Problem 2 : Two-phased simplex method

Solve the following program using a two-phased simplex method by hand :

$$Objective : max(3x + 4y)$$

\begin{equation*} s.t.

$$x + 2y \leq 7$$
$$3x - y \geq 0$$
$$x - y \leq 2$$
$$x, y \geq 0$$

\end{equation*}

(a) Standard form

$$\max \quad 3x + 4y$$

$$\text{s.t.} \quad x + 2y + s_1 = 7 \quad ✓$$
$$3x - y - e_2 = 0 \quad a_2$$
$$x - y + s_3 = 2 \quad ✓$$
$$x, y, s_1, e_2, s_3 \geq 0$$

(b) Auxiliary Problem

$$\min \quad a_2 \quad \longleftrightarrow \quad \max \quad W = -a_2$$
$$W + a_2 = 0$$

$$\text{s.t.} \quad x + 2y + s_1 = 7$$
$$3x - y - e_2 + a_2 = 0$$
$$x - y + s_3 = 2$$
$$x, y, s_1, e_2, s_3, a_2 \geq 0$$

(c) Phase I.

W	x	y	s_1	e_2	s_3	a_2	RHS	BV
1	0	0	0	0	0	1	0	W
0	1	2	1	0	0	0	7	s_1
0	3	-1	0	-1	0	1	0	a_2
0	1	-1	0	0	1	0	2	s_3

W	x	y	s_1	e_2	s_3	a_2	RHS	BV	
1	-3	1	0	1	0	0	0	W	$R_0 - R_2$
0	1	2	1	0	0	0	7	s_1	
0	3	-1	0	-1	0	1	0	a_2	
0	1	-1	0	0	1	0	2	s_3	

W	$\boxed{x}$	y	s_1	e_2	s_3	a_2	RHS	BV	ratio
1	$\boxed{-3}$	1	0	1	0	0	0	W	
0	1	2	1	0	0	0	7	s_1	$7/1 = 7$
0	3	-1	0	-1	0	1	0	a_2	$0/3 = 0 \leftarrow$ min
0	1	-1	0	0	1	0	2	s_3	$2/1 = 2$

W	x	y	s_1	e_2	s_3	a_2	RHS	BV	
1	0	0	0	0	0	1	0	W	$R_0 + R_2$
0	0	$7/3$	1	$1/3$	0	$-1/3$	7	s_1	$R_1 - R_2/3$
0	1	$-1/3$	0	$-1/3$	0	$1/3$	0	x	$R_2/3$
0	0	$-2/3$	0	$1/3$	1	$-1/3$	2	s_3	$R_3 - R_2/3$

Phase II.

W	x	y	s_1	e_2	s_3	a_2	RHS	BV
1	0	0	0	0	0	1	0	W
0	0	$7/3$	1	$1/3$	0	$-1/3$	7	s_1
0	1	$-1/3$	0	$-1/3$	0	$1/3$	0	x
0	0	$-2/3$	0	$1/3$	1	$-1/3$	2	s_3

$$\max\ z = 3x + 4y \iff z - 3x - 4y = 0$$

z	x	y	s_1	e_2	s_3	RHS	BV
1	-3	-4	0	0	0	0	z
0	0	$7/3$	1	$1/3$	0	7	s_1
0	1	$-1/3$	0	$-1/3$	0	0	x
0	0	$-2/3$	0	$1/3$	1	2	s_3

z	x	y	s_1	e_2	s_3	RHS	BV		ratio
1	0	-5	0	-1	0	0	z	$R_0 + 3R_2$	
0	0	$7/3$	1	$1/3$	0	7	s_1		$7/(7/3) = 7$ ← min
0	1	$-1/3$	0	$-1/3$	0	0	x		$0/(-1/3) = -0$ (∞)
0	0	$-2/3$	0	$1/3$	1	2	s_3		$1/(-2/3)$ (∞)

z	x	y	s_1	e_2	s_3	RHS	BV	
1	0	-5	0	-1	0	0	z	
0	0	1	$3/7$	$1/7$	0	3	y	$\frac{3}{7}R_1$
0	1	$-1/3$	0	$-1/3$	0	0	x	
0	0	$-2/3$	0	$1/3$	1	2	s_3	

