***************
Indices das variáveis básicas:(indice - valor associado)
1  1.00000
2  0.50000
3  0.33333

Valor da função objetiva: 1.83333

Matriz inversa:
   0.5000  -0.5000   0.5000
   0.2500   0.2500  -0.7500
        0        0   0.3333

Vetor p:
   0.750000
  -0.250000
   0.083333

Custos reduzidos:
4  -0.08333

 Entra na base: 4

Direção:
1  0.50000
2  -0.75000
3  0.33333

Theta*
  1.00000

Sai da base : 3

***********Iteração 1***************
Indices das variáveis básicas:(indice - valor associado)
1  0.50000
2  1.25000
4  1.00000

Valor da função objetiva: 1.75000
```

```
Matriz inversa:
   0.5000  -0.5000        0
   0.2500   0.2500        0
        0        0   1.0000

Vetor p:
   0.7500
  -0.2500
        0

Custos reduzidos:
3  0.25000

Solução ótima encontrada com custo 1.75000:
1  0.50000
2  1.25000
3  0.00000
4  1.00000
ind = 0
x =

   0.5000   1.2500        0   1.0000

d = NaN
```

Vamos comentar um pouco sobre a última iteração da primeira fase e a última iteração da segunda fase. Na última iteração da primeira fase(iteração 3) vemos que a o valor da função objetiva da função auxiliar é 0, além disso, vemos que x_7 está na base, apesar de ser uma variável artifical, portanto o programa deve retirar x_7 da base, ao analisar a linha correspondente a variável 7 da matriz $B^{-1}A$, vemos que não há nenhuma entrada com valor diferente de zero, portantoo programa conclui que existe uma restrição redundante. Note que a matriz inversa final de fato diminui de uma dimensão, ou melhor, foi de dimensão 4×4 para dimensão 3×3. Note que no começo da segunda fase, a solução básica inicial é igual a solução final da primeira fase sem a variável x_7 e com uma dimensão a menos. Na última iteração da segunda fase vemos que nenhum custo reduzido é negativo, portanto o programa conclui que a solução básica viável atual é ótima e devolve `ind` $= 0$ e a solução ótima em x. Como não é especificado o que devolver em `d` nesse caso, devolvemos `NaN`.

2.2 Método *full tableau*

2.2.1 Principais partes do programa

Assim como no método Simplex revisado, nessa implementaçã do *full tableau* estaremos usando a regra de pivoteamento de menor índice. A implementação dessa regra é feita da mesma forma que no outro método, portanto não iremos nos alongar nessa parte.

Como ja discutido anteriormente, o objetivo da primeira fase é encontra uma solução ótima para o seguinte problema:

$$\begin{aligned} min\ & y \\ sujeito\, a\ & Ax + y = b \\ & x \geq 0 \\ & y \geq 0 \end{aligned}$$

Nossa função `[ind x d] = simplex_tab(tableau)` recebe um *tableau* inicial, onde a linha zero contém a o vetor da função objetiva `c` e a coluna zero contém o vetor `b` (Note que a entrada do canto superior esquerdo não contém nada, portanto para os exemplos deixaresmo essa entrada igual a zero por padrão). Assim para transformarmos nosso `tableau` inicial no problema auxiliar armazenamos a linha zero do tableau em uma variável `c`, após isso retiramos a linha zero do e concatenamos a matriz indentidade $I_{m \times m}$. Resultando no seguinte tableau: $[b|A|I_{m \times m}]$. Por fim, colocamos de volta a linha zero, correspondente a um vetor de R^{m+n+1} onde a primeira entrada corresponde a $-\sum_{i=1}^{m} b_i$. As n seguintes entradas correspondem ao vetor - uA onde u é um vetor de R^m composto só de uns e as últimas m entradas são iguais a 0 (Esse processo de transformar o tableau em outro correspondente ao tableau do problema auxiliar é feito nas linhas 273 à 287). Com isso fazemos a iteração usual do *tableau.* Checando a entrada superior esquerda do tableau concluimos se o problema é viável ou não. Caso seja viável, ainda precisamos ver se alguma variável artificial esta presente na base. Caso esteja, então vamos olhar para a linha correspondente a essa variável artificial, caso haja alguma entrada diferente de zero (note que estamos checando somente as entradas correspondentes a variáveis não artificiais), então a variável correspondente a essa coluna entra na base e a variável artifical saia da base. Note que não estamos fazendo nada de diferente do método revisado, a maior diferença é caso todas as variáveis sejam iguais a zero, nesse caso basta retirarmos a linha correspondente a variável artificial do tableau. Ao fim desse processo todo, temos um tableau correspondente a uma solução básica viável ao problema original. Precisamos só "voltar" com o tableau para o início da segunda fase. Para isso, retiramos as últimas m colunas do tableau, também "trocamos" a linha zero do nosso tableau pelo vetor $[-c_b x_b | c' - c_b' B^{-1} A]$, e assim, estamos prontos para iniciar a segunda fase do método Simplex (Lembre que armazenamos a função objetiva do problema inicial na variável c). Essa transformação final corresponde as linhas 389 à 392.

2.2.2 Exemplos de execução

1. **Problema inviável**

   O problema que usaremos de exemplo é:

   - $tableau = \begin{pmatrix} 0 & -4 & -5 & -9 & -11 \\ 15 & 1 & 1 & 1 & 1 \\ 120 & 7 & 5 & 3 & 2 \\ 100 & 3 & 5 & 10 & 15 \end{pmatrix}$

   Ou seja, temos o problema:

$$\begin{aligned} min &\ -4x_1 - 5x_2 - 9x_3 - 11x_4 \\ sujeito\,a &\ x_1 + x_2 + x_3 + x_4 = 15 \\ &\ 7x_1 + 5x_2 + 3x_3 + 2x_4 = 120 \\ &\ 3x_1 + 5x_2 + 10x_3 + 15x_4 = 100 \end{aligned}$$

   Temos portanto a saída do método:

