Pesquisa Operacional - Lista de Exercícios 02

Métodos Gráfico, Simplex e Simplex Duas Fases

Data de Entrega: 26-02-2021

- Professor: Felipe Reis
- Aluno: Lucas Mateus Fernandes
- Número: 0035411

Pontuação

Valor: 10 pontos

Perda de pontos por atraso na entrega

A tabela abaixo detalha a nota obtida possível, de acordo com o número de dias em atraso.

Dias em atraso	Nota máxima
0	100%
1	60%
2	0%

Perda de pontos por plágio.

Em caso de plágio, a lista será zerada.

Instruções

Após a conclusão da lista, faça do download arquivo em formato *.pdf* e submeta o arquivo na atividade do Google Classroom.
Boa prática!

Modelagem e Solução de PPLs pelo Método Gráfico

Esta seção tem como objetivo fazer com que o aluno exercite a modelagem e solução de problemas de Programação Linear pelo Método Gráfico.

Pontuação

Pontuação por questão:

- Questões 1 a 4: 0,75 pontos.
- Questões 5 e 6: 0,5 pontos.

Questões

[A-01] Exercício retirado de [Belfiore e Fávero, 2013]

<u>Problema de Produção</u>

Uma empresa de brinquedos está revendo seu planejamento de produção de carrinhos e triciclos. O lucro líquido por unidade de carrinho e triciclo produzido é de 12 unidades monetárias e 60 unidades monetárias, respectivamente. As matérias-primas e os insumos necessários para a fabricação de cada um dos produtos são terceirizados, cabendo à empresa os processos de usinagem, pintura e montagem. O processo de usinagem requer 15 minutos de mão de obra especializada por unidade de carrinho e 30 minutos por unidade de triciclo produzida. O processo de pintura requer 6 minutos de mão de obra especializada por unidade de carrinho e 45 minutos por unidade de triciclo produzida. Já o processo de montagem necessita de 6 minutos e 24 minutos para uma unidade de carrinho e de triciclo produzida, respectivamente. O tempo disponível por semana é de 36, 22 e 15 horas para os processos de usinagem, pintura e montagem, respectivamente. A empresa quer determinar quanto produzir de cada produto por semana, respeitando as limitações de recursos, de forma a maximizar o lucro líquido semanal.

Formule e resolva, utilizando o método gráfico, o problema de programação linear que maximiza o lucro.

```
{// Modelagem
        Função Objetiva é Maximizar o lucro: Max( Lucro )
        Lucro = 12 * qtdCarrinhos + 60 * qtdTriciclos

        Sujeito as restrições:

        //Tempo_Usinagem
        15 *  qtdCarrinhos + 30 * qtdTriciclos <= 2160

        //Tempo_Pintura
        6 *  qtdCarrinhos + 45 * qtdTriciclos <= 1320

        //Tempo_Monatagem
         6 *  qtdCarrinhos + 24 * qtdTriciclo <= 900

        //Não Negatividade
        qtdCarrinhos, qtdTriciclo >= 0
}
```

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa qtdCarrinhos
eixo das ordenadas, y, representa qtdTriciclos

Dada a restrição de Tempo_Usinagem tenho os pontos a1= (144, 0) e a2 = (0, 72) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo_Pintura tenho os pontos b1 = (220, 0) e b2 =(0, 264/9) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo_Montagem tenho os pontos c1 =(150, 0) e c2 = (0, 75/2) que definem a reta C sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes
a’ = (0,0)
b’ = (144,0)
c’ = (138,3)
d’ = (70,20)
e’ = Ponto(b2) = (0 , 264/9)

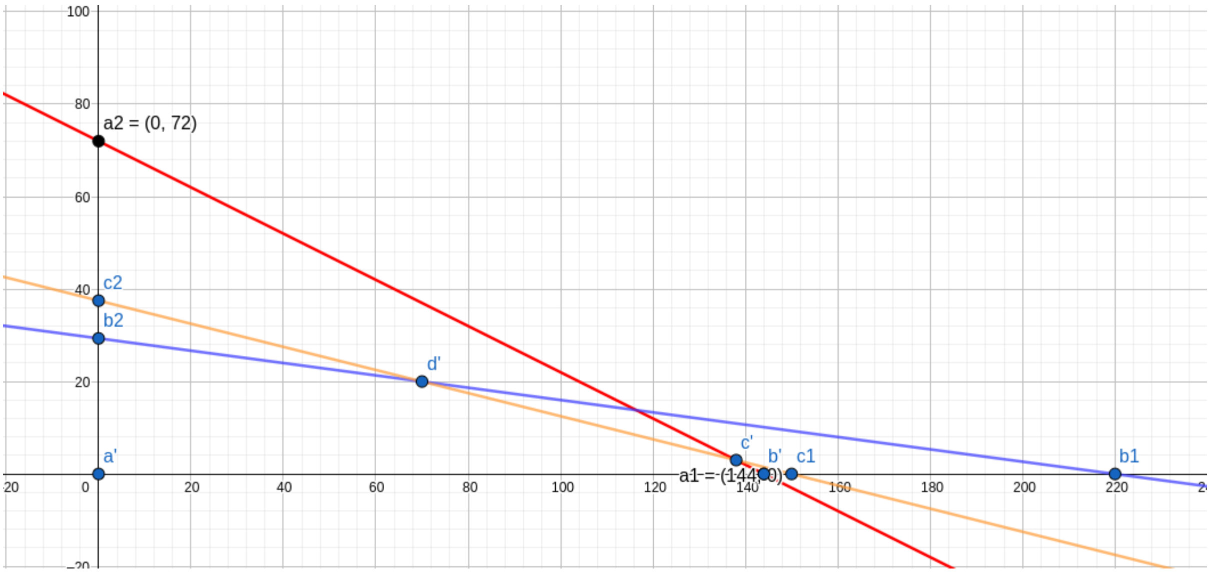
}

{//Resolução
Dada a função Lucro(qtdCarrinhos, qtdTriciclos) = 12 * qtdCarrinhos + 60 * qtdTriciclos

Temos que:

Lucro (a’) =	Lucro(0,0)	=	0
Lucro (b’) =	Lucro(144,0)	=	1728
Lucro (c’) =	Lucro(138,3)	=	1836
Lucro (d’) =	Lucro(70,20)	=	2040
Lucro (e’) =	Lucro(0 , 264/9)	=	1760

Sendo que o resultado final é qtdCarrinhos = 70, qtdTriciclos = 20
}



[A-02] Exercício retirado de [Moreira, 2013]

<u>Problema de Produção</u>

Uma fábrica produz dois produtos, A e B. Cada um deles é processado por duas máquinas, M1 e M2 . Devido à programação de outros produtos, a máquina M1 tem 24 horas de tempo disponível para os produtos A e B, enquanto que a máquina M2 tem 16 horas de tempo disponível. Para produzir uma unidade de produto A gastam-se 4 horas em cada uma das máquinas. Para produzir uma unidade de produto B gasta-se 6 horas na máquina M1 e 2 horas na máquina M2 . Cada unidade vendida do produto A gera um lucro de R$80,00, enquanto que cada unidade vendida de produto B gera um lucro de R$60,00. Existe previsão máxima de demanda para o produto B de 3 unidades, não havendo restrição quanto à demanda do produto A.