z	x	y	s_1	e_2	s_3	RHS	BV	
1	0	0	$15/7$	$-2/7$	0	15	z	$R_0 + 5R_1$
0	0	1	$3/7$	$1/7$	0	3	y	
0	1	0	$1/7$	$-6/21$	0	1	x	$R_2 + \frac{R_1}{3}$
0	0	0	$2/7$	$9/21$	1	4	s_3	$R_3 + \frac{2}{3}R_1$

z	x	y	s_1	e_2	s_3	RHS	BV	ratio
1	0	0	$15/7$	$-2/7$	0	15	z	
0	0	1	$3/7$	$1/7$	0	3	y	$3/(1/7) = 21$
0	1	0	$1/7$	$-6/21$	0	1	x	$1/(-6/21)$ (∞)
0	0	0	$2/7$	$9/21$	1	4	s_3	$4/(9/21) = 21 \times \frac{4}{9}$ ← min

z	x	y	s_1	e_2	s_3	RHS	BV	
1	0	0	15/7	−2/7	0	15	z	
0	0	1	3/7	1/7	0	3	y	
0	1	0	1/7	−6/21	0	1	x	
0	0	0	2/3	1	$\frac{21}{9}$	$\frac{84}{9}$	e_2	$\frac{21}{9}R_3$

z	x	y	s_1	e_2	s_3	RHS	BV	
1	0	0	7/3	0	2/3	53/3	z	$R_0 + \frac{2}{7}R_3$
0	0	1	1/3	0	−1/3	5/3	y	$R_1 - \frac{1}{7}R_3$
0	1	0	1/3	0	2/3	11/3	x	$R_2 + \frac{6}{21}R_3$
0	0	0	2/3	1	$\frac{7}{3}$	$\frac{28}{3}$	e_2	

No -ve ∴ Optimal.

$\therefore z = \frac{53}{3}, \quad y = \frac{5}{3}, \quad x = \frac{11}{3}, \quad e_2 = \frac{28}{3}$ #

<<< Only problem 2 and 6 will be graded. >>>

Problem 1: Simplex method

Solve the following program using the Simplex method by hand :

$$Objective : max(3x + 4y)$$

$$s.t. \begin{aligned} x + 2y &\leq 7 \\ 3x - y &\leq 5 \\ x - y &\leq 2 \\ x, y &\geq 0 \end{aligned}$$

In [1]:
```python
pass
```

Problem 2 : Two-phased simplex method

Solve the following program using a two-phased simplex method by hand :

$$Objective : max(3x + 4y)$$

$$s.t. \begin{aligned} x + 2y &\leq 7 \\ 3x - y &\geq 0 \\ x - y &\leq 2 \\ x, y &\geq 0 \end{aligned}$$

In [2]:
```python
pass
```

Problem 3 : Unrestricted variable

Solve the following program:

$$Objective : min(3x + 4y)$$

$$s.t. \begin{aligned} x + 2y &\leq 7 \\ 7x - y &\geq 2 \\ x - 2y &\leq 2 \end{aligned}$$

Find the solution of x, y in a standard form, and explain the behavior of the optimized unrestricted variables.

In [3]:
```python
pass
```

Problem 4: Proof

(Winston p.139 problem 6) For an LP in standard form with constraint $A\mathbf{x} = \mathbf{b}$ and $\mathbf{x} \geq 0$ show that $\mathbf{d}$ is a direction of unboundedness if and only if $A\mathbf{d} = 0$ and $\mathbf{d} \geq \mathbf{0}$.

In [4]:
```
pass
```

Problem 5: Multi-objective linear optimization

Solve the following program :

$$Objective : max(\{3x + 4y, 4z, y + z\})$$

$$s.t. \begin{array}{r} x + 2y - 4z \leq 7 \\ 3x - y + 2z \geq 2 \\ x - y + 3z \leq 2 \\ x, y, z \geq 0 \end{array}$$

In [5]:
```
pass
```

Problem 6: Hamtaro factory (part 2)

After finding the recipe for the Hamtaro snack, he then starts hiring a worker to work for his sweatshop.

