

# The OperationsResearch fall-2024 problem

## Summary

The goal of project is to maximize the revenue of a factory which is producing metal alloys and sends them to markets using containers. we have some limitations in different levels of problem which you can see in project doc.

## Pyomo formulation

We begin by importing the Pyomo package and creating a model abstract object:

```
from pyomo.environ import *
infinity = float('inf')
model = AbstractModel(name='OR1')
```

The sets *Ore*, *Alloys*, *Metals*, *Factories*, *Depots* and *Markets* are declared abstractly using the `Set` component:

```
model.Ore = Set()
model.Alloys = Set()
model.Metals = Set()
model.Factories = Set()
model.Depots = Set()
model.Markets = Set()
```

Similarly, we add parameters, the model parameters are defined abstractly using the `Param` component:

```
M = 999999999
epsilon = 1e-9
discount_percentage = 0.05

model.min_buy_fac = Param(model.Factories, within=NonNegativeReals,
    default=0.0)
model.max_buy_fac = Param(model.Factories, within=NonNegativeReals,
    default=infinity)
model.discount_margin = Param(model.Factories,
    within=NonNegativeReals, default=infinity)
model.contract_cost = Param(model.Factories, within=NonNegativeReals)
model.A_comb_min = Param(model.Metals, within=NonNegativeReals,
    default=0.0)
model.A_comb_max = Param(model.Metals, within=NonNegativeReals,
    default=infinity)
model.B_comb_min = Param(model.Metals, within=NonNegativeReals,
    default=0.0)
```

```

model.B_comb_max = Param(model.Metals, within=NonNegativeReals,
default=infinity)
model.price_of_alloy_fac = Param(model.Factories, model.Alloys,
within=NonNegativeReals)
model.Max_ore = Param(model.Ore,within=NonNegativeReals)
model.Ore_cost = Param(model.Ore,within=NonNegativeReals)
model.Ore_combination = Param(model.Ore, model.Metals,
within=NonNegativeReals)
model.container_cap = Param(within= NonNegativeIntegers)
model.Container_min_to_be_sent_depot = Param(model.Factories,
model.Depots, within=NonNegativeIntegers)
model.Container_Max_to_be_sent_depot = Param(model.Factories,
model.Depots, within=NonNegativeIntegers)
model.Container_cost_to_be_sent_depot = Param(model.Factories,
model.Depots , within=NonNegativeReals)
model.depots_min_to_receive = Param(model.Depots,
within=NonNegativeIntegers)
model.depots_Max_to_receive = Param(model.Depots,
within=NonNegativeIntegers)
model.Container_min_to_be_sent_market = Param(model.Depots,
model.Markets, within= NonNegativeIntegers)
model.Container_Max_to_be_sent_market = Param(model.Depots,
model.Markets, within= NonNegativeIntegers)
model.Container_cost_to_be_sent_market =
Param(model.Depots ,model.Markets, within= NonNegativeReals)
model.Max_market_demand = Param(model.Markets,model.Alloys, within=
NonNegativeReals)
model.Market_price = Param(model.Markets , model.Alloys , within=
NonNegativeReals)

```

The `within` option here is used in these parameter declarations to define expected properties of the parameters. This information is used to perform error checks on the data that is used to initialize the parameter components.

The `Var` component is used to define the decision variables: the `binary` is `{0,1}` to be clear.

```

model.Z = Var(model.Ore,model.Alloys, within=NonNegativeReals)
model.F = Var(model.Ore,model.Alloys, within=NonNegativeReals)
model.A = Var(model.Ore,model.Alloys, within=NonNegativeReals)
model.C = Var(model.Ore,model.Alloys, within=NonNegativeReals)
model.U = Var(model.Alloys,within=NonNegativeReals)
model.t = Var(model.Alloys,model.Factories,model.Depots,
within=NonNegativeReals)
model.Extracted_ore = Var(model.Ore,within=NonNegativeReals) # defined as S in report
model.h = Var(model.Factories,within= Binary)
model.B = Var(model.Factories, model.Depots,
within=NonNegativeIntegers)
model.g = Var(model.Alloys, model.Depots, model.Markets,

```

```

within=NonNegativeReals)
model.G = Var(model.Depots, model.Markets, within=
NonNegativeIntegers)
model.l = Var(model.Depots, model.Markets, within= Binary)
model.d = Var([1,2], within= Binary)
model.R = Var(model.Alloys,[1,2], within= NonNegativeReals,
initialize=0)

```

Rule functions are used to define constraint expressions in the `Constraint` component: here we have rule for maximum extraction of Ore:

```

def Max_extracted_ore_rule(model,i):
    return model.Extracted_ore[i] <= model.Max_ore[i]
model.Max_extracted_ore_limit =
Constraint(model.Ore,rule=Max_extracted_ore_rule)

```

Rule for Alloy weight limit(alloy weight is sum of metals weights in it):

```

def Alloy_sum_rule(model,j):
    return model.U[j] == sum(model.Z[i,j] for i in model.Ore)+\
sum(model.C[i,j] for i in model.Ore)+\
sum(model.A[i,j] for i in model.Ore)+\
sum(model.F[i,j] for i in model.Ore)
model.Alloy_sum_limit = Constraint(model.Alloys,rule=Alloy_sum_rule)

```

Rule for Metals in alloys(should be less than (or equal to) extracted metals from Ore):

```

def Metal_sum_rule_Z(model,i):
    return sum(model.Z[i,j] for j in model.Alloys) <=
model.Extracted_ore[i]*model.Ore_combination[i,'Zinc']
model.Metal_sum_limit_Z = Constraint(model.Ore,rule=Metal_sum_rule_Z)

def Metal_sum_rule_F(model,i):
    return sum(model.F[i,j] for j in model.Alloys) <=
model.Extracted_ore[i]*model.Ore_combination[i,'Iron']
model.Metal_sum_limit_F = Constraint(model.Ore,rule=Metal_sum_rule_F)

def Metal_sum_rule_C(model,i):
    return sum(model.C[i,j] for j in model.Alloys) <=
model.Extracted_ore[i]*model.Ore_combination[i,'Copper']
model.Metal_sum_limit_C = Constraint(model.Ore,rule=Metal_sum_rule_C)

def Metal_sum_rule_A(model,i):
    return sum(model.A[i,j] for j in model.Alloys) <=
model.Extracted_ore[i]*model.Ore_combination[i,'Aluminum']
model.Metal_sum_limit_A = Constraint(model.Ore,rule=Metal_sum_rule_A)