Formule e resolva, utilizando o método gráfico, o problema de programação linear que maximiza o lucro, a partir da produção dos produtos A e B

```
{// Modelagem
        Função Objetiva é Maximizar o lucro: Max( Lucro )
        Lucro = 80 * qtdA + 60 * qtdB

        Sujeito as restrições:

        //Tempo_M1
        4 * qtdA + 6 * qtdB <= 24

        //Tempo_M2
        4 * qtdA + 2 * qtdB <= 16

        //Limite De produção
        qtdB <= 3


        //Não Negatividade
         qtdA, qtdB >= 0
}
```

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa qtdA
eixo das ordenadas, y, representa qtdB

Dada a restrição de Tempo_M1 tenho os pontos a1= (6, 0) e a2 = (0, 4) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo_M2 tenho os pontos b1= (4, 0) e b2 = (0, 8) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo_M2 tenho os pontos c1= (0, 3) que definem a reta C sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes :
a’ = (0,0)
b’ = (4,0)
c’ = (3,2)
d’ = (3/2 ,3)
e’ = (0,3)
}

{//Resolução

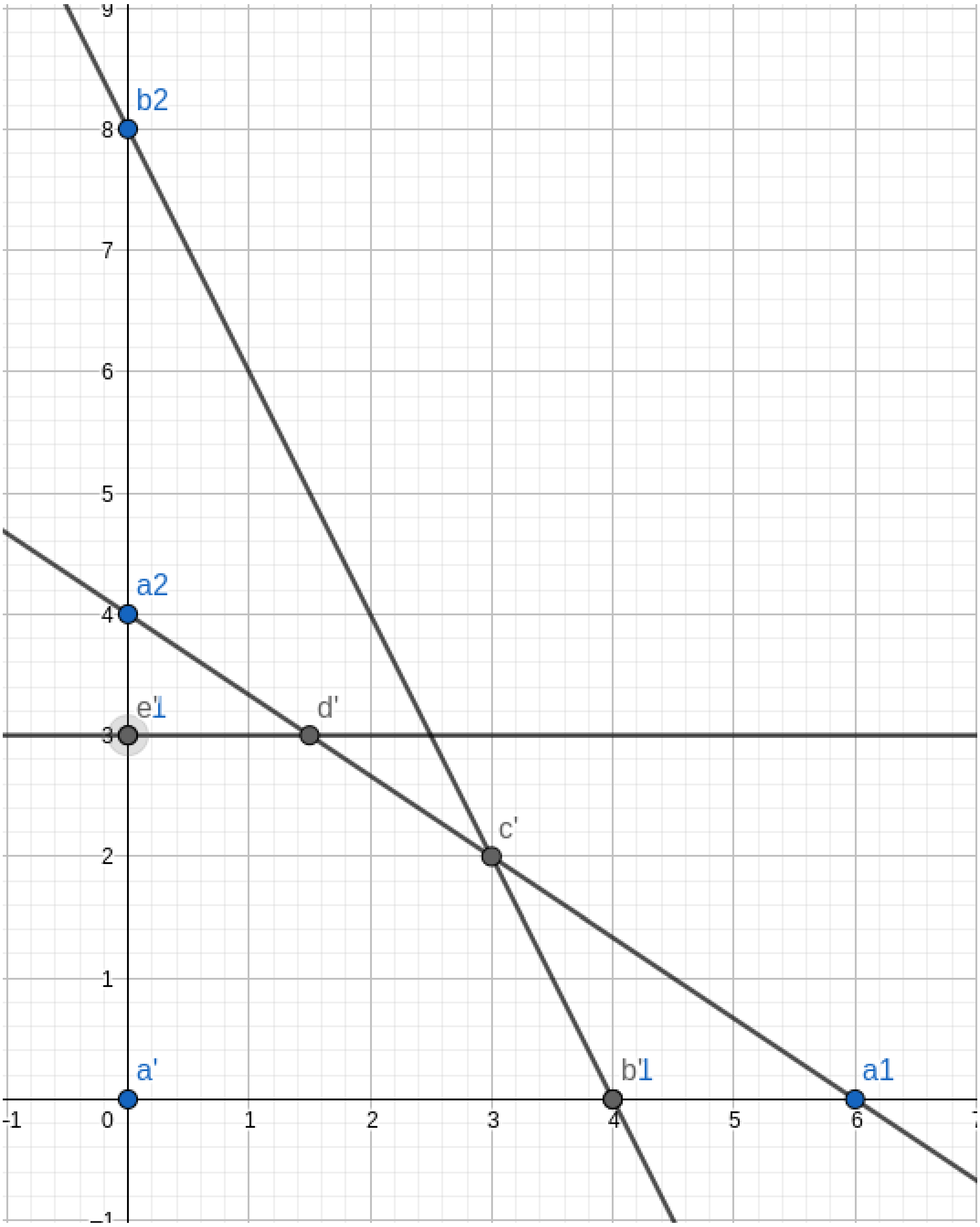
Dada a função Lucro(qtdA, qtdB) = 80 * qtdA + 60 * qtdB

Temos que:

Lucro (a') =	Lucro(0,0)	=	0
Lucro (b') =	Lucro(4,0)	=	320
Lucro (c') =	Lucro(3,2)	=	360
Lucro (d') =	Lucro(3/2, 3)	=	300
Lucro (e') =	Lucro(0 , 3)	=	180

Sendo que o resultado final é qtdA = 3, qtdB = 2

}



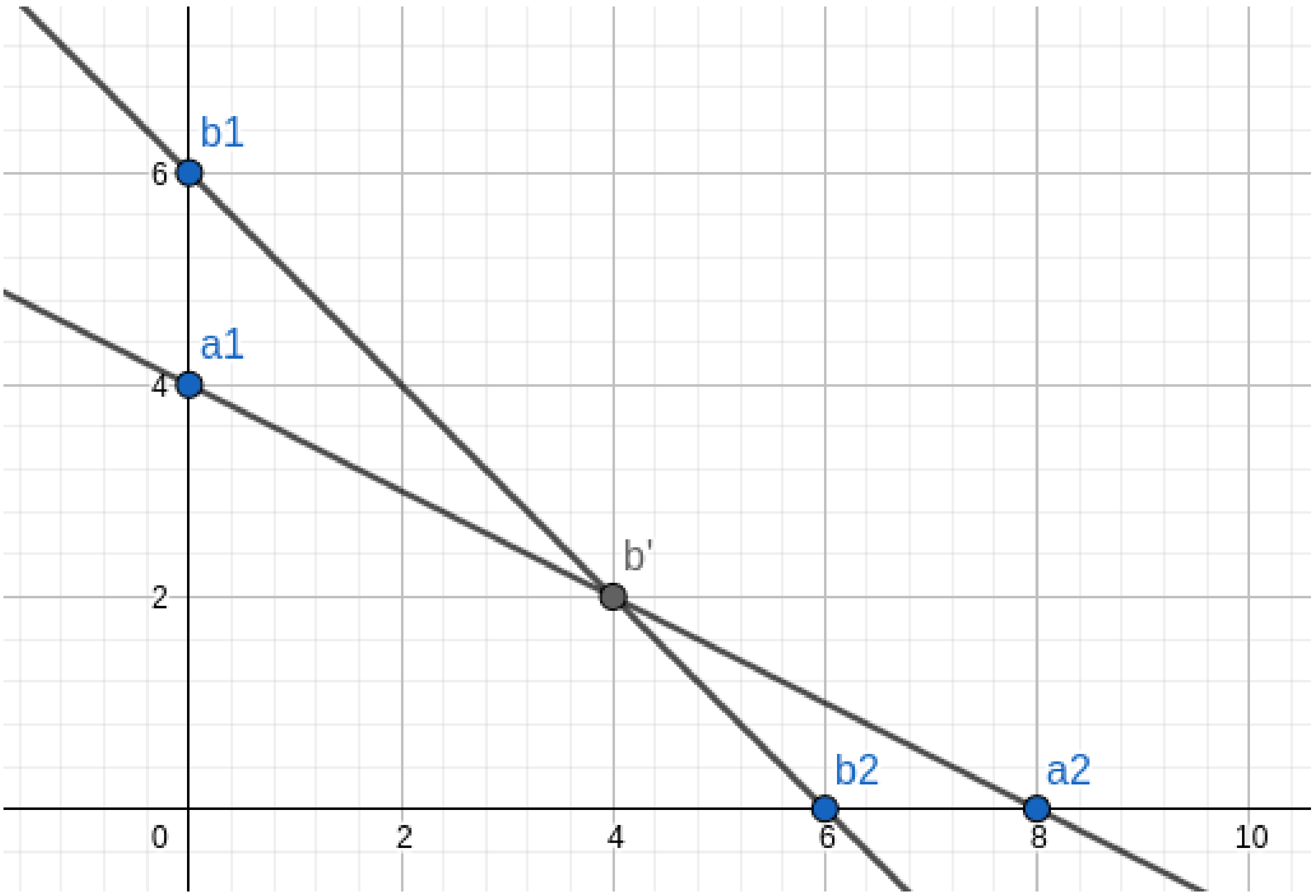
[A-03] Exercício retirado de [Silva, Silva e Gonçalves, 2010]