- Initially, he has 50 hamster workers in the factory. However, due to substandard working conditions,
- 10% of the worker ~~die~~ resign every month.

Despite that, Hamtaro does not care about this problem and just hire new workers to fulfill the factory's demand. Before working in the factory, the newly hired hamster has to undergo training for one month to become a skilled worker,

- of which 40% of the hamsters dropped out before the training finishes as they realize how terrible the Hamtaro factory is.
- The salary for each hamster worker is 8,000 THB per month,
- and it cost 500 THB to train each hamster.

As Hamtaro predicted the number of required workers each month, how many hamsters should he hire each month to satisfy the factory's demand? Formulate the problem as a linear program and solve for an optimal solution.

Month	1	2	3	4	5	6
Amount of required factory worker	40	60	80	40	100	90

Note : The optimal solution does not have to be an integer.

Parameters:

- 10% of the worker ~~die~~ resign every month.
- 40% of the hamsters dropped out before the training finishes
- salary 8,000 THB per month
- 500 THB to train each hamster

Decision Variables:

- H_t : new hired in month t
  - 40% drop
  - 60% pass: become skilled worker in month t+1
- W_t : skilled worker in month t
  - 10% die
  - 90% alive: still be skilled worker in month t+1

$W_t = 0.9W_{t-1} + 0.6H_{t-1}\ ; W_1 = 50$
$W_t = 0.9^{t-1}50 + 0.6(\sum_{n=1}^{t-1} 0.9^{t-1-n} H_n)$

Standard Form:

$$Objective:\ min\Big[\sum_{t=1}^{6}(8000W_t + 500H_t)\Big]$$

$$\begin{aligned} W_1 - e_0 &= 50 \\ W_1 - e_1 &= 40 \\ W_2 - e2 &= 60 \\ W_3 - e3 &= 80 \\ W_4 - e4 &= 40 \\ W_5 - e5 &= 100 \\ W_6 - e6 &= 90 \\ W_2 - 0.9W_1 - 0.6H_1 &= 0 \\ W_3 - 0.9W_2 - 0.6H_2 &= 0 \\ W_4 - 0.9W_3 - 0.6H_3 &= 0 \\ W_5 - 0.9W_4 - 0.6H_4 &= 0 \\ W_6 - 0.9W_5 - 0.6H_5 &= 0 \\ H_t, W_t, e_t &\geq 0 \end{aligned}$$

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{aligned}$$

$$where \quad \mathbf{c} = [\ldots], \quad \mathbf{A} = [\ldots], \quad \mathbf{x} = [\ldots], \quad \mathbf{b} = [\ldots]$$

