# Karen Loscocco

## ISYE 4133 Assignment 3

## 1 (a) and (b) and (c)

In [1]:
```
from sympy import *

c = Matrix([-1,2,0,1,3]).T
A = Matrix([[1,-1,2,-1,0],[2,0,1,-1,1]])
b = Matrix([0,1])
```

***Original LP:***

min $\begin{bmatrix} -1 & 2 & 0 & 1 & 3 \end{bmatrix} x$

$$\begin{bmatrix} 1 & -1 & 2 & -1 & 0 \\ 2 & 0 & 1 & -1 & 1 \end{bmatrix} x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$x \geq 0$

For basis: $B_1 = \{1, 4\}$

In [2]:
```
basis = [0,3]
Ab1 = A[:,basis]
cb1 = c[:,basis].T
y = Inverse(Ab1.T)*cb1
```

$A_{B_1}$ = $\begin{bmatrix} 1 & -1 \\ 2 & -1 \end{bmatrix}$

$c_{B_1}$ = $\begin{bmatrix} -1 \\ 1 \end{bmatrix}$

$y_{B_1} = (A_{B_1}^{-1})^T c_{B_1} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

In [3]:
```
newc = (c.T - A.T*y).T
newcc = (b.T*y)[0]
newa = Inverse(Ab1)*A
newb = Inverse(Ab1)*b
```

New Cost Function: $(c^T - A^T y)^T = \begin{bmatrix} 0 & 1 & 2 & 0 & 3 \end{bmatrix}$

New Constraints:

$(A_{B_1}^{-1})A = \begin{bmatrix} 1 & 1 & -1 & 0 & 1 \\ 0 & 2 & -3 & 1 & 1 \end{bmatrix}$

$(A_{B_1}^{-1})b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

***LP for basis: {1,4}***

min $\begin{bmatrix} 0 & 1 & 2 & 0 & 3 \end{bmatrix} x + 0$

$\begin{bmatrix} 1 & 1 & -1 & 0 & 1 \\ 0 & 2 & -3 & 1 & 1 \end{bmatrix} x$ = $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$x \geq 0$

***This basis is feasible and optimal.***
Solution: $x = (1, 0, 0, 1, 0)$and objective value is $0$.
All elements of the cost function are positive. Therefore, there is no need to continue the simplex method because already optimal.

For basis: $B_2 = \{3, 5\}$

In [4]:
```
basis = [2,4]
Ab1 = A[:,basis]
cb1 = c[:,basis].T
y = Inverse(Ab1.T)*cb1
```

$A_{B_1} = \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix}$

$c_{B_1} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$

$y_{B_1} = (A_{B_1}^{-1})^T c_{B_1} = \begin{bmatrix} -\frac{3}{2} \\ 3 \end{bmatrix}$

In [5]:
```
newc = (c.T - A.T*y).T
newcc = (b.T*y)[0]
newa = Inverse(Ab1)*A
newb = Inverse(Ab1)*b
```

New Cost Function: $(c^T - A^T y)^T = \begin{bmatrix} -\frac{11}{2} & \frac{1}{2} & 0 & \frac{5}{2} & 0 \end{bmatrix}$

New Constraints: $(A_{B_1}^{-1})A = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\ \frac{3}{2} & \frac{1}{2} & 0 & -\frac{1}{2} & 1 \end{bmatrix}$

$(A_{B_1}^{-1})b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

***LP for basis: {3,5}***

min $\begin{bmatrix} -\frac{11}{2} & \frac{1}{2} & 0 & \frac{5}{2} & 0 \end{bmatrix} x + 3$

$\begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 \\ \frac{3}{2} & \frac{1}{2} & 0 & -\frac{1}{2} & 1 \end{bmatrix} x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$x \geq 0$

***This basis is feasible but not optimal***

Pick $x_1$ to include in basis:
$min(\frac{0}{1/2}, \frac{1}{3/2}) = 0$
So $x_3$ leaves the basis