```

Rule for limitation of percentage of Metals in Alloys(f is bottom limit and t is top limit):

```

def Metal_in_alloy_rule_A_Z_f(model):
    value = sum(model.Z[i,'A'] for i in model.Ore)
    return model.A_comb_min['Zinc']*model.U['A']<=value
model.Metal_in_alloy_limit_A_Z_f =
Constraint(rule=Metal_in_alloy_rule_A_Z_f)
def Metal_in_alloy_rule_A_Z_t(model):
    value = sum(model.Z[i,'A'] for i in model.Ore)
    return value<=model.A_comb_max['Zinc']*model.U['A']
model.Metal_in_alloy_limit_A_Z_t =
Constraint(rule=Metal_in_alloy_rule_A_Z_t)

def Metal_in_alloy_rule_A_C_f(model):
    value = sum(model.C[i,'A'] for i in model.Ore)
    return model.A_comb_min['Copper']*model.U['A']<=value
model.Metal_in_alloy_limit_A_C_f =
Constraint(rule=Metal_in_alloy_rule_A_C_f)
def Metal_in_alloy_rule_A_C_t(model):
    value = sum(model.C[i,'A'] for i in model.Ore)
    return value<=model.A_comb_max['Copper']*model.U['A']
model.Metal_in_alloy_limit_A_C_t =
Constraint(rule=Metal_in_alloy_rule_A_C_t)

def Metal_in_alloy_rule_A_A_f(model):
    value = sum(model.A[i,'A'] for i in model.Ore)
    return model.A_comb_min['Aluminum']*model.U['A']<=value
model.Metal_in_alloy_limit_A_A_f =
Constraint(rule=Metal_in_alloy_rule_A_A_f)
def Metal_in_alloy_rule_A_A_t(model):
    value = sum(model.A[i,'A'] for i in model.Ore)
    return value<=model.A_comb_max['Aluminum']*model.U['A']
model.Metal_in_alloy_limit_A_A_t =
Constraint(rule=Metal_in_alloy_rule_A_A_t)

def Metal_in_alloy_rule_A_F_f(model):
    value = sum(model.F[i,'A'] for i in model.Ore)
    return model.A_comb_min['Iron']*model.U['A']<=value
model.Metal_in_alloy_limit_A_F_f =
Constraint(rule=Metal_in_alloy_rule_A_F_f)
def Metal_in_alloy_rule_A_F_t(model):
    value = sum(model.F[i,'A'] for i in model.Ore)
    return value<=model.A_comb_max['Iron']*model.U['A']
model.Metal_in_alloy_limit_A_F_t =
Constraint(rule=Metal_in_alloy_rule_A_F_t)

def Metal_in_alloy_rule_B_Z_f(model):
    value = sum(model.Z[i,'B'] for i in model.Ore)
    return model.B_comb_min['Zinc']*model.U['B']<=value
model.Metal_in_alloy_limit_B_Z_f =
Constraint(rule=Metal_in_alloy_rule_B_Z_f)
def Metal_in_alloy_rule_B_Z_t(model):

```

```

        value = sum(model.Z[i,'B'] for i in model.Ore)
        return value<=model.B_comb_max['Zinc']*model.U['B']
model.Metal_in_alloy_limit_B_Z_t =
Constraint(rule=Metal_in_alloy_rule_B_Z_t)

def Metal_in_alloy_rule_B_C_f(model):
    value = sum(model.C[i,'B'] for i in model.Ore)
    return model.B_comb_min['Copper']*model.U['B']<=value
model.Metal_in_alloy_limit_B_C_f =
Constraint(rule=Metal_in_alloy_rule_B_C_f)
def Metal_in_alloy_rule_B_C_t(model):
    value = sum(model.C[i,'B'] for i in model.Ore)
    return value<=model.B_comb_max['Copper']*model.U['B']
model.Metal_in_alloy_limit_B_C_t =
Constraint(rule=Metal_in_alloy_rule_B_C_t)

def Metal_in_alloy_rule_B_A_f(model):
    value = sum(model.A[i,'B'] for i in model.Ore)
    return model.B_comb_min['Aluminum']*model.U['B']<=value
model.Metal_in_alloy_limit_B_A_f =
Constraint(rule=Metal_in_alloy_rule_B_A_f)
def Metal_in_alloy_rule_B_A_t(model):
    value = sum(model.A[i,'B'] for i in model.Ore)
    return value<=model.B_comb_max['Aluminum']*model.U['B']
model.Metal_in_alloy_limit_B_A_t =
Constraint(rule=Metal_in_alloy_rule_B_A_t)

def Metal_in_alloy_rule_B_F_f(model):
    value = sum(model.F[i,'B'] for i in model.Ore)
    return model.B_comb_min['Iron']*model.U['B']<=value
model.Metal_in_alloy_limit_B_F_f =
Constraint(rule=Metal_in_alloy_rule_B_F_f)
def Metal_in_alloy_rule_B_F_t(model):
    value = sum(model.F[i,'B'] for i in model.Ore)
    return value<=model.B_comb_max['Iron']*model.U['B']
model.Metal_in_alloy_limit_B_F_t =
Constraint(rule=Metal_in_alloy_rule_B_F_t)

```

Rule for amount of exported alloy from main Factory, it should be less than(or equal to):

```

def Export_from_main_fac_rule(model,i):
    return model.U[i] >= sum(model.t[i,'Main',k] for k in
model.Depots)
model.Export_from_main_fac_limit =
Constraint(model.Alloys,rule=Export_from_main_fac_rule)

```

Rule of Limits of buying from factories:

```

def buy_from_fac_rule_f(model,i):
    value = sum(sum(model.t[j,i,k] for k in model.Depots)\
                  for j in model.Alloys)
    return model.min_buy_fac[i]*model.h[i]<=value
model.buy_from_fac_limit_f= Constraint([1,2],rule=buy_from_fac_rule_f)
def buy_from_fac_rule_t(model,i):
    value = sum(sum(model.t[j,i,k] for k in model.Depots)\
                  for j in model.Alloys)
    return value<=model.max_buy_fac[i]*model.h[i]
model.buy_from_fac_limit_t= Constraint([1,2],rule=buy_from_fac_rule_t)

```

Rule of limit for Alloys in one container from Factory to Depot:

```

def container_rule(model,i,j):
    return sum(model.t[a,i,j] for a in model.Alloys) <=
model.B[i,j]*model.container_cap
model.container_limit = Constraint(model.Factories, model.Depots,
rule=container_rule)

```

Rule of limit for transporting from fac to depots No1.:

```

def transportation_rule_t(model,i,j):
    return model.B[i,j]<=
model.Container_Max_to_be_sent_depot[i,j]*model.h[i]
model.transportation_limit_t =
Constraint(model.Factories,model.Depots, rule= transportation_rule_t)
def transportation_rule_f(model,i,j):
    return
model.Container_min_to_be_sent_depot[i,j]*model.h[i]<=model.B[i,j]
model.transportation_limit_f =
Constraint(model.Factories,model.Depots, rule= transportation_rule_f)

```

Rule of limit for transporting from fac to depots No2.:

```

def transportation_rule2(model,j):
    return inequality(model.depots_min_to_receive[j],sum(model.B[i,j]
for i in model.Factories),\
                    model.depots_Max_to_receive[j])
model.transportation_limit2 = Constraint(model.Depots,rule=
transportation_rule2)

```

Rule of limit for transporting from depots to markets:

```

def transp_from_dep_to_marker_rule(model,i,k):
    return sum(model.t[i,j,k] for j in model.Factories) >=
sum(model.g[i,k,l] for l in model.Markets)
model.transp_from_dep_to_marker_limit =
Constraint(model.Alloys,model.Depots,\

```

```

transp_from_dep_to_marker_rule)
rule=

```

Rule of limits for Alloys in containers transporting from depots to markets:

```

def container_rule2(model,i,j):
    return sum(model.g[l,i,j] for l in model.Alloys) <=
model.G[i,j]*model.container_cap
model.container_limit2 = Constraint(model.Depots, model.Markets,
rule=container_rule2)

```

Limit for containers to be sent to markets:

```

def market_sell_rule_f(model,i,j):
    return
model.Container_min_to_be_sent_market[i,j]*model.l[i,j]<=model.G[i,j]
model.market_sell_limit_f = Constraint(model.Depots,model.Markets,
rule= market_sell_rule_f)
def market_sell_rule_t(model,i,j):
    return
model.G[i,j]<=model.Container_min_to_be_sent_market[i,j]*model.l[i,j]
model.market_sell_limit_t = Constraint(model.Depots,model.Markets,
rule= market_sell_rule_t)

```

Here we have maximum market demands rule:

```

def max_market_demand_rule(model,k,i):
    return sum(model.g[i,j,k] for j in model.Depots) <=
model.Max_market_demand[k,i]
model.max_market_demand_limit = Constraint(model.Markets,
model.Alloys, rule= max_market_demand_rule)

```

The **Objective** component is used to define the revenue objective. This component uses a rule function to construct the objective expression:

*sense=maximize* means we want to maximize the revenue.

```

def revenue_rule(model):
    return sum(sum(model.Market_price[m,j]*sum(model.g[j,k,m] for k in
model.Depots) for j in model.Alloys) for m in model.Markets)-\
sum(model.Extracted_ore[i]*model.Ore_cost[i] for i in
model.Ore)-\
sum(sum(model.price_of_alloy_fac[u,j]*sum(model.t[j,u,k]
for k in model.Depots) for j in model.Alloys) for u in
model.Factories)-\
sum(model.h[u]*model.contract_cost[u] for u in
model.Factories)-\