<u>Problema de Dieta</u>

Para uma boa alimentação, o corpo necessita de vitaminas e proteínas. A necessidade mínima de vitaminas é de 32 unidades/dia, e a de proteína de 36 unidades/dia. Uma pessoa dispões de carnes e ovos para se alimentar. Cada unidade de carne contém 4 unidades de vitaminas e 6 unidades de proteínas. Cada unidade de ovo contém 8 unidades de vitaminas e e 6 unidades de proteínas. Cada unidade de carne custa $3,00, e cada unidade de ovos custa $2,50 unidades monetárias.

Formule e resolva, utilizando o método gráfico, o problema de programação linear de forma a minimizar o custo da dieta, de modo a suprir as necessidades de vitaminas e proteínas.

```
{// Modelagem
        Função Objetiva é Minimizar o custo: Min( Custo )
        Custo = 3 * qtdCarne + 2.5 * qtdOvo

        Sujeito as restrições:

        //Qtd Vitamina
        4 * qtdCarne + 8 * qtdOvo >= 32

        //Qtd Proteina
        6 * qtdCarne + 6 * qtdOvo >= 36

        //Não Negatividade
         qtdCarne, qtdOvo >= 0
}

{// Adaptação das restrições para o modelo Grafico

        eixo das abscissas, x, representa  qtdCarne
        eixo das ordenadas, y, representa qtdOvo

        Dada a restrição de Qtd Vitamina tenho os pontos a1= (8, 0) e a2 = (0, 4) que definem a reta A sendo que (0,0)  não
        satisfaz a desigualdade da reta

        Dada a restrição de Tempo_Pintura  tenho os pontos  b1 = (6, 0) e b2 =(0, 6)  que definem a reta B sendo que (0,0)
        não satisfaz a desigualdade da reta


        Sendo os pontos Limitantes
        a' = b2 = (0,6)
        b' = (4,2)
        c' = a1 = (8,0)
}

{//Resolução
        Dada a função Custo( qtdCarne, qtdOvo) =  3 * qtdCarne + 2.5 * qtdOvo

        Temos que:
                Custo (a') =        Custo(0,6)                =        15
                Custo (b') =        Custo(4,2)                =        17
                Custo (c') =        Custo(8,0)                =        24

        Sendo que o resultado final é  qtdA = 0 qtdB = 6
}
```



b1
a1
b'
b2
a2
6
4
2
0
2
4
6
8
10

[A-04] Exercício adaptado de [Goldbarg and Luna, 2005]

<u>Outros Problemas</u>

Um jovem atleta sente-se atraído pela prática de dois esportes: natação e ciclismo. Sabe por experiência que:

- **A natação exige um gasto em mensalidade do clube e deslocamento até a piscina que pode ser expresso em um custo médio de 3 (três) reais por sessão de treinamento de duas horas.**
- **O ciclismo, mais simples, acaba custando cerca de 2 (dois) reais pelo mesmo tempo de prática.**
- **O orçamento do rapaz dispõe de 50 reais para seu treinamento.**
- **Seus afazeres de aluno de graduação na universidade, lhe dão liberdade de empregar, no máximo, 20 horas mensais para os esforços físicos.**

Considerando que o rapaz goste igualmente de ambos os esportes o problema consiste em planejar seu treinamento de forma a maximizar o número de sessões de treinamento.

{// Modelagem
Função Objetiva é Maximizar o numero de sessões o custo: Max(Sessoes)
Sessoes = qtdNatacao + qtdCiclismo

Sujeito as restrições:

//Custo
3 * qtdNatacao + 2 * qtdCiclismo <= 50

//Tempo
2 * qtdNatacao + 2 * qtdCiclismo <= 20

//Não Negatividade
qtdNatacao, qtdCiclismo >= 0
}

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa qtdNatacao
eixo das ordenadas, y, representa qtdCiclismo

Dada a restrição de Custo tenho os pontos aa= (50/3, 0) e ab = (0, 25) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo tenho os pontos ba = (10, 0) e bb =(0, 10) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

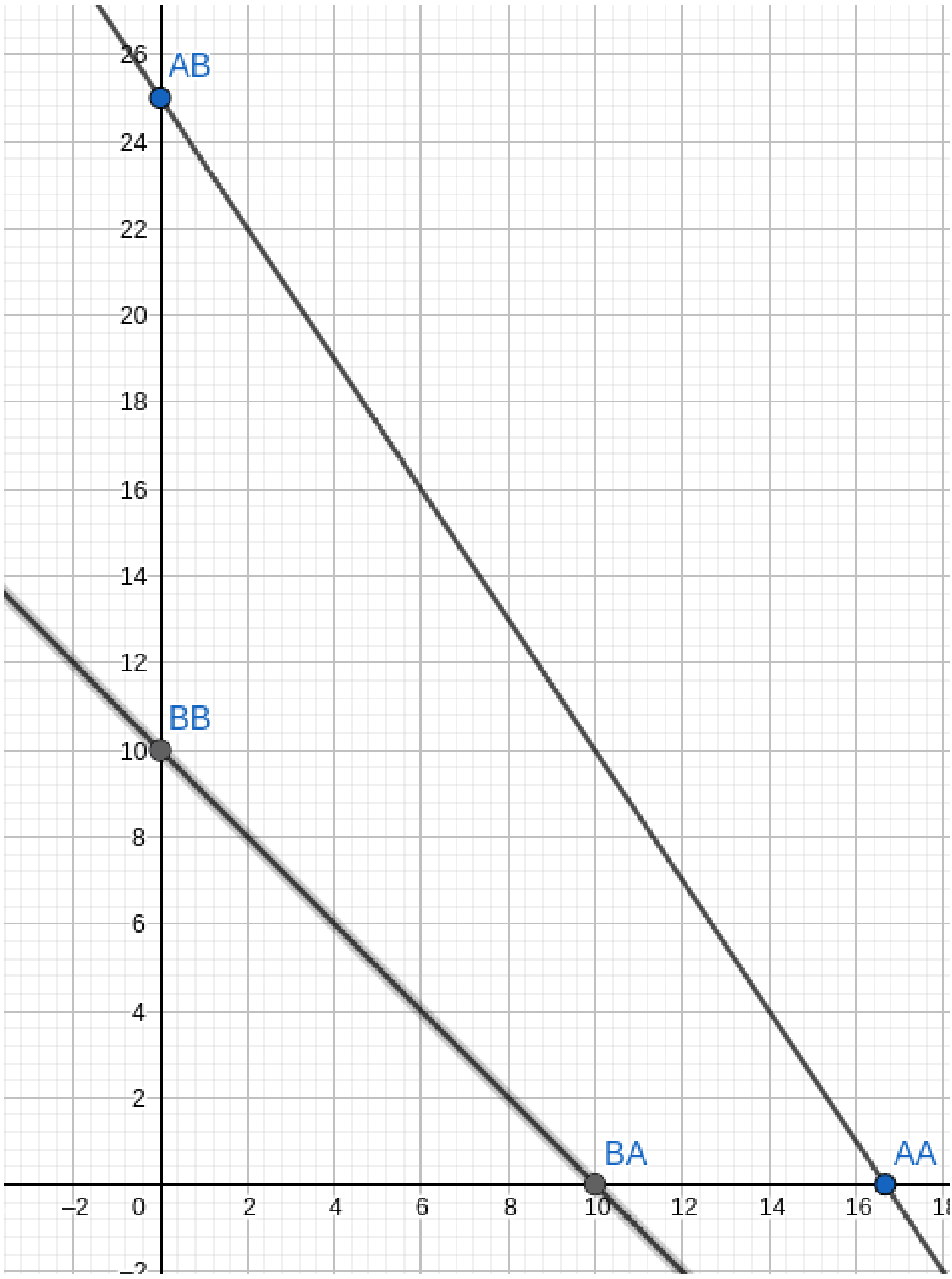
Sendo os pontos Limitantes
a' = bb = (0,6)
b' = (0,0)
c' = ba = (10,0)
}