Now basis is {1,5}

In [6]:
```
basis = [0,4]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & -5 & 11 & -3 & 0 \end{bmatrix} x + 3$

$\begin{bmatrix} 1 & -1 & 2 & -1 & 0 \\ 0 & 2 & -3 & 1 & 1 \end{bmatrix} x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$x \geq 0$

# 2 (a)

In [7]:
```
c = Matrix([-1,-3,-2]).T
A = Matrix([[1,-2,1],[-1,3,-2]])
b = Matrix([2,-3])
```

Original LP:

min $\begin{bmatrix} -1 & -3 & -2 \end{bmatrix} x$

$$\begin{bmatrix} 1 & -2 & 1 \\ -1 & 3 & -2 \end{bmatrix} x = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$$

$x \geq 0$

Phase1: Introduce new variables and iterate until obtain first basis to solve original LP

In [8]:
```
getfirstb_c = Matrix([0,0,0,1,1]).T
getfirstb_A = A.col_insert(3, Matrix([[1, 0],[0,1]]))
```

min $\begin{bmatrix} 0 & 0 & 0 & 1 & 1 \end{bmatrix} x$

$$\begin{bmatrix} 1 & -2 & 1 & 1 & 0 \\ -1 & 3 & -2 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$$

$x \geq 0$

Starting basis: {4,5}

In [9]:
```
basis = [3,4]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} -\frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \end{bmatrix} x + 0$

$$\begin{bmatrix} -\frac{1}{3} & 1 & -\frac{2}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} & 1 & \frac{2}{3} \end{bmatrix} x = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

$x \geq 0$

Pick $x_2$ to include in basis:
$min(\frac{-3}{3}) = -1$
So $x_5$ leaves the basis

Now basis is {2,4}

In [10]:
```
basis = [1,3]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} -\frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \end{bmatrix} x + 0$

$$\begin{bmatrix} -\frac{1}{3} & 1 & -\frac{2}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} & 1 & \frac{2}{3} \end{bmatrix} x = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

$x \geq 0$

Pick $x_1$ to include in basis:
$min(\frac{0}{1/3}) = 0$
So $x_4$ leaves the basis

Now basis is {1,2}

Phase2: Iterate until the stopping conditions are met

Starting basis for original LP: {1,2}

In [11]:
```
c = Matrix([-1,-3,-2]).T
A = Matrix([[1,-2,1],[-1,3,-2]])
b = Matrix([2,-3])
```

Original LP:

min $\begin{bmatrix} -1 & -3 & -2 \end{bmatrix} x$

$$\begin{bmatrix} 1 & -2 & 1 \\ -1 & 3 & -2 \end{bmatrix} x = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$$

$$x \geq 0$$

In [12]:
```
basis = [0,1]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & 0 & -6 \end{bmatrix} x + 3$

$$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix} x = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_3$ to include in basis:
All ratios are negative, so $x_3$ can be made as large as desired.
This means the ***solution is unbounded.***

Verify in Gurobi:

In [13]:
```
from gurobipy import *

def printOptimal(m):
    if m.Status == GRB.OPTIMAL:
        print('Variable Values:')
        for v in m.getVars():
            print(v.VarName, v.X)

        print('\nObjective Value: {}\n'.format(str(m.objVal)))
        print('\nRuntime: {}\n'.format(m.Runtime))
```

In [14]:
```
import numpy as np
c = np.array([-1,-3,-2])
A = np.array([[1,-2,1],[-1,3,-2]])
b = np.array([2,-3])

m = Model()

X = m.addVars(3, name = 'X', lb = 0)

m.setObjective(np.dot(c,X.values()), GRB.MINIMIZE)

for i in range(A.shape[0]):
    m.addConstr(np.dot(A[i,:],X.values()) == b[i])

m.optimize()

printOptimal(m)
```

```
Academic license - for non-commercial use only
Optimize a model with 2 rows, 3 columns and 6 nonzeros
Coefficient statistics:
  Matrix range     [1e+00, 3e+00]
  Objective range  [1e+00, 3e+00]
  Bounds range     [0e+00, 0e+00]
  RHS range        [2e+00, 3e+00]
Presolve time: 0.00s
Presolved: 2 rows, 3 columns, 6 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0   -9.0000000e+30   1.750000e+30   9.000000e+00      0s

Solved in 2 iterations and 0.01 seconds
Unbounded model
```