```

```

sum(sum(model.Container_cost_to_be_sent_depot[i,j]*model.B[i,j] for j
in model.Depots) for i in model.Factories)-\

sum(sum(model.G[i,j]*model.Container_cost_to_be_sent_market[i,j] for j
in model.Markets) for i in model.Depots)

model.revenue = Objective(rule=revenue_rule, sense=maximize)

```

here is added constraints and adjusted revenue for part B:

```

def discount_rule_1(model,u):
    return sum(sum(model.t[j,u,k] for j in model.Alloys) for k in
model.Depots)+epsilon <= model.d[u]*model.discount_margin[u] +\
model.discount_margin[u]

def discount_rule_2(model,u):
    return sum(sum(model.t[j,u,k] for j in model.Alloys) for k in
model.Depots) >= model.discount_margin[u]*model.d[u]

def discount_rule_3(model,u,j):
    return sum(model.t[j,u,k] for k in model.Depots) >= model.R[j,u]

def discount_rule_4(model,u):
    return sum(model.R[j,u] for j in model.Alloys) <= model.d[u]*M

def revenue_rule_discount_added(model):
    return sum(sum(model.Market_price[m,j]*sum(model.g[j,k,m] for k in
model.Depots) for j in model.Alloys) for m in model.Markets)-\
sum(model.Extracted_ore[i]*model.Ore_cost[i] for i in
model.Ore)-\
sum(sum(model.price_of_alloy_fac[u,j]*sum(model.t[j,u,k]
for k in model.Depots) for j in model.Alloys) for u in
model.Factories)-\
sum(model.h[u]*model.contract_cost[u] for u in
model.Factories)-\

sum(sum(model.Container_cost_to_be_sent_depot[i,j]*model.B[i,j] for j
in model.Depots) for i in model.Factories)-\

sum(sum(model.G[i,j]*model.Container_cost_to_be_sent_market[i,j] for j
in model.Markets) for i in model.Depots)+\

sum(sum(discount_percentage*model.R[j,u]*model.price_of_alloy_fac[u,j]
for u in [1,2]) for j in model.Alloys)

def apply_discount_rule():
    model.discount_limit_1 = Constraint([1,2],rule=discount_rule_1)
    model.discount_limit_2 = Constraint([1,2],rule=discount_rule_2)
    model.discount_limit_3 =

```



```

Constraint([1,2],model.Alloys,rule=discount_rule_3)
    model.discount_limit_4 = Constraint([1,2],rule=discount_rule_4)
    model.revenue = Objective(rule=revenue_rule_discount_added,
sense=maximize)

```

## model data

since we have made an abstract model, we can add the data after creating model, to see the data we are feeding the model

execute command below or have a look at params.dat:

```

!cat params.dat

set Alloys:=
    A
    B;

param: Factories:          min_buy_fac  discount_margin  max_buy_fac
contract_cost:=
    1          2000          2500          5000
120
    2          2500          3000          6000
90
    Main      0          .          .
0 ;

param: Depots:          depots_min_to_receive  depots_Max_to_receive :=
    Tehran      20          65
    Isfahan     30          70 ;

param container_cap:= 100;

set Markets:=
    Mashhad
    Kerman
    Ahvaz
    Tabriz ;

param: Metals:          A_comb_min  A_comb_max  B_comb_min
B_comb_max :=
    Iron      0          0.25          0.45          0.70
    Aluminum  0.55          1          0          0.70
    Zinc      0          0.8          0          1
    Copper    0          1          0.35          1 ;

param: Ore:          Max_ore  Ore_cost :=
    1          560          45
    2          1000          65
    3          1440          70 ;

```

param Ore\_combination:

	Iron	Aluminum	Zinc	Copper :=
1	.05	.35	.25	.30
2	.20	.30	.15	.25
3	.05	.25	.65	.05 ;

param Container\_min\_to\_be\_sent\_depot:

	Tehran	Isfahan :=
Main	5	5
1	10	10
2	5	5 ;

param Container\_Max\_to\_be\_sent\_depot:

	Tehran	Isfahan :=
Main	20	20
1	30	30
2	25	15 ;

param Container\_cost\_to\_be\_sent\_depot:

	Tehran	Isfahan :=
Main	200	230
1	180	210
2	240	220 ;

param Container\_min\_to\_be\_sent\_market:

	Mashhad	Kerman	Ahvaz	Tabriz :=
Tehran	3	6	10	5
Isfahan	4	5	5	10 ;

param Container\_Max\_to\_be\_sent\_market:

	Mashhad	Kerman	Ahvaz	Tabriz :=
Tehran	7	12	18	15
Isfahan	6	14	20	20 ;

param Container\_cost\_to\_be\_sent\_market:

	Mashhad	Kerman	Ahvaz	Tabriz :=
Tehran	110	85	120	100
Isfahan	100	100	110	90 ;

param Max\_market\_demand:

	A	B :=
Mashhad	600	400
Kerman	800	1200
Ahvaz	1500	1500

```

    Tabriz      1400    1100    ;
param Market_price:
    A          B :=
    Mashhad    520    700
    Kerman     540    690
    Ahvaz      490    730
    Tabriz     500    710    ;

param price_of_alloy_fac:
    A          B :=
    1          375    520
    2          390    540
    Main       0      0      ;

```

## Solution

To get the result of a problem, you can execute the command below:

make sure to replace `-problem-number` with correct number, use `problem-dict`.

results are also saved in `results.yaml`

```
!python model_runner.py -problem-number
```

or use code below in cases of having plots to be shown:

```
%matplotlib inline
%run model_runner.py -problem-number
```

this is problem-dictionary

```
{ '-a': 'الف', '-b': 'ب', '-c': 'ج', '-d': 'د', '-e': 'ه',
  '-f': 'و', '-g': 'ز', '-h': 'ح', '-i': 'ط' }
```

for example you can execute the code below to see results of problem `الف`:

## Problem A

```
!python model_runner.py -a
```

results for problem: `-a`

Problem:

```
- Name: unknown
  Lower bound: 1037982.72727273
  Upper bound: 1037982.72727273
  Number of objectives: 1
```

Number of constraints: 97  
Number of variables: 82  
Number of nonzeros: 304  
Sense: maximize

Solver:

- Status: ok  
Termination condition: optimal  
Statistics:

Branch and bound:

Number of bounded subproblems: 33  
Number of created subproblems: 33

Error rc: 0

Time: 0.010471343994140625

Solution:

- number of solutions: 0  
number of solutions displayed: 0

Model OR1

Variables:

Z : Size=6, Index=0re\*Alloys

Key	: Lower	: Value	: Upper	: Fixed	: Stale	:
-----	---------	---------	---------	---------	---------	---

(1, 'A')	: 0	: 140.0	: None	: False	: False	:
----------	-----	---------	--------	---------	---------	---

NonNegativeReals

(1, 'B')	: 0	: 0.0	: None	: False	: False	:
----------	-----	-------	--------	---------	---------	---

NonNegativeReals

(2, 'A')	: 0	: 150.0	: None	: False	: False	:
----------	-----	---------	--------	---------	---------	---

NonNegativeReals

(2, 'B')	: 0	: 0.0	: None	: False	: False	:
----------	-----	-------	--------	---------	---------	---

NonNegativeReals

(3, 'A')	: 0	: 153.69696969697	: None	: False	: False	:
----------	-----	-------------------	--------	---------	---------	---

NonNegativeReals

(3, 'B')	: 0	: 133.333333333333	: None	: False	: False	:
----------	-----	--------------------	--------	---------	---------	---

NonNegativeReals

F : Size=6, Index=0re\*Alloys

Key	: Lower	: Value	: Upper	: Fixed	: Stale	: Domain
-----	---------	---------	---------	---------	---------	----------