{//Resolução
Dada a função Sessoes(qtdNatacao , qtdCiclismo) = qtdNatacao + qtdCiclismo

Temos que:

Sessoes (a') =	Sessoes(0,6)	=	6
Sessoes (b') =	Sessoes(4,2)	=	6
Sessoes (c') =	<u>Sessoes</u>(8,0)	=	8

Sendo que o resultado final é qtdNatacao = 8 e qtdCiclismo = 0
}


AB
BB
BA
AA
26
24
22
20
18
16
14
12
10
8
6
4
2
−2
0
2
4
6
8
10
12
14
16
18

[A-05] Exercícios retirados de [Belfiore e Fávero, 2013].

Resolva os problemas de programação linear a seguir utilizando o método gráfico.

a)

min $z = 4x_1 - 2x_2$

suj. a

$2x_1 + x_2 \leq 10$

$x_1 - x_2 \leq 8$

$x_1, x_2 \geq 0$

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa x1
eixo das ordenadas, y, representa x2

Dada a restrição de Custo tenho os pontos aa= (5,0) e ab = (0, 10) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

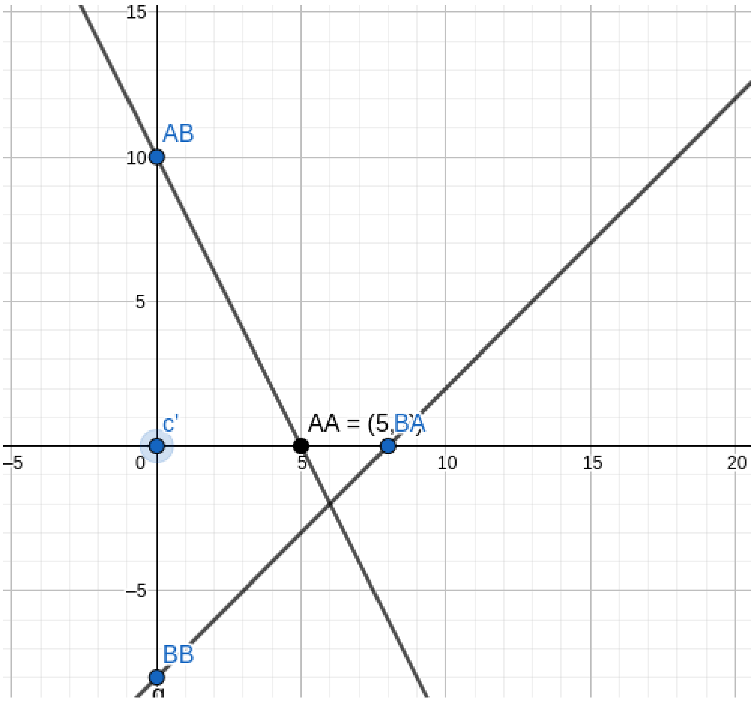
Dada a restrição de Tempo tenho os pontos ba = (8, 0) e bb =(0, -8) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes
a' = aa = (5,0)
b' = ab = (0,10)
c' = (0,0)
}
{//Resolução
Dada a função Min(x , y) = 4x - 2y

Temos que:

Min (a') = Min(5,0)	=	20
Min (b') = Min(0,10)	=	-20
Min (c') = Min(0,0)	=	0

Sendo que o resultado final é x1 = 0 e x2 = 10
}



b)

max z = $3x_1 + 2x_2$

suj. a

$x_1 + x_2 \leq 6$

$5x_1 + 2x_2 \leq 20$

$x_1, x_2 \geq 0$

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa x1
eixo das ordenadas, y, representa x2

Dada a restrição de Custo tenho os pontos aa= (6,0) e ab = (0, 6) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

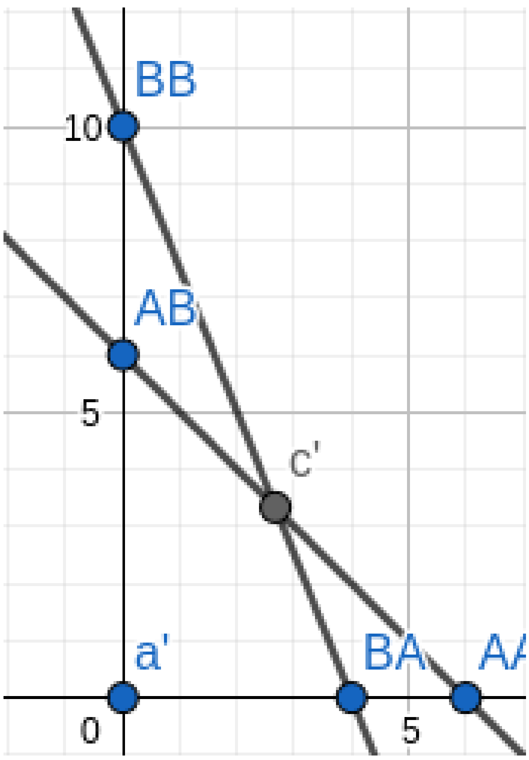
Dada a restrição de Tempo tenho os pontos ba = (4, 0) e bb = (0, 10) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes
a' = (0,0)
b' = (0,6)
c' = (8/3,10/3)
d' = (4,0)
}
{//Resolução
Dada a função Max(x , y) = 3x + 2y

Temos que:

Max (a') =	Max(0,0)	=	0
Max (b') =	Max(0,6)	=	12
Max (c') =	Max(8/3,10/3)	=	14.666666666666668
Max (d') =	Max(4,0)	=	12

Sendo que o resultado final é x1 = 8/3 e x2 = 10/3
}



[A-06] Exercícios retirados de [Taha, 2008]

Resolva os problemas de programação linear a seguir utilizando o método gráfico.

a)

max $z = 2x_1 + 3x_2$

suj. a

$2x_1 + x_2 \leq 4$

$x_1 + 2x_2 \leq 5$

$x_1, x_2 \geq 0$

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa x1
eixo das ordenadas, y, representa x2

Dada a restrição de Custo tenho os pontos aa= (2,0) e ab = (0, 4) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