## 2 (b)

In [15]:
```
c = Matrix([-27,-2,6]).T
A = Matrix([[2,4,2],[-1,-3,-2]])
b = Matrix([2,0])
```

Original LP:

$$\min \begin{bmatrix} -27 & -2 & 6 \end{bmatrix} x$$

$$\begin{bmatrix} 2 & 4 & 2 \\ -1 & -3 & -2 \end{bmatrix} x = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$x \geq 0$$

Phase1: Introduce new variables and iterate until obtain first basis to solve original LP

In [16]:
```
getfirstb_c = Matrix([0,0,0,1,1]).T
getfirstb_A = A.col_insert(3, Matrix([[1, 0],[0,1]]))
```

$$\min \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \end{bmatrix} x$$

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 0 \\ -1 & -3 & -2 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$x \geq 0$$

starting basis: {4,5}

In [17]:
```
basis = [3,4]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} -1 & -1 & 0 & 0 & 0 \end{bmatrix} x + 2$

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 0 \\ -1 & -3 & -2 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$x \geq 0$

Pick $x_1$ to include in basis:
$min(\frac{2}{2}) = 1$
So $x_4$ leaves the basis

Now basis is {1,5}

In [18]:
```
basis = [0,4]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & 1 & 1 & \frac{1}{2} & 0 \end{bmatrix} x + 1$

$\begin{bmatrix} 1 & 2 & 1 & \frac{1}{2} & 0 \\ 0 & -1 & -1 & \frac{1}{2} & 1 \end{bmatrix} x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$x \geq 0$

All elements of cost function are positive so do not perform another iteration of the simplex method. The optimal solution for this altered LP is $x = (1, 0, 0, 0, 1)$and the optimal objective value is 1.
An optimal objective value of 0 was needed for the altered LP in order to have feasibility in the original LP for $x_1$,$x_2$, and $x_3$. ($x_4 + x_5 = 0 \implies x_4 = x_5 = 0$). Since the optimal objective value for the altered LP is not 0, the ***original LP is infeasible.***

Verify in Gurobi:

In [19]:
```
c = np.array([-27,-2,6])
A = np.array([[2,4,2],[-1,-3,-2]])
b = np.array([2,0])

m = Model()

X = m.addVars(3, name = 'X', lb = 0)

m.setObjective(np.dot(c,X.values()), GRB.MINIMIZE)

for i in range(A.shape[0]):
    m.addConstr(np.dot(A[i,:],X.values()) == b[i])

m.optimize()

printOptimal(m)
```

```
Optimize a model with 2 rows, 3 columns and 6 nonzeros
Coefficient statistics:
  Matrix range     [1e+00, 4e+00]
  Objective range  [2e+00, 3e+01]
  Bounds range     [0e+00, 0e+00]
  RHS range        [2e+00, 2e+00]
Presolve removed 1 rows and 3 columns
Presolve time: 0.05s

Solved in 0 iterations and 0.05 seconds
Infeasible or unbounded model
```

# 3 (a)

$x_i$ : amount of gold type $i$

max $\begin{bmatrix} 1 & 2 & 3 & 2 \end{bmatrix} x$

$\begin{bmatrix} 1 & 2 & 2 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix} x \leq \begin{bmatrix} 6 \\ 10 \end{bmatrix}$

$x \geq 0$

# 3 (b)

Standard Form:

min $\begin{bmatrix} -1 & -2 & -3 & -2 & 0 & 0 \end{bmatrix} x$

$\begin{bmatrix} 1 & 2 & 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 2 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 6 \\ 10 \end{bmatrix}$