NonNegativeReals

(1, 'A')	: 0	: 0.0	: None	: False	: False	:
----------	-----	-------	--------	---------	---------	---

NonNegativeReals

(1, 'B')	: 0	: 28.0	: None	: False	: False	:
----------	-----	--------	--------	---------	---------	---

NonNegativeReals

(2, 'A')	: 0	: 0.0	: None	: False	: False	:
----------	-----	-------	--------	---------	---------	---

NonNegativeReals

(2, 'B')	: 0	: 200.0	: None	: False	: False	:
----------	-----	---------	--------	---------	---------	---

NonNegativeReals

(3, 'A')	: 0	: 0.0	: None	: False	: False	:
----------	-----	-------	--------	---------	---------	---

NonNegativeReals

(3, 'B')	: 0	: 72.0	: None	: False	: False	:
----------	-----	--------	--------	---------	---------	---

NonNegativeReals

```

A : Size=6, Index=0re*Alloys
  Key      : Lower : Value : Upper : Fixed : Stale : Domain
  (1, 'A') :      0 : 196.0 : None : False : False :
NonNegativeReals
  (1, 'B') :      0 :   0.0 : None : False : False :
NonNegativeReals
  (2, 'A') :      0 : 300.0 : None : False : False :
NonNegativeReals
  (2, 'B') :      0 :   0.0 : None : False : False :
NonNegativeReals
  (3, 'A') :      0 : 360.0 : None : False : False :
NonNegativeReals
  (3, 'B') :      0 :   0.0 : None : False : False :
NonNegativeReals
C : Size=6, Index=0re*Alloys
  Key      : Lower : Value : Upper : Fixed : Stale :
Domain
  (1, 'A') :      0 :           0.0 : None : False : False :
NonNegativeReals
  (1, 'B') :      0 :          168.0 : None : False : False :
NonNegativeReals
  (2, 'A') :      0 : 184.6666666666667 : None : False : False :
NonNegativeReals
  (2, 'B') :      0 : 65.33333333333333 : None : False : False :
NonNegativeReals
  (3, 'A') :      0 :           72.0 : None : False : False :
NonNegativeReals
  (3, 'B') :      0 :           0.0 : None : False : False :
NonNegativeReals
U : Size=2, Index=Alloys
  Key      : Lower : Value : Upper : Fixed : Stale :
Domain
  A :      0 : 1556.36363636364 : None : False : False :
NonNegativeReals
  B :      0 : 666.6666666666667 : None : False : False :
NonNegativeReals
t : Size=12, Index=Alloys*Factories*Depots
  Key      : Lower : Value : Upper :
Fixed : Stale : Domain
  ('A', 1, 'Isfahan') :      0 :           0.0 : None :
False : False : NonNegativeReals
  ('A', 1, 'Tehran') :      0 :           0.0 : None :
False : False : NonNegativeReals
  ('A', 2, 'Isfahan') :      0 :           0.0 : None :
False : False : NonNegativeReals
  ('A', 2, 'Tehran') :      0 :           0.0 : None :
False : False : NonNegativeReals
  ('A', 'Main', 'Isfahan') :      0 : 1256.36363636364 : None :
False : False : NonNegativeReals

```

```

('A', 'Main', 'Tehran') :      0 :      300.0 : None :
False : False : NonNegativeReals
('B', 1, 'Isfahan') :      0 : 676.969696969697 : None :
False : False : NonNegativeReals
('B', 1, 'Tehran') :      0 :      1900.0 : None :
False : False : NonNegativeReals
('B', 2, 'Isfahan') :      0 :          0.0 : None :
False : False : NonNegativeReals
('B', 2, 'Tehran') :      0 :          0.0 : None :
False : False : NonNegativeReals
('B', 'Main', 'Isfahan') :      0 : 466.666666666667 : None :
False : False : NonNegativeReals
('B', 'Main', 'Tehran') :      0 :      200.0 : None :
False : False : NonNegativeReals
  Extracted_ore : Size=3, Index=0re
    Key : Lower : Value : Upper : Fixed : Stale : Domain
      1 :      0 : 560.0 : None : False : False :
NonNegativeReals
      2 :      0 : 1000.0 : None : False : False :
NonNegativeReals
      3 :      0 : 1440.0 : None : False : False :
NonNegativeReals
    h : Size=3, Index=Factories
      Key : Lower : Value : Upper : Fixed : Stale : Domain
        1 :      0 :  1.0 :      1 : False : False : Binary
        2 :      0 :  0.0 :      1 : False : False : Binary
      Main :      0 :  1.0 :      1 : False : False : Binary
    B : Size=6, Index=Factories*Depots
      Key : Lower : Value : Upper : Fixed : Stale :
Domain
      (1, 'Isfahan') :      0 :  12.0 : None : False : False :
NonNegativeIntegers
      (1, 'Tehran') :      0 :  19.0 : None : False : False :
NonNegativeIntegers
      (2, 'Isfahan') :      0 :   0.0 : None : False : False :
NonNegativeIntegers
      (2, 'Tehran') :      0 :   0.0 : None : False : False :
NonNegativeIntegers
      ('Main', 'Isfahan') :      0 :  18.0 : None : False : False :
NonNegativeIntegers
      ('Main', 'Tehran') :      0 :   5.0 : None : False : False :
NonNegativeIntegers
    g : Size=16, Index=Alloys*Depots*Markets
      Key : Lower : Value : Upper :
: Fixed : Stale : Domain
      ('A', 'Isfahan', 'Ahvaz') :      0 :          0.0 : None
: False : False : NonNegativeReals
      ('A', 'Isfahan', 'Kerman') :      0 :      500.0 : None
: False : False : NonNegativeReals

```

```

      ('A', 'Isfahan', 'Mashhad') :      0 : 356.363636363636 : None
: False : False : NonNegativeReals
      ('A', 'Isfahan', 'Tabriz') :      0 :           400.0 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Ahvaz') :      0 :           0.0 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Kerman') :      0 :           300.0 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Mashhad') :      0 :           0.0 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Tabriz') :      0 :           0.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Ahvaz') :      0 :           500.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Kerman') :      0 :           0.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Mashhad') :      0 : 43.6363636363637 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Tabriz') :      0 :           600.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Ahvaz') :      0 :          1000.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Kerman') :      0 :           300.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Mashhad') :      0 :           300.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Tabriz') :      0 :           500.0 : None
: False : False : NonNegativeReals
      G : Size=8, Index=Depots*Markets
          Key : Lower : Value : Upper : Fixed : Stale
: Domain
      ('Isfahan', 'Ahvaz') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Kerman') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Mashhad') :      0 :    4.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Tabriz') :      0 :   10.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Ahvaz') :      0 :   10.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Kerman') :      0 :    6.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Mashhad') :      0 :    3.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Tabriz') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      l : Size=8, Index=Depots*Markets
          Key : Lower : Value : Upper : Fixed : Stale

```

```

: Domain
    ('Isfahan', 'Ahvaz') :      0 :    1.0 :      1 : False : False
: Binary
    ('Isfahan', 'Kerman') :      0 :    1.0 :      1 : False : False
: Binary
    ('Isfahan', 'Mashhad') :      0 :    1.0 :      1 : False : False
: Binary
    ('Isfahan', 'Tabriz') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Ahvaz') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Kerman') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Mashhad') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Tabriz') :      0 :    1.0 :      1 : False : False
: Binary
    d : Size=2, Index={1, 2}
        Key : Lower : Value : Upper : Fixed : Stale : Domain
        1 :      0 : None :      1 : False : True : Binary
        2 :      0 : None :      1 : False : True : Binary
    R : Size=4, Index=Alloys*{1, 2}
        Key      : Lower : Value : Upper : Fixed : Stale : Domain
        ('A', 1) :      0 :      0 : None : False : True :
NonNegativeReals
        ('A', 2) :      0 :      0 : None : False : True :
NonNegativeReals
        ('B', 1) :      0 :      0 : None : False : True :
NonNegativeReals
        ('B', 2) :      0 :      0 : None : False : True :
NonNegativeReals

Objectives:
    revenue : Size=1, Index=None, Active=True
        Key : Active : Value
        None : True : 1037982.7272727261

Constraints:
    Max_extracted_ore_limit : Size=3
        Key : Lower : Body      : Upper
        1 : None : 560.0 : 560
        2 : None : 1000.0 : 1000
        3 : None : 1440.0 : 1440
    Alloy_sum_limit : Size=2
        Key : Lower : Body : Upper
        A : None : 2.9558577807620168e-12 : 0.0
        B : None : 6.821210263296962e-13 : 0.0
    Metal_sum_limit_Z : Size=3
        Key : Lower : Body : Upper