In [6]:

```
import numpy as np
from scipy.optimize import linprog

c_T = np.array([
    500, 500, 500, 500, 500, 500,  # H1 to H6
    8000, 8000, 8000, 8000, 8000, 8000,  # W1 to W6
    0, 0, 0, 0, 0, 0, 0  # e0 to e6
])
```

```
A = np.array([
    # [H1, H2, H3, H4, H5, H6,   W1, W2, W3, W4, W5, W6,   e0, e1, e2, e3
    [0, 0, 0, 0, 0, 0,   1, 0, 0, 0, 0, 0,   -1, 0, 0, 0, 0, 0, 0],  # W1
    [0, 0, 0, 0, 0, 0,   1, 0, 0, 0, 0, 0,   0, -1, 0, 0, 0, 0, 0],  # W1
    [0, 0, 0, 0, 0, 0,   0, 1, 0, 0, 0, 0,   0, 0, -1, 0, 0, 0, 0],  # W2
    [0, 0, 0, 0, 0, 0,   0, 0, 1, 0, 0, 0,   0, 0, 0, -1, 0, 0, 0],  # W3
    [0, 0, 0, 0, 0, 0,   0, 0, 0, 1, 0, 0,   0, 0, 0, 0, -1, 0, 0],  # W4
    [0, 0, 0, 0, 0, 0,   0, 0, 0, 0, 1, 0,   0, 0, 0, 0, 0, -1, 0],  # W5
    [0, 0, 0, 0, 0, 0,   0, 0, 0, 0, 0, 1,   0, 0, 0, 0, 0, 0, -1],  # W6
    [-0.6, 0, 0, 0, 0, 0,   -0.9, 1, 0, 0, 0, 0,   0, 0, 0, 0, 0, 0, 0],
    [0, -0.6, 0, 0, 0, 0,   0, -0.9, 1, 0, 0, 0,   0, 0, 0, 0, 0, 0, 0],
    [0, 0, -0.6, 0, 0, 0,   0, 0, -0.9, 1, 0, 0,   0, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, -0.6, 0, 0,   0, 0, 0, -0.9, 1, 0,   0, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, -0.6, 0,   0, 0, 0, 0, -0.9, 1,   0, 0, 0, 0, 0, 0, 0]
])

b = np.array([
    50,
    40,
    60,
    80,
    40,
    100,
    90,
    0,
    0,
    0,
    0,
    0
])

bounds = [(0, None)] * len(c_T)

result = linprog(c=c_T, A_eq=A, b_eq=b, bounds=bounds, method='highs')

print("{}\n\nOptimal value is {} \nOptimal solution is {}".format(result,
```

```
        message: Optimization terminated successfully. (HiGHS Status 7: Op
timal)
        success: True
         status: 0
            fun: 3679500.0
              x: [ 2.500e+01  4.333e+01 ...  0.000e+00 -0.000e+00]
            nit: 0
          lower:  residual: [ 2.500e+01  4.333e+01 ...  0.000e+00
                             -0.000e+00]
                 marginals: [ 0.000e+00  0.000e+00 ...  1.603e+04
                              0.000e+00]
          upper:  residual: [       inf        inf ...        inf
                                    inf]
                 marginals: [ 0.000e+00  0.000e+00 ...  0.000e+00
                              0.000e+00]
          eqlin:  residual: [ 0.000e+00  0.000e+00  0.000e+00  0.000e+00
                              0.000e+00  0.000e+00  0.000e+00  0.000e+00
                              0.000e+00  0.000e+00  0.000e+00  0.000e+00]
                 marginals: [ 7.250e+03 -0.000e+00  8.083e+03  1.536e+04
                             -0.000e+00  1.603e+04 -0.000e+00 -8.333e+02
                             -8.333e+02  7.250e+03 -8.333e+02  8.000e+03]
        ineqlin:  residual: []
                 marginals: []
 mip_node_count: 0
 mip_dual_bound: 0.0
        mip_gap: 0.0

Optimal value is 3679500.0
Optimal solution is [ 25.          43.33333333   0.          58.66666667
0.
   0.          50.          60.          80.          72.
 100.          90.           0.          10.           0.
   0.          32.           0.          -0.        ]
```

Conclusion:

Optimal value is: 3,679,500 | Month | 1 | 2 | 3 | 4 | 5 | 6 | |:-:|:-:|:-:|:-:|:-:|:-:|:-:| | New hire in each month | 25 | 43.33 | 0 | 58.67 | 0 | 0 |

Problem 7: l_1 regression

There are some special non-linear problems that could be transformed into a linear program. Absolute value is on one of them.

Assuming that $\forall j, c_j > 0$, the program

$$Objective: min(c_1|x_1| + c_2|x_2| + \ldots + c_n|x_n|)$$

$$s.t.\ a_{i,1}x_1 + a_{i,2}x_2 + \ldots a_{i,m}x_n \geq b \quad for\ \ i = 1, 2, \ldots, m$$

could be transformed into a linear program. To transform the program above, we write :

$$x_j = x_j^+ - x_j^-$$

and replace $|x_j|$ into $x_j^+ + x_j^-$ then add $x_j^+, x_j^- \geq 0$. Therefore, the linear program for the problem is :

$$Objective : min(c_1(x_1^+ + x_1^-) + c_2(x_2^+ + x_2^-) + \ldots + c_n(x_n^+ + x_n^-))$$

$$s.t. \quad \begin{aligned} a_{i,1}(x_1^+ - x_1^-) + a_{i,2}(x_2^+ - x_2^-) + \ldots a_{i,m}(x_n^+ - x_n^-) \geq b \\ \forall j,\ x_j^+, x_j^- \geq 0 \end{aligned}$$

Being able to solve a linear program for absolute values allow us to solve new problems, of which one of them is a l_1 regression.