$x \geq 0$

# 3 (c)

In [20]:
```
c = Matrix([-1,-2,-3,-2,0,0]).T
A = Matrix([[1,2,2,1,1,0],[0,1,1,2,0,1]])
b = Matrix([6,10])
```

Phase 1: Find starting basis

In [21]:
```
getfirstb_c = Matrix([0,0,0,0,0,0,1,1]).T
getfirstb_A = A.col_insert(6, Matrix([[1, 0],[0,1]]))
```

min $\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} x$

$\begin{bmatrix} 1 & 2 & 2 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 2 & 0 & 1 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 6 \\ 10 \end{bmatrix}$

$x \geq 0$

starting basis: {7,8}

In [22]:
```
basis = [6,7]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} -1 & -3 & -3 & -3 & -1 & -1 & 0 & 0 \end{bmatrix} x + 16$

$$\begin{bmatrix} 1 & 2 & 2 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 2 & 0 & 1 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 6 \\ 10 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_1$ to include in basis:
$min(\frac{6}{1}) = 6$
So $x_7$ leaves the basis

Now basis is {1,8}

In [23]:
```
basis = [0,7]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

$$\min \begin{bmatrix} 0 & -1 & -1 & -2 & 0 & -1 & 1 & 0 \end{bmatrix} x + 10$$

$$\begin{bmatrix} 1 & 2 & 2 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 2 & 0 & 1 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 6 \\ 10 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_2$ to include in basis:
$min(\frac{6}{2}, \frac{10}{1}) = 3$
So $x_1$ leaves the basis

Now basis is {2,8}

```
In [24]: basis = [1,7]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

$$\min \begin{bmatrix} \frac{1}{2} & 0 & 0 & -\frac{3}{2} & \frac{1}{2} & -1 & \frac{3}{2} & 0 \end{bmatrix} x + 7$$

$$\begin{bmatrix} \frac{1}{2} & 1 & 1 & \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ -\frac{1}{2} & 0 & 0 & \frac{3}{2} & -\frac{1}{2} & 1 & -\frac{1}{2} & 1 \end{bmatrix} x = \begin{bmatrix} 3 \\ 7 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_4$ to include in basis:
$min(\frac{3}{1/2}, \frac{7}{3/2}) = 4.66$
So $x_8$ leaves the basis

Now basis is {2,4}

In [25]:
```
basis = [1,3]
Ab = getfirstb_A[:,basis]
cb = getfirstb_c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (getfirstb_c.T - getfirstb_A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*getfirstb_A
newb = Inverse(Ab)*b
```

LP for new basis:

$$\min \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} x + 0$$

$$\begin{bmatrix} \frac{2}{3} & 1 & 1 & 0 & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 0 & 0 & 1 & -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} \end{bmatrix} x = \begin{bmatrix} \frac{2}{3} \\ \frac{14}{3} \end{bmatrix}$$

$$x \geq 0$$

Phase 2: Starting basis is {2,4}

In [26]:
```
basis = [1,3]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

$$\min \begin{bmatrix} -\frac{1}{3} & 0 & -1 & 0 & \frac{2}{3} & \frac{2}{3} \end{bmatrix} x + -\frac{32}{3}$$

$$\begin{bmatrix} \frac{2}{3} & 1 & 1 & 0 & \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 0 & 0 & 1 & -\frac{1}{3} & \frac{2}{3} \end{bmatrix} x = \begin{bmatrix} \frac{2}{3} \\ \frac{14}{3} \end{bmatrix}$$

$$x \geq 0$$

Pick $x_1$ to include in basis:
$min(\frac{2/3}{2/3}) = 1$
So $x_2$ leaves the basis