```



```

1 : None : 0.0 : 0.0
2 : None : 0.0 : 0.0
3 : None : -648.969696969697 : 0.0
Metal_sum_limit_F : Size=3
Key : Lower : Body : Upper
1 : None : 0.0 : 0.0
2 : None : 0.0 : 0.0
3 : None : 0.0 : 0.0
Metal_sum_limit_C : Size=3
Key : Lower : Body : Upper
1 : None : 0.0 : 0.0
2 : None : 2.984279490192421e-13 : 0.0
3 : None : 0.0 : 0.0
Metal_sum_limit_A : Size=3
Key : Lower : Body : Upper
1 : None : 0.0 : 0.0
2 : None : 0.0 : 0.0
3 : None : 0.0 : 0.0
Metal_in_alloy_limit_A_Z_f : Size=1
Key : Lower : Body : Upper
None : None : -443.69696969697 : 0.0
Metal_in_alloy_limit_A_Z_t : Size=1
Key : Lower : Body : Upper
None : None : -801.393939393942 : 0.0
Metal_in_alloy_limit_A_C_f : Size=1
Key : Lower : Body : Upper
None : None : -256.66666666666697 : 0.0
Metal_in_alloy_limit_A_C_t : Size=1
Key : Lower : Body : Upper
None : None : -1299.696969696973 : 0.0
Metal_in_alloy_limit_A_A_f : Size=1
Key : Lower : Body : Upper
None : None : 2.0463630789890885e-12 : 0.0
Metal_in_alloy_limit_A_A_t : Size=1
Key : Lower : Body : Upper
None : None : -700.3636363636399 : 0.0
Metal_in_alloy_limit_A_F_f : Size=1
Key : Lower : Body : Upper
None : None : 0.0 : 0.0
Metal_in_alloy_limit_A_F_t : Size=1
Key : Lower : Body : Upper
None : None : -389.09090909091 : 0.0
Metal_in_alloy_limit_B_Z_f : Size=1
Key : Lower : Body : Upper
None : None : -133.3333333333333 : 0.0
Metal_in_alloy_limit_B_Z_t : Size=1
Key : Lower : Body : Upper
None : None : -533.3333333333339 : 0.0
Metal_in_alloy_limit_B_C_f : Size=1

```

```

Key : Lower : Body : Upper
None : None : 1.1368683772161603e-13 : 0.0
Metal_in_alloy_limit_B_C_t : Size=1
Key : Lower : Body : Upper
None : None : -433.3333333333336 : 0.0
Metal_in_alloy_limit_B_A_f : Size=1
Key : Lower : Body : Upper
None : None : 0.0 : 0.0
Metal_in_alloy_limit_B_A_t : Size=1
Key : Lower : Body : Upper
None : None : -466.66666666666686 : 0.0
Metal_in_alloy_limit_B_F_f : Size=1
Key : Lower : Body : Upper
None : None : 1.7053025658242404e-13 : 0.0
Metal_in_alloy_limit_B_F_t : Size=1
Key : Lower : Body : Upper
None : None : -166.66666666666686 : 0.0
Export_from_main_fac_limit : Size=2
Key : Lower : Body : Upper
A : None : 0.0 : 0.0
B : None : 5.684341886080802e-14 : 0.0
buy_from_fac_limit_f : Size=2
Key : Lower : Body : Upper
1 : None : -576.969696969697 : 0.0
2 : None : 0.0 : 0.0
buy_from_fac_limit_t : Size=2
Key : Lower : Body : Upper
1 : None : -2423.030303030303 : 0.0
2 : None : 0.0 : 0.0
container_limit : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -523.030303030303 : 0.0
(1, 'Tehran') : None : 0.0 : 0.0
(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : -76.96969696969308 : 0.0
('Main', 'Tehran') : None : 0.0 : 0.0
transportation_limit_t : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -18.0 : 0.0
(1, 'Tehran') : None : -11.0 : 0.0
(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : -2.0 : 0.0
('Main', 'Tehran') : None : -15.0 : 0.0
transportation_limit_f : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -2.0 : 0.0
(1, 'Tehran') : None : -9.0 : 0.0

```

```

(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : -13.0 : 0.0
('Main', 'Tehran') : None : 0.0 : 0.0
transportation_limit2 : Size=2
Key : Lower : Body : Upper
Isfahan : 30 : 30.0 : 70.0
Tehran : 20 : 24.0 : 65.0
transp_from_dep_to_market_limit : Size=4
Key : Lower : Body : Upper
('A', 'Isfahan') : None : -3.865352482534945e-12 : 0.0
('A', 'Tehran') : None : 0.0 : 0.0
('B', 'Isfahan') : None : -2.2737367544323206e-13 : 0.0
('B', 'Tehran') : None : 0.0 : 0.0
container_limit2 : Size=8
Key : Lower : Body : Upper
Upper
('Isfahan', 'Ahvaz') : None : 0.0 :
0.0
('Isfahan', 'Kerman') : None : 0.0 :
0.0
('Isfahan', 'Mashhad') : None : -3.268496584496461e-13 :
0.0
('Isfahan', 'Tabriz') : None : 0.0 :
0.0
('Tehran', 'Ahvaz') : None : 0.0 :
0.0
('Tehran', 'Kerman') : None : 0.0 :
0.0
('Tehran', 'Mashhad') : None : 0.0 :
0.0
('Tehran', 'Tabriz') : None : 0.0 :
0.0
market_sell_limit_f : Size=8
Key : Lower : Body : Upper
('Isfahan', 'Ahvaz') : None : 0.0 : 0.0
('Isfahan', 'Kerman') : None : 0.0 : 0.0
('Isfahan', 'Mashhad') : None : 0.0 : 0.0
('Isfahan', 'Tabriz') : None : 0.0 : 0.0
('Tehran', 'Ahvaz') : None : 0.0 : 0.0
('Tehran', 'Kerman') : None : 0.0 : 0.0
('Tehran', 'Mashhad') : None : 0.0 : 0.0
('Tehran', 'Tabriz') : None : 0.0 : 0.0
market_sell_limit_t : Size=8
Key : Lower : Body : Upper
('Isfahan', 'Ahvaz') : None : 0.0 : 0.0
('Isfahan', 'Kerman') : None : 0.0 : 0.0
('Isfahan', 'Mashhad') : None : 0.0 : 0.0
('Isfahan', 'Tabriz') : None : 0.0 : 0.0

```

```

('Tehran', 'Ahvaz') : None : 0.0 : 0.0
('Tehran', 'Kerman') : None : 0.0 : 0.0
('Tehran', 'Mashhad') : None : 0.0 : 0.0
('Tehran', 'Tabriz') : None : 0.0 : 0.0
max_market_demand_limit : Size=8
Key : Lower : Body : Upper
('Ahvaz', 'A') : None : 0.0 : 1500.0
('Ahvaz', 'B') : None : 1500.0 : 1500.0
('Kerman', 'A') : None : 800.0 : 800.0
('Kerman', 'B') : None : 300.0 : 1200.0
('Mashhad', 'A') : None : 356.363636363636 : 600.0
('Mashhad', 'B') : None : 343.6363636363637 : 400.0
('Tabriz', 'A') : None : 400.0 : 1400.0
('Tabriz', 'B') : None : 1100.0 : 1100.0
results saved in results.yaml