Consider the following datapoints :

In [7]:
```
import numpy as np
x = np.array([0.1, -1   , -0.4,  2.3, 1.1, 3.2, 1, 4.1, -1.2, 0.9, 5, 0, 7
y = np.array([2    , 1.2, 0.7  , 4   , 3   , 5   , 2, 3   , 0   , 25  , 6,
```

As you have already learned in COM ENG MATH I class, we could find a line that best fit these datapoints by using the least square method, which could be written in a mathematical program shown below :

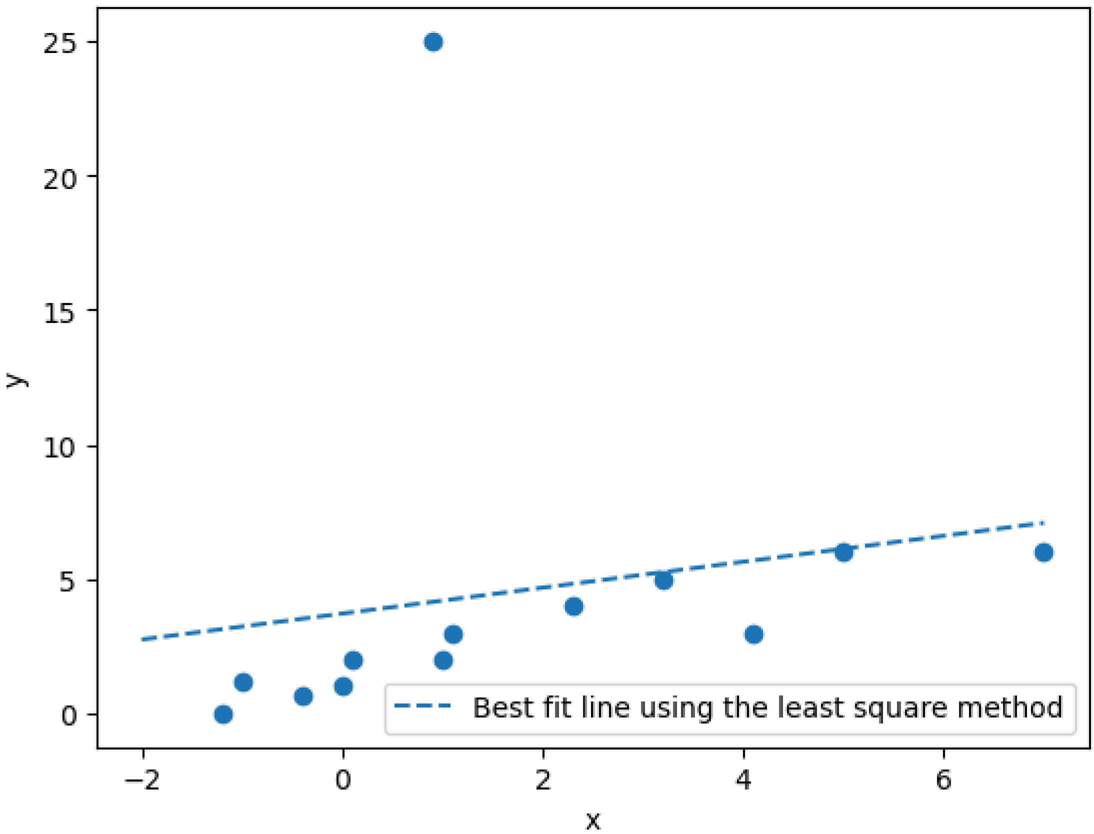
Decision variable β_1, β_0

$$Objective : min(\sum_{i=1}^{N}(y_i - (\beta_1 x_i + \beta_0))^2)$$

$$s.t.\ \beta_1, \beta_0 \in R$$

In [8]:
```
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
reg = LinearRegression().fit(x.reshape(-1, 1), y.reshape(-1, 1))
beta_1, beta_0 = (reg.coef_)[0,0], reg.intercept_[0]

x_pred = np.linspace(-2, 7, 100)
y_pred = beta_1 * x_pred + beta_0
plt.plot(x_pred, y_pred, '--', label = 'Best fit line using the least squ
plt.scatter(x, y)
plt.legend()
plt.xlabel('x')
plt.ylabel('y')
plt.show()
```



From the result above, by using the least square method, the line is not properly fit when the outliers are in the data. Therefore, in this situation, l_1 regression is often used as an alternative.

