Problem Sheet 6.

Problem 1. Solve the following LPP using An Interior-Point Algorithm
Maximize and minimize

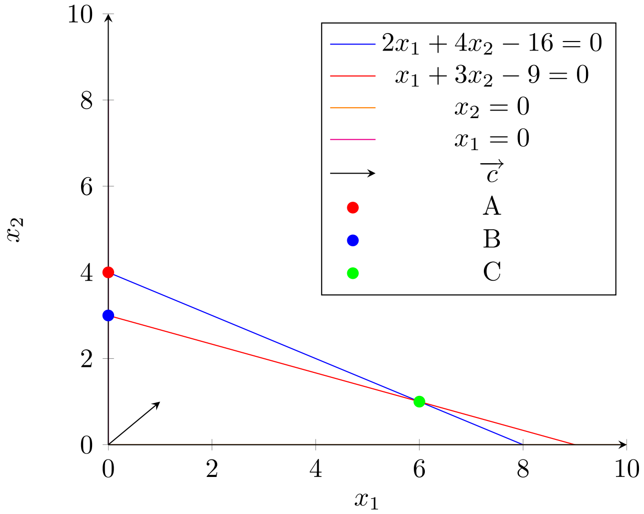
$$F(x_1, x_2) = x_1 + x_2$$

subject to

$$\begin{cases} 2x_1 + 4x_2 \leq 16, & (1) \\ x_1 + 3x_2 \geq 9, & (2) \\ x_1 \geq 0, & (3) \\ x_2 \geq 0 & (4). \end{cases}$$

Solution

Using conditions (1)-(4) we get the following feasible Solution Space:



We know (**see Lab 2, Problem 2**) that the minimal value of F is equal to

$$F(0, 3) = 0 + 3 = 3$$

and the maximal value of F equals

$$F(6, 1) = 6 + 1 = 7.$$

To begin with we need to write our problem in the following equality form
Maximize and minimize

$$F(x_1, x_2) = x_1 + x_2 + 0x_3 + 0x_4$$

subject to

$$\begin{cases} 2x_1 + 4x_2 + x_3 = 16, & (1') \\ x_1 + 3x_2 - x_4 = 9, & (2') \\ x_1 \geq 0, & (3') \\ x_2 \geq 0 & (4'), \\ x_3 \geq 0 & (5'), \\ x_4 \geq 0 & (6'). \end{cases}$$

Iteration 1. We need to take the initial trial solution that lies in **the interior** of the feasible region, i.e., inside the boundary of the feasible region. For example,

$$(x_1, x_2, x_3, x_4) = \left(\frac{1}{2}, \frac{7}{2}, 1, 2\right)$$

satisfies the conditions $(1') - (6')$.

1. Using this the current trial solution, we can fill the following matrix

$$\mathbf{D} = \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 7/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}.$$

We put that (see $(1') - (6')$)

$$\mathbf{A} = \begin{bmatrix} 2 & 4 & 1 & 0 \\ 1 & 3 & 0 & -1 \end{bmatrix}, \qquad \mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

2. Calculate

$$\widetilde{\mathbf{A}} = \mathbf{AD} = \begin{bmatrix} 2 & 4 & 1 & 0 \\ 1 & 3 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 7/2 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 14 & 1 & 0 \\ 1/2 & 21/2 & 0 & -2 \end{bmatrix}$$

and

$$\widetilde{\mathbf{c}} = \mathbf{Dc} = \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 0 & 7/2 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 7/2 \\ 0 \\ 0 \end{bmatrix}$$