```

## Explanation

In the report above, we have three parts, **variables**, **objective** and **constraints**, you can see value of variables in **Value** column of variables section, and as you see, value of objective(revenue) is equal to **1037982.72727273**

## Problem B

```

!python model_runner.py -b

results for problem: -b
WARNING: Implicitly replacing the Component attribute revenue
(type=<class
'pyomo.core.base.objective.ScalarObjective'>) on block OR1 with a new
Component (type=<class 'pyomo.core.base.objective.ScalarObjective'>).
This is
usually indicative of a modelling error. To avoid this warning, use
block.del_component() and block.add_component().

Problem:
- Name: unknown
  Lower bound: 1107067.57575758
  Upper bound: 1107067.57575758
  Number of objectives: 1
  Number of constraints: 107
  Number of variables: 88
  Number of nonzeros: 342
  Sense: maximize
Solver:
- Status: ok
  Termination condition: optimal
  Statistics:
    Branch and bound:
      Number of bounded subproblems: 29

```

Number of created subproblems: 29

Error rc: 0

Time: 0.010447025299072266

Solution:

- number of solutions: 0

number of solutions displayed: 0

Model OR1

Variables:

Z : Size=6, Index=Ore\*Alloys

Key	: Lower	: Value	: Upper	: Fixed	: Stale	:
Domain						
(1, 'A')	: 0	: 38.66666666666667	: None	: False	: False	:
NonNegativeReals						
(1, 'B')	: 0	: 101.3333333333333	: None	: False	: False	:
NonNegativeReals						
(2, 'A')	: 0	: 150.0	: None	: False	: False	:
NonNegativeReals						
(2, 'B')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(3, 'A')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(3, 'B')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						

F : Size=6, Index=Ore\*Alloys

Key	: Lower	: Value	: Upper	: Fixed	: Stale	: Domain
(1, 'A')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(1, 'B')	: 0	: 28.0	: None	: False	: False	:
NonNegativeReals						
(2, 'A')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(2, 'B')	: 0	: 200.0	: None	: False	: False	:
NonNegativeReals						
(3, 'A')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(3, 'B')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						

A : Size=6, Index=Ore\*Alloys

Key	: Lower	: Value	: Upper	: Fixed	: Stale	: Domain
(1, 'A')	: 0	: 196.0	: None	: False	: False	:
NonNegativeReals						
(1, 'B')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						
(2, 'A')	: 0	: 300.0	: None	: False	: False	:
NonNegativeReals						
(2, 'B')	: 0	: 0.0	: None	: False	: False	:
NonNegativeReals						

```

(3, 'A') :      0 :    0.0 :  None : False : False :
NonNegativeReals
(3, 'B') :      0 :    0.0 :  None : False : False :
NonNegativeReals
C : Size=6, Index=Ore*Alloys
Key      : Lower : Value      : Upper : Fixed : Stale :
Domain
(1, 'A') :      0 : 144.48484848484848 :  None : False : False :
NonNegativeReals
(1, 'B') :      0 :          0.0 :  None : False : False :
NonNegativeReals
(2, 'A') :      0 : 72.66666666666667 :  None : False : False :
NonNegativeReals
(2, 'B') :      0 : 177.33333333333333 :  None : False : False :
NonNegativeReals
(3, 'A') :      0 :          0.0 :  None : False : False :
NonNegativeReals
(3, 'B') :      0 :          0.0 :  None : False : False :
NonNegativeReals
U : Size=2, Index=Alloys
Key : Lower : Value      : Upper : Fixed : Stale :
Domain
A :      0 : 901.81818181818182 :  None : False : False :
NonNegativeReals
B :      0 : 506.66666666666667 :  None : False : False :
NonNegativeReals
t : Size=12, Index=Alloys*Factories*Depots
Key      : Lower : Value      : Upper :
Fixed : Stale : Domain
('A', 1, 'Isfahan') :      0 :          0.0 :  None :
False : False : NonNegativeReals
('A', 1, 'Tehran') :      0 :          0.0 :  None :
False : False : NonNegativeReals
('A', 2, 'Isfahan') :      0 :          0.0 :  None :
False : False : NonNegativeReals
('A', 2, 'Tehran') :      0 :          0.0 :  None :
False : False : NonNegativeReals
('A', 'Main', 'Isfahan') :      0 : 408.48484848484848 :  None :
False : False : NonNegativeReals
('A', 'Main', 'Tehran') :      0 : 493.33333333333333 :  None :
False : False : NonNegativeReals
('B', 1, 'Isfahan') :      0 : 1991.5151515151515 :  None :
False : False : NonNegativeReals
('B', 1, 'Tehran') :      0 :        1400.0 :  None :
False : False : NonNegativeReals
('B', 2, 'Isfahan') :      0 :          0.0 :  None :
False : False : NonNegativeReals
('B', 2, 'Tehran') :      0 :          0.0 :  None :
False : False : NonNegativeReals

```

```

      ('B', 'Main', 'Isfahan') :      0 :      0.0 : None :
False : False : NonNegativeReals
      ('B', 'Main', 'Tehran') :      0 : 506.6666666666667 : None :
False : False : NonNegativeReals
      Extracted_ore : Size=3, Index=0re
      Key : Lower : Value : Upper : Fixed : Stale : Domain
      1 :      0 : 560.0 : None : False : False :
NonNegativeReals
      2 :      0 : 1000.0 : None : False : False :
NonNegativeReals
      3 :      0 :      0.0 : None : False : False :
NonNegativeReals
      h : Size=3, Index=Factories
      Key : Lower : Value : Upper : Fixed : Stale : Domain
      1 :      0 :      1.0 :      1 : False : False : Binary
      2 :      0 :      0.0 :      1 : False : False : Binary
      Main :      0 :      1.0 :      1 : False : False : Binary
      B : Size=6, Index=Factories*Depots
      Key : Lower : Value : Upper : Fixed : Stale :
Domain
      (1, 'Isfahan') :      0 : 25.0 : None : False : False :
NonNegativeIntegers
      (1, 'Tehran') :      0 : 14.0 : None : False : False :
NonNegativeIntegers
      (2, 'Isfahan') :      0 :      0.0 : None : False : False :
NonNegativeIntegers
      (2, 'Tehran') :      0 :      0.0 : None : False : False :
NonNegativeIntegers
      ('Main', 'Isfahan') :      0 :      5.0 : None : False : False :
NonNegativeIntegers
      ('Main', 'Tehran') :      0 :     10.0 : None : False : False :
NonNegativeIntegers
      g : Size=16, Index=Alloys*Depots*Markets
      Key : Lower : Value : Upper :
: Fixed : Stale : Domain
      ('A', 'Isfahan', 'Ahvaz') :      0 :      0.0 : None
: False : False : NonNegativeReals
      ('A', 'Isfahan', 'Kerman') :      0 :      0.0 : None
: False : False : NonNegativeReals
      ('A', 'Isfahan', 'Mashhad') :      0 :     300.0 : None
: False : False : NonNegativeReals
      ('A', 'Isfahan', 'Tabriz') :      0 : 108.48484848484848 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Ahvaz') :      0 :      0.0 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Kerman') :      0 : 201.81818181818182 : None
: False : False : NonNegativeReals
      ('A', 'Tehran', 'Mashhad') :      0 :      0.0 : None
: False : False : NonNegativeReals