Now basis is {1,4}

In [27]:
```
basis = [0,3]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 1 & \frac{1}{2} \end{bmatrix} x + -11$

$$\begin{bmatrix} 1 & \frac{3}{2} & \frac{3}{2} & 0 & 1 & -\frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} & 1 & 0 & \frac{1}{2} \end{bmatrix} x = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

$x \geq 0$

Pick $x_3$ to include in basis:
$min(\frac{1}{3/2}, \frac{5}{1/2}) = 0.66$
So $x_1$ leaves the basis

Now basis is {3,4}

In [28]:
```
basis = [2,3]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

$$\min \begin{bmatrix} \frac{1}{3} & 1 & 0 & 0 & \frac{4}{3} & \frac{1}{3} \end{bmatrix} x + -\frac{34}{3}$$

$$\begin{bmatrix} \frac{2}{3} & 1 & 1 & 0 & \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 0 & 0 & 1 & -\frac{1}{3} & \frac{2}{3} \end{bmatrix} x = \begin{bmatrix} \frac{2}{3} \\ \frac{14}{3} \end{bmatrix}$$

$$x \geq 0$$

All elements of cost function are positive.
Therefore the optimal solution is $\bar{x} = (0, 0, \frac{2}{3}, \frac{14}{3}, 0, 0)$ and the objective value is $-\frac{34}{3}$

Gurobi:

In [29]:
```
import numpy as np
c = np.array([-1,-2,-3,-2,0,0])
A = np.array([[1,2,2,1,1,0],[0,1,1,2,0,1]])
b = np.array([6,10])
```

In [30]:
```
m = Model()

X = m.addVars(6, name = 'X', lb = 0)

m.setObjective(np.dot(c,X.values()), GRB.MINIMIZE)

for i in range(A.shape[0]):
    m.addConstr(np.dot(A[i,:],X.values()) == b[i])

m.optimize()

printOptimal(m)
```

```
Optimize a model with 2 rows, 6 columns and 9 nonzeros
Coefficient statistics:
  Matrix range     [1e+00, 2e+00]
  Objective range  [1e+00, 3e+00]
  Bounds range     [0e+00, 0e+00]
  RHS range        [6e+00, 1e+01]
Presolve removed 0 rows and 3 columns
Presolve time: 0.00s
Presolved: 2 rows, 3 columns, 5 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0   -1.2000000e+01   1.994000e+00   0.000000e+00      0s
       2   -1.1333333e+01   0.000000e+00   0.000000e+00      0s

Solved in 2 iterations and 0.01 seconds
Optimal objective -1.133333333e+01
Variable Values:
X[0] 0.0
X[1] 0.0
X[2] 0.6666666666666666
X[3] 4.666666666666667
X[4] 0.0
X[5] 0.0

Objective Value: -11.333333333333334


Runtime: 0.008266925811767578
```

# 4 (a)

Dual:
min $-3y_1 + 9y_2 + 5y_3$

$y_1 + 9y_3 + y_4 \geq 1$
$y_1 + y_2 - y_4 \geq 3$
$7y_1 - y_2 + y_4 = 1$

$y_1 \geq 0$, $y_2$ free, $y_3 \geq 0$, $y_4$ free

# 4 (b)

max $b^T y$
$A^T y \leq c$
$D^T v = d$
$y$,$v$ free

# 5 (a)

**(P)**
max $-2x_1 - 1x_2$

$$\begin{aligned} x_1 + 3x_2 + 2x_3 &\geq 5 \\ -x_1 + 4x_2 + 2x_3 &\leq 7 \end{aligned}$$

$x_1 \leq 0, x_2 \geq 0, x_3 \geq 0$

**(D)**
min $5y_1 + 7y_2$

$$\begin{aligned} y_1 - \ y_2 &\leq -2 \\ 3y_1 + 4y_2 &\geq -1 \\ 2y_1 + 2y_2 &\geq 0 \end{aligned}$$

$y_1 \leq 0, y_2 \geq 0$

Complementary slackness:
**(P)**
max $-2x_1 - 1x_2$

$$\begin{aligned} x_1 + 3x_2 + 2x_3 - s_1 &= 5 \\ -x_1 + 4x_2 + 2x_3 + s_2 &= 7 \end{aligned}$$