3. Calculate

$$\mathbf{P} = \mathbf{I} - \widetilde{\mathbf{A}}^T (\widetilde{\mathbf{A}}\widetilde{\mathbf{A}}^T)^{-1} \widetilde{\mathbf{A}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 1/2 \\ 14 & 21/2 \\ 1 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} 198 & 295/2 \\ 295/2 & 229/2 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 14 & 1 & 0 \\ 1/2 & 21/2 & 0 & -2 \end{bmatrix} =$$

$$\begin{bmatrix} 0.981962 & -0.06696 & -0.04455 & -0.1060 \\ -0.06696 & 0.00901886 & -0.05931 & 0.03061 \\ -0.04455 & -0.05931 & 0.874829 & -0.3225 \\ -0.1060 & 0.03061 & -0.3225 & 0.13419 \end{bmatrix}$$

and

$$\mathbf{c}_p = \mathbf{P}\widetilde{\mathbf{c}} = \begin{bmatrix} 0.256627 \\ -0.00191 \\ -0.229844 \\ 0.0541131 \end{bmatrix}$$

4. Identify the negative component of c_p having the largest absolute value, and set ν equal to this absolute value. Obviously, $\nu = 0.229844$. Then

$$\widetilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \frac{\alpha}{\nu}\mathbf{c}_p = \begin{bmatrix} 1.55826 \\ 0.995838 \\ 0.5 \\ 1.11772 \end{bmatrix}$$

5. Calculate $x = \mathbf{D}\widetilde{\mathbf{x}}$ as the trial solution for the next iteration (step 1). Consequently,

$$(x_1, x_2, x_3, x_4) = \begin{bmatrix} 0.779132 \\ 3.48543 \\ 0.5 \\ 2.23543 \end{bmatrix}.$$

Iteration 2. We put that the initial trial solution is

$$(x_1, x_2, x_3, x_4) = \begin{bmatrix} 0.779132 \\ 3.48543 \\ 0.5 \\ 2.23543 \end{bmatrix}.$$

In iteration 1 we have $x = [0.77913199; \quad 3.48543401; \quad 0.5; \quad 2.23543401]$
In iteration 2 we have $x = [1.88058104; \quad 2.99720948; \quad 0.25; \quad 1.87220948]$
In iteration 3 we have $x = [3.80836045; \quad 2.04258143; \quad 0.21295338; \quad 0.93610474]$
In iteration 4 we have $x = [4.78702489; \quad 1.56034249; \quad 0.18458025; \quad 0.46805237]$
In the last iteration 5 we have
$x = [5.99999375e+00; \quad 1.00000267e+00; \quad 1.78547810e-06; \quad 1.78547810e-06]$

We used

```
import numpy
import numpy as np
from numpy.linalg import norm

x = np.array([2, 2, 4,3], float)
alpha = 0.5
A = np.array([[2,-2,8,0],[-6,-1,0,-1]],float)
c = np.array([-2, 3, 0,0], float)




i = 1

while True:
    v = x
    D = np.diag(x)

    AA = np.dot(A,D)
    cc = np.dot(D,c)

    I = np.eye(4)

    F  = np.dot(AA, np.transpose(AA))
    FI = np.linalg.inv(F)
    H = np.dot(np.transpose(AA), FI)

    P = np.subtract(I, np.dot(H, AA) )

    cp = np.dot(P,cc)

    nu = np.absolute(np.min(cp))
    y =  np.add(np.ones(4,float),(alpha/nu)*cp)

    yy = np.dot(D,y)

    x = yy

    if i==1 or i == 2  or i == 3 or i == 4:
        print("In iteration  ", i , " we have x = ", x , "\n" )
        i = i +  1



    if norm(np.subtract(yy,v),ord = 2)<  0.00001:
        break


print("In the last iteration  ", i, "  we have x=  \n", x)
```

If we consider minimization problem we will have

In iteration 1 we have $x = [0.77913199; \quad 3.48543401; \quad 0.5; \quad 2.23543401]$
In iteration 2 we have $x = [1.88058104; \quad 2.99720948; \quad 0.25; \quad 1.87220948]$
In iteration 3 we have $x = [3.80836045; \quad 2.04258143; \quad 0.21295338; \quad 0.93610474]$
In iteration 4 we have $x = [4.78702489; \quad 1.56034249; \quad 0.1845802; \quad 50.46805237]$
In the last iteration 5 we have
$x = [5.99999375e+00; \quad 1.00000267e+00; \quad 1.78547810e-06; \quad 1.78547810e-06]$.