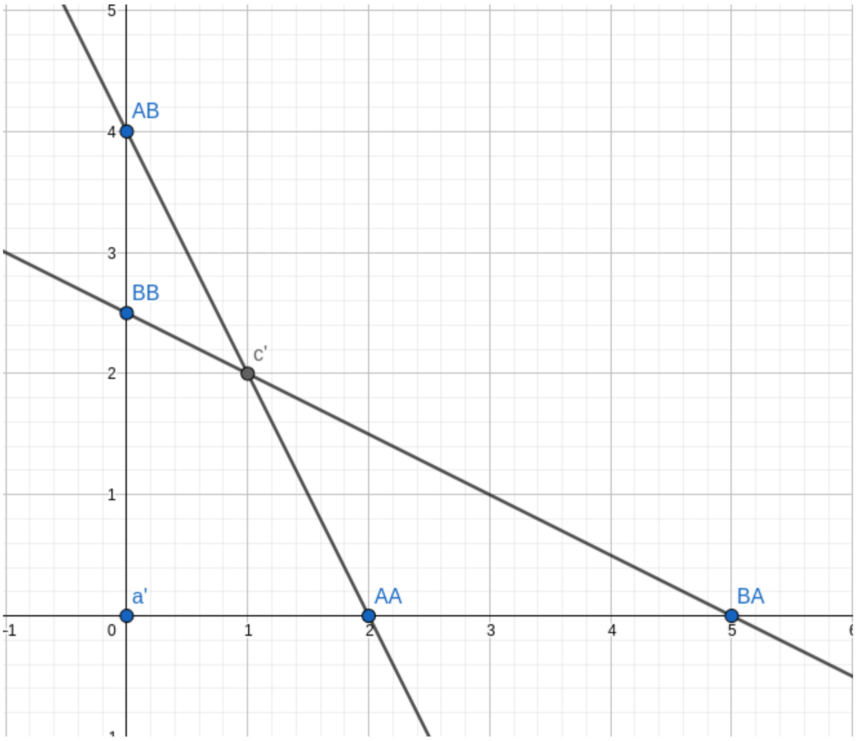
Dada a restrição de Tempo tenho os pontos ba = (5, 0) e bb = (0, 5/2) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes
a' = (0,0)
b' = aa = (2,0)
c' = (1,2)
d' = bb = (0,5/2)
}
{//Resolução
Dada a função Max(x , y) = 2x + 3y

Temos que:

Max (a') =	Max(0,0)	=	0
Max (b') =	Max(2,0)	=	4
Max (c') =	Max(1,2)	=	8
Max (d') =	Max(0,5/2)	=	7.5

Sendo que o resultado final é x1 = 1 e x2 = 2
}



b)

max z = $2x_1 + 3x_2$

suj. a

> **$x_1 + 3x_2 \leq 6$**
>
> **$3x_1 + 2x_2 \leq 6$**
>
> **$x_1, x_2 \geq 0$**

{// Adaptação das restrições para o modelo Grafico

eixo das abscissas, x, representa x1
eixo das ordenadas, y, representa x2

Dada a restrição de Custo tenho os pontos aa= (6,0) e ab = (0, 2) que definem a reta A sendo que (0,0) satisfaz a desigualdade da reta

Dada a restrição de Tempo tenho os pontos ba = (2, 0) e bb = (0, 3) que definem a reta B sendo que (0,0) satisfaz a desigualdade da reta

Sendo os pontos Limitantes
a' = (0,0)
b' = ab = (0,2)
c' = (857142/999999 , (714285+999999)/999999)
d' = ba = (2,0)

}

{//Resolução

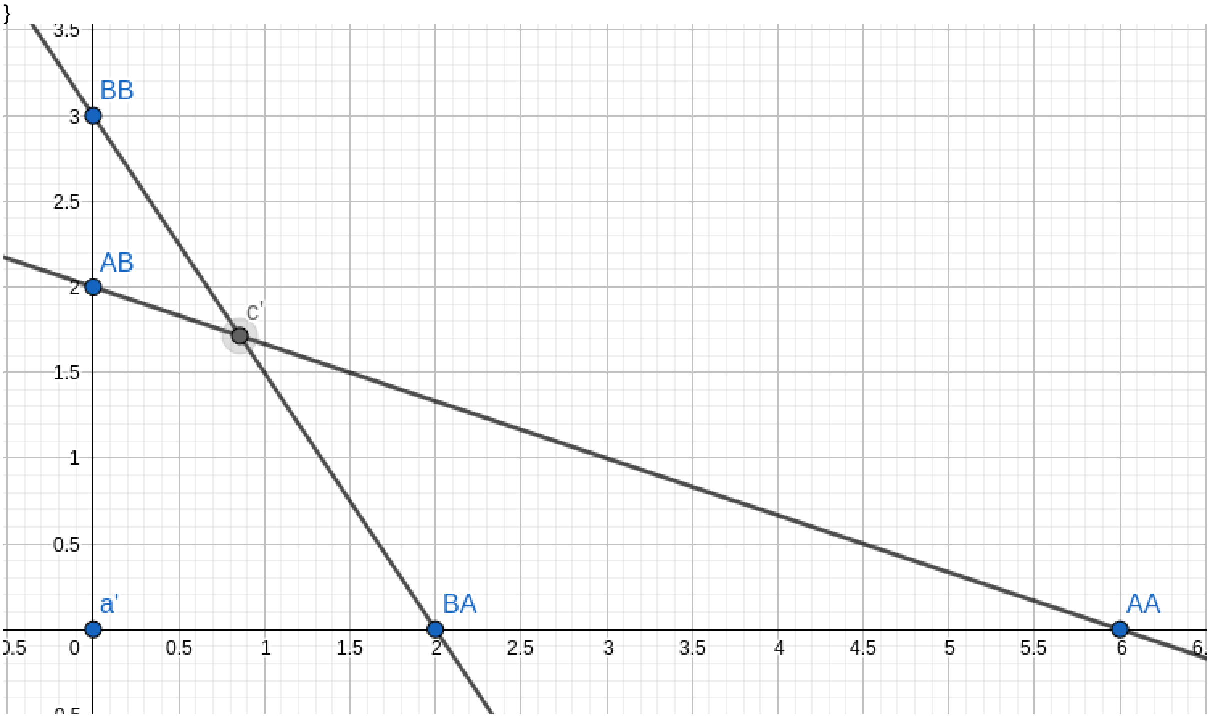
Dada a função Max(x , y) = 2x + 3y

Temos que:

Max (a') =	Max(0,0)	=	0
Max (b') =	Max(0,2)	=	6
Max (c') =	Max(857142/999999 , (714285+999999)/999999)	=	6.857142857142857
Max (d') =	Max(2,0)	=	4

Sendo que o resultado final é x1 = 857142/999999 e x2 = (714285+999999)/999999)

}



Solução de PPLs pelo Método Simplex e Simplex Duas Fases

Esta seção tem como objetivo fazer com que o aluno exercite a solução de problemas de Programação Linear pelos Métodos Simplex e Simplex Duas Fases.

Pontuação

Pontuação por questão:

- Questões 1 a 3: 1 ponto.
- Questões 4 e 5: 1,5 pontos.

[B-01] Exercício retirado de [Moreira, 2013]

<u>Problema de Produção</u>

Uma fábrica produz dois produtos, A e B. Cada um deles é processado por duas máquinas, M1 e M2 . Devido à programação de outros produtos, a máquina M1 tem 24 horas de tempo disponível para os produtos A e B, enquanto que a máquina M2 tem 16 horas de tempo disponível. Para produzir uma unidade de produto A gastam-se 4 horas em cada uma das máquinas. Para produzir uma unidade de produto B gasta-se 6 horas na máquina M1 e 2 horas na máquina M2 . Cada unidade vendida do produto A gera um lucro de R$80,00, enquanto que cada unidade vendida de produto B gera um lucro de R$60,00. Existe previsão máxima de demanda para o produto B de 3 unidades, não havendo restrição quanto à demanda do produto A.