```
----------      Fase 1      ----------
***********Iteração 0***************
Tableau:
  -235   -11   -11   -14   -18     0     0     0
    15     1     1     1     1     1     0     0
   120     7     5     3     2     0     1     0
   100     3     5    10    15     0     0     1


 Entra na base : 1


Theta*
15.00000

Sai da base : 5

***********Iteração 1***************
Tableau:
  -70    0    0   -3   -7   11    0    0
   15    1    1    1    1    1    0    0
   15    0   -2   -4   -5   -7    1    0
   55    0    2    7   12   -3    0    1
```

```
 Entra na base : 3


Theta*
7.85714

Sai da base : 7

***********Iteração 2***************
Tableau:
 Columns 1 through 6:

  -46.4286         0    0.8571         0   -1.8571    9.7143
    7.1429    1.0000    0.7143         0   -0.7143    1.4286
   46.4286         0   -0.8571         0    1.8571   -8.7143
    7.8571         0    0.2857    1.0000    1.7143   -0.4286

 Columns 7 and 8:

         0    0.4286
         0   -0.1429
    1.0000    0.5714
         0    0.1429


 Entra na base : 4


Theta*
4.58333

Sai da base : 3

***********Iteração 3***************
Tableau:
 Columns 1 through 6:

  -37.9167         0    1.1667    1.0833         0    9.2500
   10.4167    1.0000    0.8333    0.4167         0    1.2500
   37.9167         0   -1.1667   -1.0833   -0.0000   -8.2500
    4.5833         0    0.1667    0.5833    1.0000   -0.2500

 Columns 7 and 8:

         0    0.5833
         0   -0.0833
```

```
    1.0000    0.4167
         0    0.0833

Problema inviável.
ans = 1
```

Vamos analisar a primeira e última iteração do método. Note que no *tableua* da primeira iteração de fato tivemos o aumento da dimensão de $(m+1) \times (n+1)$ para $(m+1) \times (n+m+1)$. Também temos que na linha zero todas as variáveis não-básicas tem custo negativo, portanto, utilizando a regra de pivoteamento de menor índice escolhemos a variável x_1 para entrar na base. Após isso, analisamos a coluna correspondente a x_1 e é escolhido a variável x_5 para sair da base. Indo agora para a última iteração, note que a entrada do canto superior esquerdo do *tableua* esta menor do que zero e não há nenhuma variável com custo negativo, portanto o problema auxiliar esta com uma solução ótima maior que zero, com isso o programa conclui que o problema é inviável.

2. **Problema ilimitado**

   O problema que usaremos de exemplo é:

   - $tableau = \begin{pmatrix} 0 & -2 & -1 & 0 & 0 \\ 1 & -1 & 1 & 1 & 0 \\ 2 & 1 & -2 & 0 & 1 \end{pmatrix}$

   Ou seja, temos o problema:

$$\begin{aligned} min \quad & -2x_1 - x_2 \\ sujeito\, a \quad & -x_1 + x_2 + x_3 = 1 \\ & x_1 - 2x_2 + x_4 = 2 \end{aligned}$$

   Temos portanto a seguinte saída do método:

