

```

# -*- coding: utf-8 -*-
"""
Created on Fri Jan 18 14:18:25 2019

@author: ifrommer
"""

from pulp import *
from amply import *

def diet_lp(dat_file_name):
    diet_lp_model = LpProblem(name='Diet LP',sense=LpMinimize)

    # Names within must match names in .dat file
    data = Amply("""
    set food_items;
    set nutrients;
    param objective{food_items};
    param reqs{nutrients};
    param supply{food_items,nutrients};
    param max_for_item{food_items};
    """)

    # dat_file_name = 'wardRoom1.dat'
    data.load_file(open(dat_file_name))

    # diagnostic to see if things loaded and get used to indexing
    for nutr in data.nutrients:
        print('required ',nutr,data.reqs[nutr])
        for food in data.food_items:
            print('supply ',nutr,food,data.supply[food][nutr])

    # Dec vars
    x = LpVariable.dicts('x',data.food_items,0)

    # objective
    diet_lp_model += lpSum(data.objective[i] * x[i] for i in data.food_items)

    # constraints - loop through reqs adding an lpSum for each
    for i in data.nutrients:
        constr_LHS = lpSum(data.supply[j][i] * x[j] for j in data.food_items)
        diet_lp_model += constr_LHS >= data.reqs[i], i

    # upper limits
    for i in data.food_items:
        diet_lp_model += x[i] <= data.max_for_item[i], i

```

```

# the rest as before...
# Print formulated model and optionally write it to a file
print(diet_lp_model)
diet_lp_model.writeLP('diet_lp_model.txt')

# Solve
result = diet_lp_model.solve(GLPK(msg=False))

# Print results
print("Status: ", LpStatus[result])
for variable in diet_lp_model.variables():
    print(str(variable).ljust(12), ' = ', value(variable))
print(f"Objective value = {value(diet_lp_model.objective):8.2f}")

return diet_lp_model
### wardroom_1.dat
set food_items := Noodles TurkeyDivan PotatoSalad Beets CongoBars;

set nutrients := Protein      Iron      Niacin      Thiamin      VitaminC;

param      objective :=
    Noodles      5000
    TurkeyDivan  5000
    PotatoSalad  7900
    Beets        300
    CongoBars    14300;

param      reqs :=
    Protein      63000
    Iron         10
    Niacin        15
    Thiamin       1
    VitaminC      50;

param      supply :
               Protein  Iron    Niacin  Thiamin  VitaminC :=
    Noodles      5000    1.1     1.4    0.18    0.0
    TurkeyDivan  29300   1.8     5.4    0.06    0.0
    PotatoSalad  5300    0.5     0.9    0.06    10.0
    Beets3000    2.2     0.5     0.07   28.0
    CongoBars    4000    1.2     0.6    0.15    3.0;

param      max_for_item :=
    Noodles      3
    TurkeyDivan  3
    PotatoSalad  2
    Beets        1

```

```

CongoBars 1;
### Solns
"""
Diet_LP:
MINIMIZE
300.0*x_Beets + 14300.0*x_CongoBars + 5000.0*x_Noodles + 7900.0*x_PotatoSalad
+ 5000.0*x_TurkeyDivan + 0.0
SUBJECT TO
Protein: 3000 x_Beets + 4000 x_CongoBars + 5000 x_Noodles + 5300 x_PotatoSalad
+ 29300 x_TurkeyDivan >= 63000

Iron: 2.2 x_Beets + 1.2 x_CongoBars + 1.1 x_Noodles + 0.5 x_PotatoSalad
+ 1.8 x_TurkeyDivan >= 10

Niacin: 0.5 x_Beets + 0.6 x_CongoBars + 1.4 x_Noodles + 0.9 x_PotatoSalad
+ 5.4 x_TurkeyDivan >= 15

Thiamin: 0.07 x_Beets + 0.15 x_CongoBars + 0.18 x_Noodles + 0.06 x_PotatoSalad
+ 0.06 x_TurkeyDivan >= 1

VitaminC: 28 x_Beets + 3 x_CongoBars + 10 x_PotatoSalad >= 50

Noodles: x_Noodles <= 3

TurkeyDivan: x_TurkeyDivan <= 3

PotatoSalad: x_PotatoSalad <= 2

Beets: x_Beets <= 1

CongoBars: x_CongoBars <= 1

VARIABLES
x_Beets Continuous
x_CongoBars Continuous
x_Noodles Continuous
x_PotatoSalad Continuous
x_TurkeyDivan Continuous

Status: Optimal
x_Beets = 1.0
x_CongoBars = 0.666667
x_Noodles = 3.0
x_PotatoSalad = 2.0
x_TurkeyDivan = 2.83333
Objective value = 54799.99
"""

```

```

%% wardroom_2.dat
set food_items := oats chicken eggs milk cake beans;

set nutrients := calories protein calcium;

param objective :=
    oats 30
    chicken 240
    eggs 130
    milk 90
    cake 200
    beans 60;

param          reqs :=
    calories 2000
    protein 55
    calcium 800;

param    supply :
                                calories protein calcium :=
    oats 110 4 2
    chicken 205 32 12
    eggs 160 13 54
    milk 160 8 285
    cake 420 4 22
    beans 260 14 80;

param    max_for_item :=
    oats 4
    chicken 3
    eggs 2
    milk 8
    cake 2
    beans 2;
%% Solns
"""
Diet_LP:
MINIMIZE
60.0*x_beans + 200.0*x_cake + 240.0*x_chicken + 130.0*x_eggs + 90.0*x_milk +
30.0*x_oats + 0.0
SUBJECT TO
calories: 260 x_beans + 420 x_cake + 205 x_chicken + 160 x_eggs + 160 x_milk
+ 110 x_oats >= 2000

protein: 14 x_beans + 4 x_cake + 32 x_chicken + 13 x_eggs + 8 x_milk
+ 4 x_oats >= 55

```

```
calcium: 80 x_beans + 22 x_cake + 12 x_chicken + 54 x_eggs + 285 x_milk  
+ 2 x_oats >= 800
```

```
oats: x_oats <= 4
```

```
chicken: x_chicken <= 3
```

```
eggs: x_eggs <= 2
```

```
milk: x_milk <= 8
```

```
cake: x_cake <= 2
```

```
beans: x_beans <= 2
```

```
VARIABLES
```

```
x_beans Continuous
```

```
x_cake Continuous
```

```
x_chicken Continuous
```

```
x_eggs Continuous
```

```
x_milk Continuous
```

```
x_oats Continuous
```

```
Status: Optimal
```

```
x_beans      = 2.0
```

```
x_cake       = 1.68084
```

```
x_chicken    = 0.0
```

```
x_eggs       = 0.0
```

```
x_milk       = 2.08779
```

```
x_oats       = 4.0
```

```
Objective value = 764.07
```

```
"""
```