```

```

      ('A', 'Tehran', 'Tabriz') :      0 : 291.515151515152 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Ahvaz') :      0 :           500.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Kerman') :      0 :           500.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Mashhad') :      0 :           100.0 : None
: False : False : NonNegativeReals
      ('B', 'Isfahan', 'Tabriz') :      0 : 891.515151515152 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Ahvaz') :      0 :          1000.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Kerman') :      0 : 398.181818181818 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Mashhad') :      0 :           300.0 : None
: False : False : NonNegativeReals
      ('B', 'Tehran', 'Tabriz') :      0 : 208.484848484848 : None
: False : False : NonNegativeReals
      G : Size=8, Index=Depots*Markets
          Key : Lower : Value : Upper : Fixed : Stale
: Domain
      ('Isfahan', 'Ahvaz') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Kerman') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Mashhad') :      0 :    4.0 : None : False : False
: NonNegativeIntegers
      ('Isfahan', 'Tabriz') :      0 :   10.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Ahvaz') :      0 :   10.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Kerman') :      0 :    6.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Mashhad') :      0 :    3.0 : None : False : False
: NonNegativeIntegers
      ('Tehran', 'Tabriz') :      0 :    5.0 : None : False : False
: NonNegativeIntegers
      l : Size=8, Index=Depots*Markets
          Key : Lower : Value : Upper : Fixed : Stale
: Domain
      ('Isfahan', 'Ahvaz') :      0 :    1.0 :    1 : False : False
: Binary
      ('Isfahan', 'Kerman') :      0 :    1.0 :    1 : False : False
: Binary
      ('Isfahan', 'Mashhad') :      0 :    1.0 :    1 : False : False
: Binary
      ('Isfahan', 'Tabriz') :      0 :    1.0 :    1 : False : False
: Binary
      ('Tehran', 'Ahvaz') :      0 :    1.0 :    1 : False : False

```



```

: Binary
    ('Tehran', 'Kerman') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Mashhad') :      0 :    1.0 :      1 : False : False
: Binary
    ('Tehran', 'Tabriz') :      0 :    1.0 :      1 : False : False
: Binary
    d : Size=2, Index={1, 2}
        Key : Lower : Value : Upper : Fixed : Stale : Domain
            1 :      0 :    1.0 :      1 : False : False : Binary
            2 :      0 :    0.0 :      1 : False : False : Binary
    R : Size=4, Index=Alloys*{1, 2}
        Key      : Lower : Value      : Upper : Fixed : Stale :
Domain
    ('A', 1) :      0 :          0.0 : None : False : False :
NonNegativeReals
    ('A', 2) :      0 :          0.0 : None : False : False :
NonNegativeReals
    ('B', 1) :      0 : 3391.51515151515 : None : False : False :
NonNegativeReals
    ('B', 2) :      0 :          0.0 : None : False : False :
NonNegativeReals

Objectives:
    revenue : Size=1, Index=None, Active=True
        Key : Active : Value
        None :    True : 1107067.5757575764

Constraints:
    Max_extracted_ore_limit : Size=3
        Key : Lower : Body      : Upper
            1 : None : 560.0 : 560
            2 : None : 1000.0 : 1000
            3 : None : 0.0 : 1440
    Alloy_sum_limit : Size=2
        Key : Lower : Body      : Upper
            A : None : 5.684341886080801e-13 : 0.0
            B : None : 1.0231815394945443e-12 : 0.0
    Metal_sum_limit_Z : Size=3
        Key : Lower : Body      : Upper
            1 : None : -2.984279490192421e-13 : 0.0
            2 : None : 0.0 : 0.0
            3 : None : 0.0 : 0.0
    Metal_sum_limit_F : Size=3
        Key : Lower : Body : Upper
            1 : None : 0.0 : 0.0
            2 : None : 0.0 : 0.0
            3 : None : 0.0 : 0.0
    Metal_sum_limit_C : Size=3

```

```

Key : Lower : Body : Upper
1 : None : -23.515151515152013 : 0.0
2 : None : -2.984279490192421e-13 : 0.0
3 : None : 0.0 : 0.0
Metal_sum_limit_A : Size=3
Key : Lower : Body : Upper
1 : None : 0.0 : 0.0
2 : None : 0.0 : 0.0
3 : None : 0.0 : 0.0
Metal_in_alloy_limit_A_Z_f : Size=1
Key : Lower : Body : Upper
None : None : -188.66666666666669 : 0.0
Metal_in_alloy_limit_A_Z_t : Size=1
Key : Lower : Body : Upper
None : None : -532.7878787878789 : 0.0
Metal_in_alloy_limit_A_C_f : Size=1
Key : Lower : Body : Upper
None : None : -217.15151515151467 : 0.0
Metal_in_alloy_limit_A_C_t : Size=1
Key : Lower : Body : Upper
None : None : -684.66666666666673 : 0.0
Metal_in_alloy_limit_A_A_f : Size=1
Key : Lower : Body : Upper
None : None : 1.1368683772161603e-13 : 0.0
Metal_in_alloy_limit_A_A_t : Size=1
Key : Lower : Body : Upper
None : None : -405.818181818182 : 0.0
Metal_in_alloy_limit_A_F_f : Size=1
Key : Lower : Body : Upper
None : None : 0.0 : 0.0
Metal_in_alloy_limit_A_F_t : Size=1
Key : Lower : Body : Upper
None : None : -225.4545454545455 : 0.0
Metal_in_alloy_limit_B_Z_f : Size=1
Key : Lower : Body : Upper
None : None : -101.3333333333333 : 0.0
Metal_in_alloy_limit_B_Z_t : Size=1
Key : Lower : Body : Upper
None : None : -405.33333333333405 : 0.0
Metal_in_alloy_limit_B_C_f : Size=1
Key : Lower : Body : Upper
None : None : 4.547473508864641e-13 : 0.0
Metal_in_alloy_limit_B_C_t : Size=1
Key : Lower : Body : Upper
None : None : -329.33333333333405 : 0.0
Metal_in_alloy_limit_B_A_f : Size=1
Key : Lower : Body : Upper
None : None : 0.0 : 0.0
Metal_in_alloy_limit_B_A_t : Size=1

```

```

Key : Lower : Body : Upper
None : None : -354.6666666666669 : 0.0
Metal_in_alloy_limit_B_F_f : Size=1
Key : Lower : Body : Upper
None : None : 1.7053025658242404e-13 : 0.0
Metal_in_alloy_limit_B_F_t : Size=1
Key : Lower : Body : Upper
None : None : -126.66666666666691 : 0.0
Export_from_main_fac_limit : Size=2
Key : Lower : Body : Upper
A : None : -1.0231815394945443e-12 : 0.0
B : None : 0.0 : 0.0
buy_from_fac_limit_f : Size=2
Key : Lower : Body : Upper
1 : None : -1391.5151515151501 : 0.0
2 : None : 0.0 : 0.0
buy_from_fac_limit_t : Size=2
Key : Lower : Body : Upper
1 : None : -1608.48484848485 : 0.0
2 : None : 0.0 : 0.0
container_limit : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -508.4848484848501 : 0.0
(1, 'Tehran') : None : 0.0 : 0.0
(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : -91.51515151515201 : 0.0
('Main', 'Tehran') : None : 0.0 : 0.0
transportation_limit_t : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -5.0 : 0.0
(1, 'Tehran') : None : -16.0 : 0.0
(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : -15.0 : 0.0
('Main', 'Tehran') : None : -10.0 : 0.0
transportation_limit_f : Size=6
Key : Lower : Body : Upper
(1, 'Isfahan') : None : -15.0 : 0.0
(1, 'Tehran') : None : -4.0 : 0.0
(2, 'Isfahan') : None : 0.0 : 0.0
(2, 'Tehran') : None : 0.0 : 0.0
('Main', 'Isfahan') : None : 0.0 : 0.0
('Main', 'Tehran') : None : -5.0 : 0.0
transportation_limit2 : Size=2
Key : Lower : Body : Upper
Isfahan : 30 : 30.0 : 70.0
Tehran : 20 : 24.0 : 65.0
transp_from_dep_to_market_limit : Size=4

