salaries

October 31, 2022

[ ]:

```julia
#Import JuMP package to build an optimization model
using JuMP
#Import HiGHS solver
using HiGHS

#Create a JuMP model named picframe1 that will be solved using the HiGHS solver
picframe1 = Model(HiGHS.Optimizer);

#Add the variables
@variable(picframe1,tom>= 0);
@variable(picframe1,peter>=0);
@variable(picframe1,nina>=0);
@variable(picframe1,samir>=0);
@variable(picframe1,gary>=0);
@variable(picframe1,bob>=0);
@variable(picframe1,linda>=0);
@variable(picframe1,IT>=0);
@variable(picframe1,Customer>=0);

#Add constraint

@constraint(picframe1, constarint1, tom>=30000);
@constraint(picframe1, constarint2, nina>=tom+8000);
@constraint(picframe1, constarint3, peter>=tom+8000);
@constraint(picframe1, constarint4, samir>=tom+8000);
@constraint(picframe1, constarint5, gary>=tom+peter);
@constraint(picframe1, constarint6, linda==500+gary);
@constraint(picframe1, constarint7, nina+samir>=2*(tom+peter));
@constraint(picframe1, constarint8, bob>=peter);
@constraint(picframe1, constarint9, bob>=samir);
@constraint(picframe1, constarint10, bob+peter>=75000);
@constraint(picframe1, constarint11, linda<=bob+tom);

#convert problem to convex
@constraint(picframe1,constarint12,IT>=tom);
@constraint(picframe1,constarint13,IT>=peter);
@constraint(picframe1,constarint14,IT>=nina);
```

```
@constraint(picframe1,constarint15,IT>=samir);
@constraint(picframe1,constarint16,Customer>=gary);
@constraint(picframe1,constarint17,Customer>=bob);
@constraint(picframe1,constarint18,Customer>=linda);

#objective function
@objective(picframe1,Min,IT+Customer);



print(picframe1);
```

```
Min IT + Customer
Subject to
 constarint6 : -gary + linda = 500.0
 constarint1 : tom   30000.0
 constarint2 : -tom + nina   8000.0
 constarint3 : -tom + peter   8000.0
 constarint4 : -tom + samir   8000.0
 constarint5 : -tom - peter + gary   0.0
 constarint7 : -2 tom - 2 peter + nina + samir   0.0
 constarint8 : -peter + bob   0.0
 constarint9 : -samir + bob   0.0
 constarint10 : peter + bob   75000.0
 constarint12 : -tom + IT   0.0
 constarint13 : -peter + IT   0.0
 constarint14 : -nina + IT   0.0
 constarint15 : -samir + IT   0.0
 constarint16 : -gary + Customer   0.0
 constarint17 : -bob + Customer   0.0
 constarint18 : -linda + Customer   0.0
 constarint11 : -tom - bob + linda   0.0
 tom   0.0
 peter   0.0
 nina   0.0
 samir   0.0
 gary   0.0
 bob   0.0
 linda   0.0
 IT   0.0
 Customer   0.0
```

[ ]:

```
optimize!(picframe1);
@show objective_value(picframe1);
@show value(Customer);
@show value(IT);
```

```
Presolving model
```

```
16 rows, 8 cols, 36 nonzeros
13 rows, 5 cols, 30 nonzeros
6 rows, 4 cols, 13 nonzeros
3 rows, 3 cols, 6 nonzeros
Presolve : Reductions: rows 3(-15); columns 3(-6); elements 6(-33)
Solving the presolved LP
Using EKK dual simplex solver - serial
  Iteration        Objective     Infeasibilities num(sum)
          0     1.0650020661e+05 Pr: 1(60000) 0s
          2     1.3650000000e+05 Pr: 0(0) 0s
Solving the original LP from the solution after postsolve
Model   status      : Optimal
Simplex   iterations: 2
Objective value     :  1.3650000000e+05
HiGHS run time      :          0.00
objective_value(picframe1) = 136500.0
value(Customer) = 68500.0
value(IT) = 68000.0
```