$x_1 \leq 0, x_2 \geq 0, x_3 \geq 0,$
$s_1, s_2 \geq 0$

**(D)**
min $5y_1 + 7y_2$

$$\begin{aligned} y_1 - \ y_2 + s_1 &= -2 \\ 3y_1 + 4y_2 - s_2 &= -1 \\ 2y_1 + 2y_2 - s_3 &= 0 \end{aligned}$$

$y_1 \leq 0, y_2 \geq 0,$
$s_1, s_2, s_3 \geq 0$

# 5 (b)

**(P)**
min $-5x_1 - 7x_2 - 5x_3$

$$\begin{aligned} x_1 + 2x_2 + 3x_3 &\geq 3 \\ 4x_1 + 5x_2 + 6x_3 &\leq 9 \\ 7x_1 + 8x_2 + 9x_3 &\geq 2 \end{aligned}$$

$$x_1, x_2, x_3 \geq 0$$

**(D)**
max $3y_1 + 9y_2 + 2y_3$

$$\begin{aligned} y_1 - 4y_2 + 7y_3 &\leq -5 \\ 2y_1 + 5y_2 + 8y_3 &\leq -7 \\ 3y_1 + 6y_2 + 9x_3 &\leq -5 \end{aligned}$$

$$y_1 \geq 0, y_2 \leq 0, y_3 \geq 0$$

Complementary slackness:

**(P)**
min $-5x_1 - 7x_2 - 5x_3$

$$\begin{aligned} x_1 + 2x_2 + 3x_3 - s_1 3 \\ 4x_1 + 5x_2 + 6x_3 + s_2 9 \\ 7x_1 + 8x_2 + 9x_3 - s_3 2 \end{aligned}$$

$$\begin{aligned} &x_1, x_2, x_3 \geq 0, \\ &s_1, s_2, s_3 \geq 0 \end{aligned}$$

**(D)**
max $3y_1 + 9y_2 + 2y_3$

$$y_1 - 4y_2 + 7y_3 + s_1 - 5$$
$$2y_1 + 5y_2 + 8y_3 + s_2 - 7$$
$$3y_1 + 6y_2 + 9x_3 + s_3 - 5$$

$$y_1 \geq 0, y_2 \leq 0, y_3 \geq 0,$$
$$s_1, s_2, s_3 \geq 0$$

# 6

In [31]:
```
c = Matrix([-10,-3,0,0,0]).T
A = Matrix([[1,1,1,0,0],[5,2,0,1,0],[0,1,0,0,1]])
b = Matrix([4,11,4])
```

Original LP:

min $\begin{bmatrix} -10 & -3 & 0 & 0 & 0 \end{bmatrix} x$

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 5 & 2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 4 \\ 11 \\ 4 \end{bmatrix}$$

$$x \geq 0$$

Starting basis for original LP: {2,3,4}

In [32]:
```
basis = [1,2,3]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} -10 & 0 & 0 & 0 & 3 \end{bmatrix} x + -12$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & -1 \\ 5 & 0 & 0 & 1 & -2 \end{bmatrix} x = \begin{bmatrix} 4 \\ 0 \\ 3 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_1$ to include in basis:
$min(\frac{0}{1}, \frac{3}{5}) = 0$
So $x_3$ leaves the basis

Now basis is {1,2,4}

In [33]:
```
basis = [0,1,3]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & 0 & 10 & 0 & -7 \end{bmatrix} x + -12$

$$\begin{bmatrix} 1 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & -5 & 1 & 3 \end{bmatrix} x = \begin{bmatrix} 0 \\ 4 \\ 3 \end{bmatrix}$$

$$x \geq 0$$

Pick $x_5$ to include in basis:
$min(\frac{4}{1}, \frac{3}{3}) = 0$
So $x_4$ leaves the basis