Problem 2.
Solve following LPP using An Interior-Point Algorithm.
Maximize

$$F(x_1, x_2, x_3) = 9x_1 + 10x_2 + 16x_3$$

subject to

$$\begin{cases} 18x_1 + 15x_2 + 12x_3 \leq 360, & (1) \\ 6x_1 + 4x_2 + 8x_3 \leq 192, & (2) \\ 5x_1 + 3x_2 + 3x_3 \leq 180, & (3) \\ x_1 \geq 0, & (4) \\ x_2 \geq 0, & (5) \\ x_3 \geq 0. & (6) \end{cases}$$

Solution

We know (**see Lab 3, Problem 1**) that that optimal solution

$$x_1 = 0, x_2 = 8, x_3 = 20$$

and the maximal value of F is equal to 400.

To begin with we need to rewrite this problem in standard form. We have
Maximize $9x_1 + 10x_2 + 16x_3 + 0 \cdot s_1 + 0 \cdot s_2 + 0 \cdot s_3$

$$\begin{cases} 18x_1 + 15x_2 + 12x_3 + x_4 = 360, \\ 6x_1 + 4x_2 + 8x_3 + x_5 = 192, \\ 5x_1 + 3x_2 + 3x_3 + x_6 = 180, \\ x_1 \geq 0, \\ x_2 \geq 0, \\ x_3 \geq 0, \\ x_4 \geq 0, \\ x_5 \geq 0, \\ x_6 \geq 0. \end{cases}$$

Iteration 1. We need to take the initial trial solution that lies in **the interior** of the feasible region, i.e., inside the boundary of the feasible region. For example,

$$(x_1, x_2, x_3, x_4, x_5, x_6) = (1, 1, 1, 315, 174, 169)$$

satisfies all conditions above.

1. Using this the current trial solution, we can fill the following matrix

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 315 & 0 & 0 \\ 0 & 0 & 0 & 0 & 181 & 0 \\ 0 & 0 & 0 & 0 & 0 & 169 \end{bmatrix}.$$

We put that (see $(1') - (6')$)

$$\mathbf{A} = \begin{bmatrix} 18 & 15 & 12 & 1 & 0 & 0 \\ 6 & 4 & 8 & 0 & 1 & 0 \\ 5 & 3 & 3 & 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{c} = \begin{bmatrix} 9 \\ 10 \\ 16 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

2. Calculate

$$\widetilde{\mathbf{A}} = \mathbf{A}\mathbf{D} = \begin{bmatrix} 18 & 15 & 12 & 1 & 0 & 0 \\ 6 & 4 & 8 & 0 & 1 & 0 \\ 5 & 3 & 3 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 315 & 0 & 0 \\ 0 & 0 & 0 & 0 & 181 & 0 \\ 0 & 0 & 0 & 0 & 0 & 169 \end{bmatrix}$$

and

$$\widetilde{\mathbf{c}} = \mathbf{D}\mathbf{c} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 315 & 0 & 0 \\ 0 & 0 & 0 & 0 & 181 & 0 \\ 0 & 0 & 0 & 0 & 0 & 169 \end{bmatrix} \begin{bmatrix} 9 \\ 10 \\ 16 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

3. Calculate

$$\mathbf{P} = \mathbf{I} - \widetilde{\mathbf{A}}^T (\widetilde{\mathbf{A}}\widetilde{\mathbf{A}}^T)^{-1} \widetilde{\mathbf{A}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} - \dots$$

and

$$\mathbf{c}_p = \mathbf{P}\widetilde{\mathbf{c}};$$

4. Identify the negative component of c_p having the largest absolute value, and set ν equal to this absolute value. Then

$$\widetilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \frac{\alpha}{\nu}\mathbf{c}_p.$$

