# Homework 1 -- Jae Yul Woo (7466-8871-96)

## Problem (a)

In [8]:

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

# Given matrices
s = np.array([0,0,0,0,0,0,0,0,0,1,1,1])
d = np.array([[0,0,0,1,1,0,0,0,0,1,0,0],
              [0,0,0,0,0,1,1,0,0,0,1,0],
              [0,0,0,0,0,0,0,1,1,0,0,1]])
c = np.array([[-1,0,0,1,0,0,1,0,1,0,0,0],
              [0,-1,0,0,1,1,0,0,1,0,0,0],
              [0,0,-1,0,1,0,1,1,0,0,0,0],
              [1,1,4,0,0,0,0,0,0,0,0,0]])
db = np.array([2.05,2.05,1.5])
cb = np.array([0,0,0,10])

print(s)
print(d)
print(c)
```

```
[0 0 0 0 0 0 0 0 0 1 1 1]
[[0 0 0 1 1 0 0 0 0 1 0 0]
 [0 0 0 0 0 1 1 0 0 0 1 0]
 [0 0 0 0 0 0 0 1 1 0 0 1]]
[[-1  0  0  1  0  0  1  0  1  0  0  0]
 [ 0 -1  0  0  1  1  0  0  1  0  0  0]
 [ 0  0 -1  0  1  0  1  1  0  0  0  0]
 [ 1  1  4  0  0  0  0  0  0  0  0  0]]
```

In [12]:

```
res = linprog(s, A_eq=d, b_eq=db, A_ub=c, b_ub=cb,
              options={"disp": True,'tol': 0.001},method = 'simplex')
print(res.x)
```

```
Optimization terminated successfully.
         Current function value: -0.000000
         Iterations: 8
[2.1  2.1  1.45 2.05 0.   2.05 0.   1.45 0.05 0.   0.   0.  ]
```

## Problem (b)

In [0]:

```
#!pip install pyomo
```

```
Collecting pyomo
  Downloading https://files.pythonhosted.org/packages/e4/df/3f4a54d494
d429102c035308168bfd71aa0fac31832385ab356cb44560df/Pyomo-5.6.1-py3-non
e-any.whl (https://files.pythonhosted.org/packages/e4/df/3f4a54d494d42
9102c035308168bfd71aa0fac31832385ab356cb44560df/Pyomo-5.6.1-py3-none-a
ny.whl) (2.1MB)
    100% |████████████████████████████████| 2.1MB 7.6MB/s eta 0:00:01
 5% |██                              | 122kB 1.8MB/s eta 0:00:02
Collecting appdirs (from pyomo)
  Downloading https://files.pythonhosted.org/packages/56/eb/810e700ed1
349edde4cbdc1b2a21e28cdf115f9faf263f6bbf8447c1abf3/appdirs-1.4.3-py2.p
y3-none-any.whl (https://files.pythonhosted.org/packages/56/eb/810e700
ed1349edde4cbdc1b2a21e28cdf115f9faf263f6bbf8447c1abf3/appdirs-1.4.3-py
2.py3-none-any.whl)
Collecting PyUtilib>=5.6.5 (from pyomo)
  Downloading https://files.pythonhosted.org/packages/43/04/9174c992ab
7b5d5c9c29c0dce3ffbe2440dcfaf054a83b536f8253ce8384/PyUtilib-5.6.5-py2.
py3-none-any.whl (https://files.pythonhosted.org/packages/43/04/9174c9
92ab7b5d5c9c29c0dce3ffbe2440dcfaf054a83b536f8253ce8384/PyUtilib-5.6.5-
py2.py3-none-any.whl) (250kB)
    100% |████████████████████████████████| 256kB 4.2MB/s ta 0:00:011
Requirement already satisfied: ply in /anaconda3/lib/python3.6/site-pa
ckages (from pyomo) (3.11)
Requirement already satisfied: six>=1.4 in /anaconda3/lib/python3.6/si
te-packages (from pyomo) (1.11.0)
Requirement already satisfied: nose in /anaconda3/lib/python3.6/site-p
ackages (from PyUtilib>=5.6.5->pyomo) (1.3.7)
Installing collected packages: appdirs, PyUtilib, pyomo
Successfully installed PyUtilib-5.6.5 appdirs-1.4.3 pyomo-5.6.1
```

In [3]:

```
from pyutilib.services import register_executable, registered_executable
register_executable(name='glpsol')
```

In [4]:

```
from pyomo.environ import *
from pyomo.opt import SolverFactory
model = ConcreteModel()

X = ['x1','x2','x3']; F = ['f11','f12','f21','f22','f31','f32']; S = ['s1','s2','s3

model.x = Var(X,within=NonNegativeReals)
model.f = Var(F,within=NonNegativeReals)
model.s = Var(S,within=NonNegativeReals)

model.obj = Objective(expr=model.s['s1']+model.s['s2']+model.s['s3'], sense=minimize
```

In [5]:

```
model.d1 = Constraint(expr=model.f['f11']+model.f["f12"]+model.s['s1'] == 2.05)
model.d2 = Constraint(expr=model.f['f21']+model.f["f22"]+model.s['s2'] == 2.05)
model.d3 = Constraint(expr=model.f['f31']+model.f["f32"]+model.s['s3'] == 1.5)
model.c1 = Constraint(expr=-model.x['x1']+model.f["f11"]+model.f["f22"]+model.f['f32
model.c2 = Constraint(expr=-model.x['x2']+model.f["f12"]+model.f["f21"]+model.f['f32
model.c3 = Constraint(expr=-model.x['x3']+model.f["f12"]+model.f["f22"]+model.f['f31
model.budget = Constraint(expr=model.x['x1']+model.x['x2']+4*model.x['x3'] <= 10)
```

In [6]:

```
opt = SolverFactory("glpk")
results = opt.solve(model)

print('Objective Value')
print(str(value(model.obj))+'\n')
for v in model.component_objects(Var, active=True):
    print ("Variable component object",v)
    for index in v:
        print ("    ", index, v[index].value)
```

```
Objective Value
0.0

Variable component object x
    x1 2.1
    x2 2.1
    x3 1.45
Variable component object f
    f21 2.05
    f22 0.0
    f12 0.0
    f31 1.45
    f11 2.05
    f32 0.0499999999999998
Variable component object s
    s1 0.0
    s2 0.0
    s3 0.0
```