A mathematical program for l_1 regression is:

Decision variable β_1, β_0

$$Objective : min(\sum_{i=1}^{N} |y_i - (\beta_1 x_i + \beta_0)|)$$

$$s.t.\ \beta_1, \beta_0 \in R$$

Find β_1, β_0 using l_1 regression by reformulating the problem as a linear program, and compare the result with the least square method by plotting the line generated l_1 regression. Which one is better, and why?

WARNING : Be careful.

In [9]:
```
pass
```

Problem 8: Duality Theorem

Corresponding to a given **primal form** linear programming problem

$$Objective : max(c^T x)$$

$$s.t.\begin{array}{r} Ax \leq b \\ x \geq 0 \end{array}$$

there is another problem called **dual form** as follow:

$$Objective: min(b^T y)$$

$$s.t.\begin{array}{r} A^T y \geq c \\ y \geq 0 \end{array}$$

The **strong duality theorem** says that the objective of the two LPs will be equal.

In this problem we will use this property to convert between forms and show the potential usefulness of duality.

8.1

Solve the following primal form LP :

$$Objective: min(3x_1 - 2x_2 + 4x_3)$$

$$s.t.\begin{array}{r} -2x_1 + 5x_2 - 4x_3 \leq -7 \\ -6x_1 - x_2 + 3x_3 \leq -4 \\ 7x_1 + 2x_2 + x_3 \leq 10 \\ 1x_1 - 2x_2 - 5x_3 \leq -3 \\ -2x_1 + 7x_2 - 2x_3 \leq -2 \\ x_1, x_2, x_3 \geq 0 \end{array}$$

8.2

Compare the primal and dual form, which one take more iterations to solve? What might have caused this? Discuss the potential advantages of this technique in real world problems.

Ans:

8.3 Proving **weak duality theorem**

The weak duality theorem states that the dual will be an upper-bound of the primal. In other words,

If $x \in R^n$ is a feasible solution (not necessary optimal) for the primal and $y \in R^m$ is a feasible solution for the dual, then

$$c^T x \leq b^T y$$

Prove the weak duality theorem

(Hint: if x and y are feasible solutions, the constrains must be true)

In [10]:
```python
pass
```

Additional info about Duality (optional)

The strong duality can be shown from the weak duality if we use the simplex method or can be proved using Farkas's Lemma. See TA's solution for the proof of the strong duality theorem.

For LPs, the strong duality always hold. For general optimization problems, only weak duality applies.

The duality theorem provides another way to look into a particular problem and can be a powerful tool for problem interpretation.

Let's consider a very trivial problem to illustrate this.

Primal problem (max)

A tree can be used to produce 3 chairs or 2 tables. A chair sells for 30. A table sells for 40. Find the optimal revenue for a carpenter using this tree.

Let c and t be the amount of chairs and tables to produce, respectively. \

$$Objective: max(30c + 40t)$$

$$s.t. \quad \begin{aligned} \frac{1}{3}c + \frac{1}{2}t \leq 1 \\ c, t \geq 0 \end{aligned}$$

Dual problem (min)

A business man wants to buy the tree from the carpenter. Find the optimal price for the businessman. In this case, the constraint dictates that the price should be high enough for the carpenter to sell if he were to fully make the tree into a chair or a table.

Let p be the price of a tree that the businessman should pay for. \

$$Objective: min(1p)$$

$$s.t. \quad \begin{aligned} \frac{1}{3}p \geq 30 \\ \frac{1}{2}p \geq 40 \\ p \geq 0 \end{aligned}$$

The conversion between Primal and Dual provided here is very specific. However, there is a more generic way to convert between the two forms which make it applicable to more types of problems. For more details on duality see Chapter 6 in the book.

In [11]:

```
pass
```