5. Calculate $x = \mathbf{D}\widetilde{\mathbf{x}}$ as the trial solution for the next iteration (step 1). Consequently,

In iteration 1 we have

$$x = [3.79800303; \quad 4.12638468; \quad 6.0200146; \quad 157.5; \quad 104.54632627; \quad 130.57078699]$$

In iteration 2 we have

$$x = [4.19318272; \quad 5.14657344; \quad 10.71450912; \quad 78.75; \quad 60.53853696; \quad 111.45083872]$$

In iteration 3 we have

$$x = [3.23188407; \quad 4.90476314; \quad 15.34004682; \quad 44.17407787; \quad 30.26926848; \quad 103.10614979]$$

In iteration 4 we have

$$x = [2.37343412; \quad 4.4584146; \quad 18.09888783; \quad 33.21531292; \quad 15.13463424100.46092212]$$

In the last iteration 5 we have

$$x = [2.96713352e{-}06; \quad 7.99999106e{+}00; \quad 2.00000010e{+}01; \quad 6.67605042e{-}058.90140056e{-}06; \quad 9.60000086e{+}01]$$

```
import numpy
import numpy as np
from numpy.linalg import norm

x = np.array([1, 1, 1, 315, 174, 169], float)
alpha = 0.5
A = np.array([[18,15,12,1,0,0],[6,4,8,0, 1, 0], [5,3,3,0, 0, 1]],float)
c = np.array([9, 10, 16, 0, 0,0], float)
```

```python
i = 1

while True:
    v = x
    D = np.diag(x)

    AA = np.dot(A,D)
    cc = np.dot(D,c)
    I = np.eye(len(c))
    F  = np.dot(AA, np.transpose(AA))
    FI = np.linalg.inv(F)
    H = np.dot(np.transpose(AA), FI)
    P = np.subtract(I, np.dot(H, AA) )
    cp = np.dot(P,cc)
    nu = np.absolute(np.min(cp))
    y =  np.add(np.ones(len(c),float),(alpha/nu)*cp)
    yy = np.dot(D,y)

    x = yy

    if i==1 or i == 2  or i == 3 or i == 4:
        print("In iteration  ", i , " we have x = ", x , "\n" )
        i = i +  1

    if norm(np.subtract(yy,v),ord = 2)< 0.0001:
        break


print("In the last iteration  ", i, "  we have x=  \n", x)
```

Problem 3. Solve the following LPP using the M-method.
Minimize

$$F(x_1, x_2, x_3, x_4) = -2x_1 + 3x_2 - 6x_3 - x_4$$

subject to

$$\begin{cases} 2x_1 + x_2 - 2x_3 + x_4 = 24, & (1) \\ x_1 + 2x_2 + 4x_3 \leq 22, & (2) \\ x_1 - x_2 + 2x_3 \geq 10, & (3) \\ x_1 \geq 0, & (4) \\ x_2 \geq 0, & (5) \\ x_3 \geq 0, & (6) \\ x_4 \geq 0. & (7) \end{cases}$$

Solution

To begin with we need to rewrite this problem in standard form. We have
Maximize

$$F(x_1, x_2, x_3, x_4, s_1, s_2) = 2x_1 - 3x_2 + 6x_3 + x_4 + 0 \cdot s_1 + 0 \cdot s_2$$

subject to

$$\begin{cases} 2x_1 + x_2 - 2x_3 + x_4 = 24, & (1) \\ x_1 + 2x_2 + 4x_3 + s_1 = 22, & (2) \\ x_1 - x_2 + 2x_3 - s_2 = 10, & (3) \\ x_1 \geq 0, & (4) \\ x_2 \geq 0, & (5) \\ x_3 \geq 0, & (6) \\ x_4 \geq 0, & (7) \\ s_1 \geq 0, & (8) \\ s_2 \geq 0. & (9) \end{cases}$$

The first and second equations have a basic variable, x_4 and s_1, but the third equation does not. Thus, we add the artificial variables R_1 in the third equation and penalize them in the objective function with $-MR_1$ (because we are maximizing).