Formule e resolva, utilizando o método gráfico, o problema de programação linear que maximiza o lucro, a partir da produção dos produtos A e B

```
{// Modelagem
        Função Objetiva é Maximizar o lucro da empresa: Max( Lucro )
        Lucro =  80*qtdA +  60*qtdB

        Sujeito as restrições:

        //Tempo M1
        4*qtdA +  6*qtdB <= 24

        //Tempo M2
        4*qtdA +  2*qtdB <= 16

        //Restrição superior
        qtdB <= 3

        //Não Negatividade
         qtdA, qtdB >= 0
}

{// Adaptação das restrições para o Simplex

        //Tempo M1
        4*qtdA +  6*qtdB +a1 = 24

        //Tempo M2
        4*qtdA +  2*qtdB +a2 = 16

        //Restrição superior
        qtdB +a3 = 3

        //Não Negatividade
         qtdA, qtdB ,a1, a2, a3>= 0
}
```

Transcreve para uma tabela e identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

	z	qtdA	qtdB	a1	a2	a3	Razão
z	**0**	-80	-60	0	0	0	
a1	**24**	4	6	1	0	0	**24/4 = 6**
a2	**16**	4	2	0	1	0	**16/4 = 4**
a3	**3**	0	1	0	0	1	**3/0 = Inf.**

{

QtdA = Coluna com menor valor negativo na função objetiva

a2 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô

qtdA	**4**	1	**0,5**	0	**0,25**	0

Fazer Gaus Jordam para Escalonar a tabela

	z	qtdA	qtdB	a1	a2	a3	Razão
z	**320**	**0**	**-20**	**0**	**20**	**0**	
a1	**8**	**0**	**4**	**1**	**-1**	**0**	**8/4 = 2**
qtdA	**4**	**1**	**0,5**	**0**	**0,25**	**0**	**4/0.5 = 8**
a3	**3**	**0**	**1**	**0**	**0**	**1**	**3/1 = 3**

Repetir o Loop

{

QtdB = Coluna com menor valor negativo na função objetiva

a1 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô

a1	**2**	**0**	**1**	**0,25**	**-0,25**	**0**

Fazer Gaus Jordam para Escalonar a tabela

	z	qtdA	qtdB	a1	a2	a3
z	**360**	**0**	**0**	**5**	**15**	**0**
qtdB	**2**	**0**	**1**	**0,25**	**-0,25**	**0**
qtdA	**3**	**1**	**0**	**-0,125**	**0,375**	**0**
a3	**1**	**0**	**0**	**-0,25**	**0,25**	**1**

Valor Otimo:
qtaA = 3
qtdB = 2

[B-02] <u>Outros Problemas</u>

A empresa Jardins Verdes possui dois tipos de atividades, poda de pequenas árvores e jardins (A) e paisagismo (B). A atividade A leva 2 horas, por m^2, para ser executada e rende aos cofres da empresa um lucro/hora de R$50,00. A atividade B, de maior valor agregado, leva 3 horas, por m^2, para ser executada e gera um lucro/hora de R$80,00. Segundo estimativas, o próximo mês irá requerer até 400 horas de prestação de serviços na atividade A e até 200 horas de prestação de serviços na atividade B. De acordo com o quadro de funcionários, a empresa possui capacidade de atendimento de até 500 horas mensais. Suponha que funcionários da empresa possam realizar ambas as atividades.

Formule e resolva, utilizando o método simplex, o problema de programação linear que maximiza o lucro da empresa.

```
{// Modelagem
        Função Objetiva é Maximizar o lucro da empresa: Max( Lucro )
        Lucro =  50*qtdA +  80*qtdB

        Sujeito as restrições:

        //Tempo
        2*qtdA +  3*qtdB <= 500
        2*qtdA  <= 400
        3*qtdB <= 200

        //Não Negatividade
         qtdA, qtdB >= 0
}

{// Adaptação das restrições para o Simplex

        2*qtdA +  3*qtdB  + a1 = 500
        2*qtdA  + a2 = 400
        3*qtdB +a3 = 200
}
```

Transcreve para uma tabela e identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

	z	qtdA	qtdB	a1	a2	a3	Razão
z	0	-50	-80	0	0	0	
a1	500	2	3	1	0	0	166,66666
a2	400	2	0	0	1	0	Inf
a3	200	0	3	0	0	1	66,666666

{

QtdB = Coluna com menor valor negativo na função objetiva

a3 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô

qtdB	66,66666666666	0	1	0	0	0,333333333333

Fazer Gaus Jordam para Escalonar a tabela

	z	qtdA	qtdB	a1	a2	a3	Razão
z	5333,3333	-50	0	0	0	26,666666	
a1	300	2	0	1	0	-1	150
a2	400	2	0	0	1	0	200
qtdB	66,666666	0	1	0	0	0,3333333	Inf

{

QtdA = Coluna com menor valor negativo na função objetiva

a1 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô

	z	qtdA	qtdB	a1	a2	a3
z	12833,3333	0	0	25	0	1,66666666
qtdA	150	1	0	0,5	0	-0,5
a2	100	0	0	-1	1	1
qtdB	66,6666666	0	1	0	0	0,33333333

Valor Otimo:
qtaA = 150
qtdB = 600/9

[B-03] Resolva usando o método Simplex ou Simplex Duas Fases.

max z = 9 x_1 + 8 x_2

suj. a

$4x_1 + 2x_2 <= 16$

$x_1 + x_2 <= 5$

$x_1 >= 1$

$x_2 <= 3$

$x_1 , x_2 >= 0$

{// Adaptação das restrições para o Simplex duas fases

4x1 + 2x2 +x3 = 16
x1 + x2 +x4 = 5
x1 -x5 + a1 = 1
x2 + x6 = 3
x1, x2, x3, x4, x5, x6, a1 >= 0

Variaveis auxiliares “x3 x4 x5 x6 a1”
Função objetiva auxiliar:
Min w = a1
Min w-a1 = 0

Eliminando as variáveis auxiliares da função objetiva auxiliar para poder inseri-la na base
w-a1 = 0
x1-x5+a1 = 1

w + x1 - x5 = 1

A função objetiva auxiliar se transformar em uma função de Max
Min w + x1 - x5 = 1 → Max -w -x1 +x5 = -1

Cria uma Tabela com a função objetiva auxiliar

	w	x1	x2	x3	x4	x5	x6	a1
w	-1	-1	0	0	0	1	0	0
x3	16	4	2	1	0	0	0	0
x4	5	1	1	0	1	0	0	0
a1	1	1	0	0	0	-1	0	1
x6	3	0	1	0	0	0	1	0

}

Resolvendo as tabelas intermediarias

	w	x1	x2	x3	x4	x5	x6	a1	Razão
w	-1	-1	0	0	0	1	0	0	
x3	16	4	2	1	0	0	0	0	16/4 = 4
x4	5	1	1	0	1	0	0	0	5/1 = 5
a1	1	1	0	0	0	-1	0	1	1/1 = 1
x6	3	0	1	0	0	0	1	0	3/0 = Inf

{

x1 = Coluna com menor valor negativo na função objetiva

a1 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

x1	1	1	0	0	0	-1	0	1

Fazer Gaus Jordam para Escalonar a tabela

	w	x1	x2	x3	x4	x5	x6	a1
w	0	0	0	0	0	0	0	1
x3	12	0	2	1	0	4	0	-4
x4	4	0	1	0	1	1	0	-1
x1	1	1	0	0	0	-1	0	1
x6	3	0	1	0	0	0	1	0

{// Adaptação das restrições para o Simplex duas fases parte 2 (ignorando as variaveis auxiliares)

max z = 9 x1 + 8 x2

sujeito a:

2x2 + x3 + 4x5 = 12
x2 + x4 + x5 = 4
x1 -x5 = 1
x2 + x6 = 3

OBS: Como x1 faz parte da base de acordo com a ultima tabela ele não pode fazer parte da função objetiva então é necessário substituir x1 por algo equivalente na função objetiva

Max z = 9x1 + 8x2 - > Max z - 9x1 – 8x2 = 0

z - 9x1 – 8x2 = 0
(x1 – x5 = 1) *9 [multiplica tudo por nove para poder anular o x1 da função objetiva , semelhante ao escalonamento]

z-9x5-8x2=9

}

Transcreve a nova tabela

	z	x1	x2	x3	x4	x5	x6	Razão
z	9	0	-8	0	0	-9	0	
x3	12	0	2	1	0	4	0	12/4=3
x4	4	0	1	0	1	1	0	4/1=4
x1	1	1	0	0	0	-1	0	1/-1=-1
x6	3	0	1	0	0	0	1	3/0=Inf

Identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

x5 = Coluna com menor valor negativo na função objetiva

x3 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

x3	12	0	2	1	0	4	0

Fazer Gaus Jordam para Escalonar a tabela

	z	x1	x2	x3	x4	x5	x6	Razão
z	36	0	-3,5	2,25	0	0	0	
x5	3	0	0,5	0,25	0	1	0	3/0,5=6
x4	1	0	0,5	-0,25	1	0	0	1/0,5=2
x1	4	1	0,5	0,25	0	0	0	4/0,5=8
x6	3	0	1	0	0	0	1	3/1=3

Transcreve para uma tabela e dentifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

x2 = Coluna com menor valor negativo na função objetiva

x4 = Linha com menor razão positiva

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

x2	2	0	1	-0,5	2	0	0

Fazer Gaus Jordam para Escalonar a tabela

	z	x1	x2	x3	x4	x5	x6
z	43	0	0	0,5	7	0	0
x5	2	0	0	0,5	-1	1	0
x2	2	0	1	-0,5	2	0	0
x1	3	1	0	0,5	-1	0	0
x6	1	0	0	0,5	-2	0	1

Valor Otimo:
x1 = 3
x2 = 2

[B-04] Exercício retirado de [Silva, Silva e Gonçalves, 2010]

<u>Problema de Dieta</u>

Para uma boa alimentação, o corpo necessita de vitaminas e proteínas. A necessidade mínima de vitaminas é de 32 unidades/dia, e a de proteína de 36 unidades/dia. Uma pessoa dispões de carnes e ovos para se alimentar. Cada unidade de carne contém 4 unidades de vitaminas e 6 unidades de proteínas. Cada unidade de ovo contém 8 unidades de vitaminas e e 6 unidades de proteínas. Cada unidade de carne custa $3,00, e cada unidade de ovos custa $2,50 unidades monetárias.

Formule e resolva, utilizando o método gráfico, o problema de programação linear de forma a minimizar o custo da dieta, de modo a suprir as necessidades de vitaminas e proteínas.

{// Modelagem
Função Objetiva é minimizar o custo: Min(Custo)
Custo = 3*qtdCarne + 2.5*qtdOvo

Sujeito as restrições:

//Quantidade
4*qtdCarne + 8*qtdOvo >= 32
6*qtdCarne + 6*qtdOvo>= 36

//Não Negatividade
qtdCarne, qtdOvo >= 0
}

{// Adaptação das restrições para o Simplex duas fases
Func Objetiva
-3*qtdCarne – 2.5*qtdOvo

Sujeito a :

4*qtdCarne + 8*qtdOvo - x1 + a1= 32
6*qtdCarne + 6*qtdOvo -x2 + a2 = 36

Variaveis auxiliares “x1, x2, a1, a2”
Função objetiva auxiliar:
Min w = a1+a2
Min w-a1-a2 = 0

Eliminando as variáveis auxiliares da função objetiva auxiliar para poder inseri-la na base
w-a1-a2 = 0
4*qtdCarne + 8*qtdOvo - x1 + a1= 32
6*qtdCarne + 6*qtdOvo -x2 + a2 = 36

w + 10*qtdCarne + 14*qtdOvo – x1 -x2 = 68

A função objetiva auxiliar se transformar em uma função de Max
Min w + 10*qtdCarne + 14*qtdOvo – x1 -x2 = 68 → Max -w - 10*qtdCarne - 14*qtdOvo + x1 +x2 = -68

Cria uma Tabela com a função objetiva auxiliar

	w	qtdCarne	qtdOvo	x1	x2	a1	a2
w	-68	-10	-14	1	2	0	0
a1	32	4	8	-1	0	1	0
a2	36	6	6	0	-1	0	1

}

Resolvendo as tabelas intermediarias

	w	qtdCarne	qtdOvo	x1	x2	a1	a2	Razão
w	-68	-10	-14	1	2	0	0	
a1	32	4	8	-1	0	1	0	32/8= 4
a2	36	6	6	0	-1	0	1	36/6= 6

Transcreve para uma tabela e identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

qtdOvo = Coluna com menor valor negativo na função objetiva .

a1 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdOvo	4	0,5	1	-0,125	0	0,125	0

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdCarne	qtdOvo	x1	x2	a1	a2	Razão
w	-12	-3	0	-0,75	2	1,75	0	
qtdOvo	4	0,5	1	-0,13	0	0,13	0	4/0,5=8
a2	12	3	0	0,75	-1	-0,75	1	12/3=4

Transcreve para uma tabela e identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

qtdCarne = Coluna com menor valor negativo na função objetiva .

a2 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdCarne	4	1	0	0,25	-0,33	-0,25	0,33

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdCarne	qtdOvo	x1	x2	a1	a2
w	0	0	0	0	1	1	1
qtdOvo	2	0	1	-0,25	0,17	0,25	-0,17
qtdCarne	4	1	0	0,25	-0,33	-0,25	0,33

{// Adaptação das restrições para o Simplex duas fases parte 2 (ignorando as variaveis auxiliares)

max -z = -3*qtdCarne - 2.5*qtdOvo

sujeito a:

qtdOvo +(-0,25)x1 + (0.17)x2 = 2
qtdCarne +(0,25)x1 + (-0.33)x2 = 4

OBS: Como qtdOvo e qtdCarne fazem parte da base de acordo com a ultima tabela eles não pode fazer parte da função objetiva então é necessário substituir qtdOvo e qtdCarne por algo equivalente na função objetiva max -z = -3*qtdCarne – 2.5*qtdOvo → max -z + 3*qtdCarne + 2.5*qtdOvo = 0

max -z + 3*qtdCarne + 2.5*qtdOvo = 0
[(qtdOvo +(-0,25)*x1 + (0.17)*x2 = 2] *(-2.5)
[qtdCarne +(0,25) *x1 + (-0.33)*x2 = 4] *(-3)

max -z +(-0,125)X1 + (0,575)x2 = -17

}
Transcreve a nova tabela

	z	qtdCarne	qtdOvo	x1	x2	Razão
z	-17	0	0	-0,13	0,58	
qtdOvo	2	0	1	-0,25	0,17	2/-0,25= -8
qtdCarne	4	1	0	0,25	-0,33	4/(0,25)=16

Identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

x1 = Coluna com menor valor negativo na função objetiva .

qtdCarne = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

	z	qtdCarne	qtdOvo	x1	x2
z	-14,92	0,52	0	0	0,4084
qtdOvo	6	1	1	0	-0,16
x1	16	4	0	1	-1,32

Valor Otimo:
qtdOvo = 6
qtdCarne = 0

[B-05] <u>Problema de Produção (Software)</u>

Uma empresa de desenvolvimento de software possui três tipos de desenvolvedores: front-end, back-end e full-stack. O desenvolvedor front-end pode atuar em tarefas do tipo A, o desenvolvedor back-end pode atuar em tarefas do tipo B, enquanto o desenvolvedor full-stack pode atuar tanto em tarefas do tipo A quanto em tarefas do tipo B. Um determinado projeto requer 500 horas de desenvolvimento do tipo A e 1000 horas de desenvolvimento do tipo B. O desenvolvedor front-end recebe um salário de R$20,00 por hora trabalhada, os desenvolvedores back-end e full-stack recebem R$25,00 por hora trabalhada. O quadro de funcionários da empresa possibilita, durante o prazo de desenvolvimento do projeto, a execução de até 400 horas de trabalho por desenvolvedores front-end, 800 horas de trabalho de desenvolvedores back-end e até 600 horas de trabalho para desenvolvedores full-stack.