```
tableau =

   0  -2  -1   0   0
   1  -1   1   1   0
   2   1  -2   0   1

----------      Fase 1      ----------
***********Iteração 0***************
Tableau:
```

```
  -3   0   1  -1  -1   0   0
   1  -1   1   1   0   1   0
   2   1  -2   0   1   0   1


 Entra na base : 3


Theta*
1.00000

Sai da base : 5

***********Iteração 1***************
Tableau:
  -2  -1   2   0  -1   1   0
   1  -1   1   1   0   1   0
   2   1  -2   0   1   0   1


 Entra na base : 1


Theta*
2.00000

Sai da base : 6

***********Iteração 2***************
Tableau:
   0   0   0   0   0   1   1
   3   0  -1   1   1   1   1
   2   1  -2   0   1   0   1


Tableau final da fase 1:
   0   0   0   0   0   1   1
   3   0  -1   1   1   1   1
   2   1  -2   0   1   0   1

----------      Fase 2      ----------
***********Iteração 0***************

Tableau:
   4   0  -5   0   2
   3   0  -1   1   1
```

```
   2   1  -2   0   1

 Entra na base: 2

Problema ilimitado
ind = -1
x = NaN
d =

  -1
  -2
```

Vamos explicar brevemente a última iteração da primeira fase, e analisar a única iteração da segunda fase. Na última iteração do método, podemos perceber que nenhuma entrada da linha zero é menor que zero, portanto estamos em uma solução ao problema auxiliar, como a entrada do canto superior esquerda é igual a zero, o problema é viável. Após isso o nosso programa checa as variáveis da base para ver se há alguma variável artificil, como não há nenhuma ele printa o *tableau* final da primeira fase. Note que de fato, o número de colunas do tableau da primeira iteração da segunda fase tem menos $m = 2$ colunas e somente vemos mudança correspondente a linha zero de um *tableau* a outro. Note que na única iteração da segunda fase, a única variável com custo negativo é x_2, porém ao analisarmos sua coluna vemos que não há nenhuma entrada positiva e portanto o programa conclui que o problema é ilimitado.

3. **Problema com solução ótima**

   O problema que usaremos nesse exemplo é:

   - $tableau = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 3 & 1 & 2 & 3 & 0 \\ 2 & -1 & 2 & 6 & 0 \\ 5 & 0 & 4 & 9 & 0 \\ 1 & 0 & 0 & 3 & 1 \end{pmatrix}$

   Ou seja, temos o problema:

$$
\begin{aligned}
min\ & x_1 + x_2 + x_3 \\
sujeito\, a\ & x_1 + 2x_2 + 3x_3 = 3 \\
& -x_1 + 2x_2 + 6x_3 = 2 \\
& 4x_2 + 9x_3 = 5 \\
& 3x_3 + x_4 = 1
\end{aligned}
$$

Temos portanto a seguinte saída do método:

```
----------      Fase 1      ----------
***********Iteração 0***************
Tableau:
  -11    0   -8  -21   -1    0    0    0    0
    3    1    2    3    0    1    0    0    0
    2   -1    2    6    0    0    1    0    0
    5    0    4    9    0    0    0    1    0
    1    0    0    3    1    0    0    0    1


 Entra na base : 2


Theta*
1.00000

Sai da base : 6

***********Iteração 1***************
Tableau:
 Columns 1 through 7:

  -3.0000  -4.0000        0   3.0000  -1.0000        0   4.0000
   1.0000   2.0000        0  -3.0000        0   1.0000  -1.0000
   1.0000  -0.5000   1.0000   3.0000        0        0   0.5000
   1.0000   2.0000        0  -3.0000        0        0  -2.0000
   1.0000        0        0   3.0000   1.0000        0        0