Therefore, we get the following LPP

Maximize

$$F(x_1, x_2, x_3, x_4, s_1, s_2, R_1) = 2x_1 - 3x_2 + 6x_3 + x_4 + 0 \cdot s_1 + 0 \cdot s_2 - MR_1.$$

subject to

$$\begin{cases} 2x_1 + x_2 - 2x_3 + x_4 = 24, & (1) \\ x_1 + 2x_2 + 4x_3 + s_1 = 22, & (2) \\ x_1 - x_2 + 2x_3 - s_2 + R_1 = 10, & (3) \\ x_1 \geq 0, & (4) \\ x_2 \geq 0, & (5) \\ x_3 \geq 0, & (6) \\ x_4 \geq 0, & (7) \\ s_1 \geq 0, & (8) \\ s_2 \geq 0, & (9) \\ R_1 \geq 0. & (9) \end{cases}$$

The starting basic solution is $(x_1, x_2, x_3, x_4, s_1, s_2, R_1) =$ (0,0,0,24, 22,0,10). It appears reasonable to set $M = 100$.

Iteration	Basic	x_1	$\boxed{x_2}$	x_3	x_4	s_1	s_2	R_1	*Solution*	c_i
0	z	-2	3	-6	-1	0	0	100		
	x_4	2	1	-2	1	0	0	0	24	1
	s_1	1	2	4	0	1	0	0	22	0
	R_1	1	-1	2	0	0	-1	1	10	-100

Before proceeding with the simplex method computations, the z-row must be made consistent with the rest of the tableau.

$$z_{new} = z_{old} + (1 \cdot x_4Row + 0 \cdot s_1Row - 100 \cdot R_1Row)$$

where

$$z_{old} = (-2, 3, -6, -1, 0, 0, 100)$$

Iteration	Basic	x_1	x_2	x_3	x_4	s_1	s_2	R_1	*Solution*
0	z	-100	104	**-208**	0	0	100	0	-976
x_3 enters	x_4	2	1	-2	1	0	0	0	24
	s_1	1	2	4	0	1	0	0	22
R_1 leaves	R_1	1	-1	2	0	0	-1	1	10
1	z	4	0	0	0	0	**-4**		64
s_2 enters	x_4	3	0	0	1	0	-1		34
	s_1	-1	4	0	0	1	2		2
s_1 leaves	x_3	1/2	-1/2	1	0	0	-1/2		5
2	z	2	8	0	0	2	0		68
	x_4	5/2	2	0	1	1/2	0		35
	s_2	-1/2	2	0	0	1/2	1		1
	x_3	1/4	1/2	1	0	1/4	0		11/2

Problem 4. Solve the following LPP using Two-Phase Method.
Maximize

$$F(x_1, x_2) = x_1 + x_2$$

subject to

$$\begin{cases} 2x_1 + 4x_2 \le 16, & (1) \\ -4x_1 + 2x_2 \le 8, & (2) \\ x_1 + 3x_2 \ge 9, & (3) \\ x_1 \ge 0, & (4) \\ x_2 \ge 0 & (5). \end{cases}$$

Solution

To begin with we need to rewrite this problem in standard form. We have
Maximize

$$F(x_1, x_2, s_1, s_2, s_3) = x_1 + x_2$$

subject to

$$\begin{cases} 2x_1 + 4x_2 + s_1 = 16, & (1) \\ -4x_1 + 2x_2 + s_2 = 8, & (2) \\ x_1 + 3x_2 - s_3 = 9, & (3) \\ x_1 \ge 0, & (4) \\ x_2 \ge 0, & (5) \\ s_1 \ge 0, & (6) \\ s_2 \ge 0, & (7) \\ s_3 \ge 0. & (8) \end{cases}$$