Now basis is {1,2,5}

In [34]:
```
basis = [0,1,4]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & 0 & -\frac{5}{3} & \frac{7}{3} & 0 \end{bmatrix} x + -19$

$$\begin{bmatrix} 1 & 0 & -\frac{2}{3} & \frac{1}{3} & 0 \\ 0 & 1 & \frac{5}{3} & -\frac{1}{3} & 0 \\ 0 & 0 & -\frac{5}{3} & \frac{1}{3} & 1 \end{bmatrix} x = \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix}$$

$x \geq 0$

Pick $x_3$ to include in basis:
$min(\frac{3}{5/3}) = 1.799$
So $x_2$ leaves the basis

Now basis is {1,3,5}

In [35]:
```
basis = [0,2,4]
Ab = A[:,basis]
cb = c[:,basis].T
y = Inverse(Ab.T)*cb

newc = (c.T - A.T*y)
newcc = (b.T*y)[0]
newa = Inverse(Ab)*A
newb = Inverse(Ab)*b
```

LP for new basis:

min $\begin{bmatrix} 0 & 1 & 0 & 2 & 0 \end{bmatrix} x + -22$

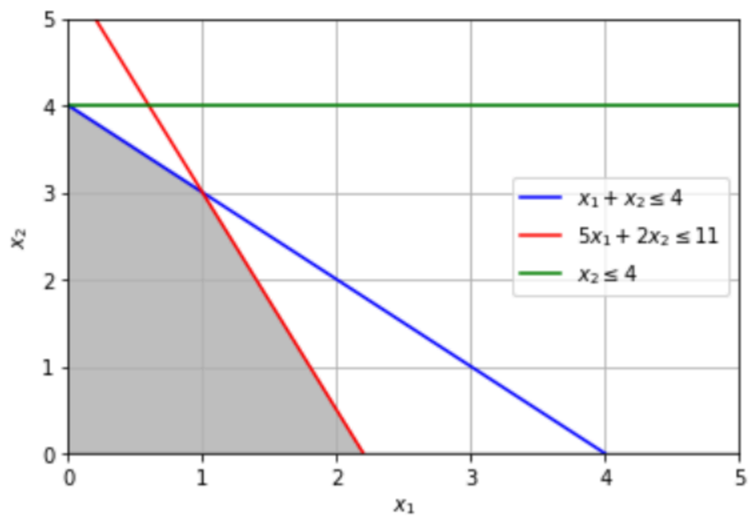
$$\begin{bmatrix} 1 & \frac{2}{5} & 0 & \frac{1}{5} & 0 \\ 0 & \frac{3}{5} & 1 & -\frac{1}{5} & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} \frac{11}{5} \\ \frac{9}{5} \\ 4 \end{bmatrix}$$

$x \geq 0$

All elements of cost function are positive.
Therefore the optimal solution is $\bar{x} = (\frac{11}{5}, 0, \frac{9}{5}, 0, 4)$ and the objective value is $-22$

In [36]:
```
import matplotlib.pyplot as plt

x = np.linspace(0,5,100)

plt.plot(x, 4-x, '-b', label=r'$x_1+x_2 \leq4$')
plt.plot(x, (11/2)-(5/2)*x , '-r', label=r'$5x_1+2x_2 \leq 11$')
plt.plot(x, 4 + 0*x, '-g', label=r'$x_2 \leq 4$')

plt.xlabel(r'$x_1$')
plt.ylabel(r'$x_2$')
plt.legend()
plt.grid()
plt.xlim(0,5)
plt.ylim(0,5)

plt.fill_between(x ,np.minimum(4-x,(11/2)-(5/2)*x), color='grey', alph
a='0.5')
plt.show()
```

```
<Figure size 640x480 with 1 Axes>
```



$x_1 + x_2 \leq 4$
$5x_1 + 2x_2 \leq 11$
$x_2 \leq 4$
$x_1$
$x_2$
0
1
2
3
4
5

Extreme points: $(\frac{11}{5}, 0), (1, 3), (0, 4), (0, 0)$

Based on the above simplex calculation:
Start at $(0, 4)$
Move to $(0, 4)$
Move to $(1, 3)$
Move to $(\frac{11}{5}, 0)$