Formule e resolva, usando o método Simplex, o problema de programação linear que minimize o custo com salário de desenvolvedores de software na empresa.

```
{// Modelagem
        Função Objetiva é minimizar o custo: Min( Custo )
        Custo =  20*qtdFront + 25*qtdBack + 25*(qtdFullA + qtdFullB)

        Sujeito as restrições:

        //Quantidade
        qtdFrontA+ qtdFullA = 500
        qtdBack + qtdFullB = 1000
        qtdFront <= 400
        qtdBack <= 800
        qtdFullA + qtdFullB <= 600

        //Não Negatividade
         qtdFront, qtdBack, qtdFullA, qtdFullB  >= 0


}
{// Adaptação das restrições para o Simplex duas fases
        Func Objetiva
        20*qtdFront + 25*qtdBack + 25*(qtdFullA + qtdFullB)

        Sujeito a :

        qtdFront + qtdFullA +a1          = 500
        qtdBack + qtdFullB +a2           = 1000
        qtdFront +x1                     = 400
        qtdBack +x2                      = 800
        qtdFullA + qtdFullB  +x3         = 600

        Variaveis auxiliares "x1, x2,,x3, a1, a2, a3, a4, a5"
        Função objetiva auxiliar:
                Min w = a1+a2
                Min w-a1-a2= 0
                ____________
                Eliminando as variáveis auxiliares  da função objetiva auxiliar para poder inseri-la na base
                Min w-a1-a2                                      = 0
                qtdFront           + qtdFullA           +a1      = 500
                         qtdBack             +qtdFullB      +a2  = 1000
                ___________________________
                w + qtdFront + qtdBack +qtdFullA +qtdFullB       = 1500

        A função  objetiva auxiliar se transformar em uma função de Max
                Min   w + qtdFront + qtdBack +qtdFullA +qtdFullB    = 1500
                Max -w - qtdFront - qtdBack  -qtdFullA    -qtdFullB   = -1500

        Cria uma Tabela com a função objetiva auxiliar
```

	w	qtdFront	qtdBack	qtdFull A	qtdFull B	x1	x2	x3	a1	a2	RAZÂO
w	-1500	-1	-1	-1	-1	0	0	0	0	0	
a1	500	1	0	1	0	0	0	0	1	0	500
a2	1000	0	1	0	1	0	0	0	0	1	Inf.
x1	400	1	0	0	0	1	0	0	0	0	Inf.
x2	800	0	1	0	0	0	1	0	0	0	Inf.
x3	600	0	0	1	1	0	0	1	0	0	600

}

Transcreve para uma tabela e identifica a coluna com menor valor negativo e a linha com menor razão positiva (Z / coluna_MenorValor)

{

qtdFullA = Coluna com menor valor negativo na função objetiva .

a1 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdFullA	500	1	0	1	0	0	0	0	1	0

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3	a1	a2	RAZÂO
w	-1000	0	-1	0	-1	0	0	0	1	0	
qtdFullA	500	1	0	1	0	0	0	0	1	0	Inf.
a2	1000	0	1	0	1	0	0	0	0	1	1000
x1	400	1	0	0	0	1	0	0	0	0	Inf.
x2	800	0	1	0	0	0	1	0	0	0	Inf.
x3	100	-1	0	0	1	0	0	1	-1	0	100

{

qtdFullB = Coluna com menor valor negativo na função objetiva .

x3 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdFullB	100	-1	0	0	1	0	0	1	-1	0

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3	a1	a2	RAZÂO
w	-900	-1	-1	0	0	0	0	1	0	0	
qtdFullA	500	1	0	1	0	0	0	0	1	0	Inf.
a2	900	1	1	0	0	0	0	-1	1	1	900
x1	400	1	0	0	0	1	0	0	0	0	Inf.
x2	800	0	1	0	0	0	1	0	0	0	800
qtdFullB	100	-1	0	0	1	0	0	1	-1	0	Inf.

{

qtdBack = Coluna com menor valor negativo na função objetiva .

x2 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdBack	800	0	1	0	0	0	1	0	0	0

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3	a1	a2	RAZÂO
w	-100	-1	0	0	0	0	1	1	0	0	
qtdFullA	500	1	0	1	0	0	0	0	1	0	500
a2	100	1	0	0	0	0	-1	-1	1	1	100
x1	400	1	0	0	0	1	0	0	0	0	400
qtdBack	800	0	1	0	0	0	1	0	0	0	Inf.
qtdFullB	100	-1	0	0	1	0	0	1	-1	0	-100

{

qtdFront = Coluna com menor valor negativo na função objetiva .

a2 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

qtdFront	100	1	0	0	0	0	-1	-1	1	1

Fazer Gaus Jordam para Escalonar a tabela

	w	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3	a1	a2
w	0	0	0	0	0	0	0	0	1	1
qtdFullA	400	0	0	1	0	0	1	1	0	-1
qtdFront	100	1	0	0	0	0	-1	-1	1	1
x1	300	0	0	0	0	1	1	1	-1	-1
qtdBack	800	0	1	0	0	0	1	0	0	0
qtdFullB	200	0	0	0	1	0	-1	0	0	1

{

// Adaptação das restrições para o Simplex duas fases parte 2 (ignorando as variáveis auxiliares)

Min z → Max -z
Max -z = (-20)*qtdFront + (-25)*qtdBack + (-25)*(qtdFullA + qtdFullB)
sujeito a:
qtdFullA +x2 +3 = 400
qtdFront -x2 -x3 = 100
qtdBack +x2 = 800
qtdFullB – x2 = 200

OBS: Como qtdFront, qtdBack, qtdFullA e qtdFullB fazem parte da base de acordo com a ultima tabela eles não pode fazer parte da função objetiva então é necessário substituir qtdBack e qtdFull por algo equivalente na função objetiva

fazendo a transformação da função objetiva:
Max -z +(25)*qtdBack + (20)*qtdFront + (25)*(qtdFullA + qtdFullB)   =   0
-25qtdBack   -25x2   = -20000
-20qtdFront   +20x2 +20x3 = -2000
-25qtdFullA   -25x2 - 25x3 = -10000
-25qtdFullB +25 x2   = -5000

Max -z   -5x2   -5x3   = -37000
}

	z	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3	RAZÂO
z	37000	0	0	0	0	0	-5	-5	
qtdFullA	400	0	0	1	0	0	1	1	400
qtdFront	100	1	0	0	0	0	-1	-1	-100
x1	300	0	0	0	0	1	1	1	300
qtdBack	800	0	1	0	0	0	1	0	800
qtdFullB	200	0	0	0	1	0	-1	0	-200

{

x2 = Coluna com menor valor negativo na função objetiva .

x1 = Linha com menor razão positiva.

}

Dividir a linha pivô pelo (valor da interseção da linha pivô com a coluna pivô) substituindo a variável da linha pivô pela variável da coluna pivô.

x2	300	0	0	0	0	1	1	1	300

Fazer Gaus Jordam para Escalonar a tabela

	z	qtdFront	qtdBack	qtdFullA	qtdFullB	x1	x2	x3
z	38500	0	0	0	0	5	0	0
qtdFullA	100	0	0	1	0	-1	0	0
qtdFront	400	1	0	0	0	1	0	0
x2	300	0	0	0	0	1	1	1
qtdBack	500	0	1	0	0	-1	0	-1
qtdFullB	500	0	0	0	1	1	0	1

Resultado Final:

qtdFront = 400
qtdBack = 500
qtdFullA = 100
qtdFullB = 500

Referências

1. GOLDBARG, Marco Cesar; LUNA, Henrique Pacca L. **Otimização combinatória e programação linear: modelos e algoritmos**, 2a edição. Rio de Janeiro: Elsevier, 2005. ISBN: 9788535215205.

2. BELFIORE, P., FÁVERO, L. P. **Pesquisa Operacional para cursos de engenharia**. Editora Campus, 2013. ISBN: 9788535248937.

3. TAHA, H. A. **Pesquisa Operacional**. 8a edição. Editora Prentice-Hall Brasil, ISBN 9788576051503, 2008.

4. MOREIRA, D. A. **Pesquisa operacional: curso introdutório. 2a edição revista e atualizada.** Editora Cengage Learning. 2a edição. 2013.

5. SILVA, E. M. d., SILVA, E. M. d., GONÇALVES, V. **Pesquisa Operacional para os Cursos de Administração e Engenharia.** Editora Atlas. 4a edição. 2010