Phase I
Minimize

$$r = R_1$$

or Maximize

$$r = -R_1$$

subject to

$$\begin{cases} 2x_1 + 4x_2 + s_1 = 16, & (1) \\ -4x_1 + 2x_2 + s_2 = 8, & (2) \\ x_1 + 3x_2 - s_3 + R_1 = 9, & (3) \\ x_1 \ge 0, & (4) \\ x_2 \ge 0, & (5) \\ s_1 \ge 0, & (6) \\ s_2 \ge 0, & (7) \\ s_3 \ge 0, & (8) \\ R_1 \ge 0. & (9) \end{cases}$$

Using the previous equalities, we get

The starting basic solution is $(x_1, x_2, s_1, s_2, s_3, R_1) = (0, 0, 16, 8, 0, 9)$.

Iteration	Basic	x_1	x_2	s_1	s_2	s_3	R_1	*Solution*	c_i
0	r	0	0	0	0	0	1		
	x_3	2	4	1	0	0	0	16	0
	x_4	-4	2	0	1	0	0	8	0
	R_1	1	3	0	0	-1	1	9	-1

Before proceeding with the simplex method computations, the z-row must be made consistent with the rest of the tableau.

$$z_{new} = z_{old} + (0 \cdot s_1 Row + 0 \cdot -s_2 Row - 1 \cdot R_1 Row)$$

where

$$z_{old} = (0, 0, 0, 0, 0, 0, -1)$$

We have

Iteration	Basic	x_1	x_2	s_1	s_2	s_3	R_1	*Solution*
0	r	-1	**-3**	0	0	1	0	-9
x_2 enters	x_3	2	4	1	0	0	0	16
	x_4	-4	2	0	1	0	0	8
R_1 leaves	R_1	1	3	0	0	-1	1	9
1	r	0	0	0	0	0	1	0
	x_3	2/3	0	1	0	4/3	4/3	4
	x_4	-14/3	0	0	1	2/3	2/3	2
	x_2	1/3	1	0	0	-1/3	1/3	3

Phase II

Now we have the following problem Maximize

$$F(x_1, x_2) = x_1 + x_2$$

subject to

$$\begin{cases} 2/3x_1 + s_1 + 4/3s_3 = 4, & (1) \\ -14/3x_1 + s_2 + 2/3s_3 = 2, & (2) \\ 1/3x_1 + x_2 - 1/3s_3 = 3, & (3) \\ x_1 \geq 0, & (4) \\ x_2 \geq 0, & (5) \\ s_1 \geq 0, & (6) \\ s_2 \geq 0, & (7) \\ s_3 \geq 0. & (8) \end{cases}$$

We get

Iteration	Basic	x_1	x_2	s_1	s_2	s_3	*Solution*	c_i
0	z	-1	-1	0	0	0	0	
	s_1	2/3	0	1	0	4/3	4	0
	s_2	-14/3	0	0	1	2/3	2	0
	x_2	1/3	1	0	0	-1/3	3	1

Using that

$$z_{new} = z_{old} + (0 \cdot x_3 Row + 0 \cdot x_4 Row + 1 \cdot x_2 Row)$$

where

$$z_{old} = (0, 0, 0, 0, 0, 0, -1)$$

we have

Iteration	Basic	x_1	x_2	s_1	s_2	s_3	*Solution*
0	z	**-2/3**	0	0	0	-1/5	3
x_1 enters	x_3	2/3	0	1	0	4/3	4
	x_4	-14/3	0	0	1	2/3	2
x_3 leaves	x_2	1/3	1	0	0	-1/3	3
1	z	0	0	1	0	17/15	7
	x_1	1	0	3/2	0	2	6
	x_4	0	0	7	1	10	30
	x_2	0	1	-1/2	0	-13/15	1

$$(x_1, x_2) = (6, 1).$$

and

$$F(x_1, x_2) = 7.$$