 Columns 8 and 9:

        0        0
        0        0
        0        0
   1.0000        0
        0   1.0000


 Entra na base : 1


Theta*
0.50000

Sai da base : 5
```

```
***********Iteração 2***************
Tableau:
 Columns 1 through 7:

  -1.0000        0        0  -3.0000  -1.0000   2.0000   2.0000
   0.5000   1.0000        0  -1.5000        0   0.5000  -0.5000
   1.2500        0   1.0000   2.2500        0   0.2500   0.2500
        0        0        0        0        0  -1.0000  -1.0000
   1.0000        0        0   3.0000   1.0000        0        0

 Columns 8 and 9:

        0        0
        0        0
        0        0
   1.0000        0
        0   1.0000


 Entra na base : 3


Theta*
0.33333

Sai da base : 8

***********Iteração 3***************
Tableau:
 Columns 1 through 7:

        0        0        0        0        0   2.0000   2.0000
   1.0000   1.0000        0        0   0.5000   0.5000  -0.5000
   0.5000        0   1.0000        0  -0.7500   0.2500   0.2500
        0        0        0        0        0  -1.0000  -1.0000
   0.3333        0        0   1.0000   0.3333        0        0

 Columns 8 and 9:

        0   1.0000
        0   0.5000
        0  -0.7500
   1.0000        0
        0   0.3333
```

```
    -Retirando a variável artificial 7 da base-

Tableau:
 Columns 1 through 7:

        0        0        0        0        0   2.0000   2.0000
   1.0000   1.0000        0        0   0.5000   0.5000  -0.5000
   0.5000        0   1.0000        0  -0.7500   0.2500   0.2500
        0        0        0        0        0  -1.0000  -1.0000
   0.3333        0        0   1.0000   0.3333        0        0

 Columns 8 and 9:

        0   1.0000
        0   0.5000
        0  -0.7500
   1.0000        0
        0   0.3333


Linha 4 do tableau corresponde a uma restrição redundante, portan
to vamos retirá-la.
Retiramos também a variável 7 das variáveis básicas.

Tableau final da fase 1:
 Columns 1 through 7:

        0        0        0        0        0   2.0000   2.0000
   1.0000   1.0000        0        0   0.5000   0.5000  -0.5000
   0.5000        0   1.0000        0  -0.7500   0.2500   0.2500
   0.3333        0        0   1.0000   0.3333        0        0

 Columns 8 and 9:

        0   1.0000
        0   0.5000
        0  -0.7500
        0   0.3333

----------      Fase 2      ----------
***********Iteração 0***************

Tableau:
  -1.8333        0        0        0  -0.0833
   1.0000   1.0000        0        0   0.5000
```

```
   0.5000        0   1.0000        0  -0.7500
   0.3333        0        0   1.0000   0.3333

 Entra na base: 4


Theta*
1.00000

Sai da base : 3

***********Iteração 1***************

Tableau:
  -1.7500        0        0   0.2500        0
   0.5000   1.0000        0  -1.5000        0
   1.2500        0   1.0000   2.2500        0
   1.0000        0        0   3.0000   1.0000

Solução ótima encontrada com custo 1.75000:
1 0.50000
2 1.25000
3 0.00000
4 1.00000
ind = 0
x =

   0.5000   1.2500        0   1.0000

d = NaN
```

Vamos analisar a última iteração de cada fase. Na última iteração da primeira fase vemos que nenhuma variável tem custo negativo, portanto estamos em uma solução ótima do problema auxiliar, como a entrada do canto superior esquerdo é igual a zero, concluimos que o problema é viável. Depois disso o programa indentifica que a variável artificial x_7 esta na base e correspone a 4 linha do tableau. Olhando para a quarta linha percebemos que todas as entradas das colunas de variáveis artificiais é igual a zero, portanto a quarta linha corresponde a uma restrição redundate e retiramos essa linha do tableau. Note que no *tableua* final da primeira fase tem uma linha a menos do que tinha no começo da iteração. Vamos agora para a última iteração da segunda fase, note que nenhuma entrada da linha zero do *tableau* é negativa, portanto o programa conclui que a solução atual é ótima e termina sua execução.