```

Key	: Lower	: Body	: Upper
('A', 'Isfahan')	: None	: 0.0	: 0.0
('A', 'Tehran')	: None	: 1.0800249583553523e-12	: 0.0
('B', 'Isfahan')	: None	: 2.0463630789890885e-12	: 0.0
('B', 'Tehran')	: None	: -9.094947017729282e-13	: 0.0
container_limit2 : Size=8			
Key	: Lower	: Body	:
Upper			
('Isfahan', 'Ahvaz')	: None	:	0.0 :
0.0			
('Isfahan', 'Kerman')	: None	:	0.0 :
0.0			
('Isfahan', 'Mashhad')	: None	:	0.0 :
0.0			
('Isfahan', 'Tabriz')	: None	: -4.263256414560601e-14	:
0.0			
('Tehran', 'Ahvaz')	: None	:	0.0 :
0.0			
('Tehran', 'Kerman')	: None	: 2.842170943040401e-14	:
0.0			
('Tehran', 'Mashhad')	: None	:	0.0 :
0.0			
('Tehran', 'Tabriz')	: None	:	0.0 :
0.0			
market_sell_limit_f : Size=8			
Key	: Lower	: Body	: Upper
('Isfahan', 'Ahvaz')	: None	: 0.0	: 0.0
('Isfahan', 'Kerman')	: None	: 0.0	: 0.0
('Isfahan', 'Mashhad')	: None	: 0.0	: 0.0
('Isfahan', 'Tabriz')	: None	: 0.0	: 0.0
('Tehran', 'Ahvaz')	: None	: 0.0	: 0.0
('Tehran', 'Kerman')	: None	: 0.0	: 0.0
('Tehran', 'Mashhad')	: None	: 0.0	: 0.0
('Tehran', 'Tabriz')	: None	: 0.0	: 0.0
market_sell_limit_t : Size=8			
Key	: Lower	: Body	: Upper
('Isfahan', 'Ahvaz')	: None	: 0.0	: 0.0
('Isfahan', 'Kerman')	: None	: 0.0	: 0.0
('Isfahan', 'Mashhad')	: None	: 0.0	: 0.0
('Isfahan', 'Tabriz')	: None	: 0.0	: 0.0
('Tehran', 'Ahvaz')	: None	: 0.0	: 0.0
('Tehran', 'Kerman')	: None	: 0.0	: 0.0
('Tehran', 'Mashhad')	: None	: 0.0	: 0.0
('Tehran', 'Tabriz')	: None	: 0.0	: 0.0
max_market_demand_limit : Size=8			
Key	: Lower	: Body	: Upper
('Ahvaz', 'A')	: None	:	0.0 : 1500.0
('Ahvaz', 'B')	: None	: 1500.0	: 1500.0
('Kerman', 'A')	: None	: 201.818181818182	: 800.0

```

      ('Kerman', 'B') : None : 898.181818181818 : 1200.0
      ('Mashhad', 'A') : None : 300.0 : 600.0
      ('Mashhad', 'B') : None : 400.0 : 400.0
      ('Tabriz', 'A') : None : 400.0 : 1400.0
      ('Tabriz', 'B') : None : 1100.0 : 1100.0
discount_limit_1 : Size=2
  Key : Lower : Body : Upper
    1 : None : -1608.4848484838499 : 0.0
    2 : None : -2999.999999999 : 0.0
discount_limit_2 : Size=2
  Key : Lower : Body : Upper
    1 : None : -891.5151515151501 : 0.0
    2 : None : 0.0 : 0.0
discount_limit_3 : Size=4
  Key : Lower : Body : Upper
    (1, 'A') : None : 0.0 : 0.0
    (1, 'B') : None : 0.0 : 0.0
    (2, 'A') : None : 0.0 : 0.0
    (2, 'B') : None : 0.0 : 0.0
discount_limit_4 : Size=2
  Key : Lower : Body : Upper
    1 : None : -999996607.4848485 : 0.0
    2 : None : 0.0 : 0.0
results saved in results.yaml

```

## explanation

In the report above, we have three parts, **variables**, **objective** and **constraints**, you can see value of variables in **Value** column of variables section, and as you see, value of objective(revenue) is equal to **1107067.57575758**

you can see result file using code below:

```

!cat results.yaml

# =====
# = Solver Results                                     =
# =====
# -----
#   Problem Information
# -----
Problem:
- Name: unknown
  Lower bound: 1107067.57575758
  Upper bound: 1107067.57575758
  Number of objectives: 1
  Number of constraints: 107
  Number of variables: 88
  Number of nonzeros: 342

```

```

Sense: maximize
# -----
# Solver Information
# -----
Solver:
- Status: ok
  Termination condition: optimal
  Statistics:
    Branch and bound:
      Number of bounded subproblems: 29
      Number of created subproblems: 29
  Error rc: 0
  Time: 0.012421846389770508
# -----
# Solution Information
# -----
Solution:
- number of solutions: 0
  number of solutions displayed: 0

```

## Sensitivity Analysis

here we produce sensitivity analysis, be aware that, here we should eliminate binaries and integers to make sensitivity analysis, because glpk does not work with MIP(Mixed Integer Problem) problems, we will use this file later on.

```

!glpsol -m model.lp --lp --ranges sensit.sen

GLPSOL--GLPK LP/MIP Solver 5.0
Parameter(s) specified in the command line:
  -m model.lp --lp --ranges sensit.sen
Reading problem data from 'model.lp'...
97 rows, 82 columns, 304 non-zeros
731 lines were read
GLPK Simplex Optimizer 5.0
97 rows, 82 columns, 304 non-zeros
Preprocessing...
89 rows, 82 columns, 286 non-zeros
Scaling...
  A: min|aij| = 5.000e-02  max|aij| = 6.000e+03  ratio = 1.200e+05
  GM: min|aij| = 5.953e-01  max|aij| = 1.680e+00  ratio = 2.822e+00
  EQ: min|aij| = 3.704e-01  max|aij| = 1.000e+00  ratio = 2.700e+00
Constructing initial basis...
Size of triangular part is 89
      0: obj = -0.000000000e+00  inf = 7.935e+02 (2)
      8: obj = -2.613400000e+05  inf = 0.000e+00 (0)
*    95: obj = 1.040631552e+06  inf = 0.000e+00 (0)
OPTIMAL LP SOLUTION FOUND
Time used: 0.0 secs

```

Memory used: 0.2 Mb (173261 bytes)  
Write sensitivity analysis report to 'sensit.sen'...

## Sensit file

to see sensit.sen, we run code below:

```
!cat sensit.sen

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT
Page 1

Problem:
Objective: revenue = 1660813.37 (MAXimum)

No. Row name      St      Activity      Slack      Lower bound
Activity          Obj coef  Obj value at Limiting
range            range    break point variable
-----
1 c_u_Max_extracted_ore_limit(1)_
      NU      560.00000      .      -Inf
471.42857      -209.57253      1.64225e+06
c_u_max_market_demand_limit(Mashhad_B)_
      209.57253      560.00000
628.57143      +Inf      1.67518e+06 g(B_Isfahan_Mashhad)

2 c_u_Max_extracted_ore_limit(2)_
      NU      1000.00000      .      -Inf
896.66667      -325.65010      1.62716e+06
c_u_max_market_demand_limit(Mashhad_B)_
      325.65010      1000.00000
1077.74879      +Inf      1.68613e+06 c_u_discount_limit_4(1)_

3 c_u_Max_extracted_ore_limit(3)_
      NU      1440.00000      .      -Inf
1316.00000      -127.51434      1.645e+06
c_u_max_market_demand_limit(Mashhad_B)_
      127.51434      1440.00000
1536.00000      +Inf      1.67305e+06 g(B_Isfahan_Mashhad)

4 c_u_Alloy_sum_limit(A)_
      NU      .      .      -Inf -
576.96970      .      1.66081e+06 c_u_Metal_sum_limit_Z(3)_
      .
225.69697      +Inf      1.66081e+06 Z(3_A)

5 c_u_Alloy_sum_limit(B)_
      NU      .      .      -Inf -
```

```

533.33333      .      1.66081e+06 c_u_Metal_in_alloy_limit_B_Z_t_
133.33333      +Inf   1.66081e+06 c_u_Metal_in_alloy_limit_B_Z_f_
      6 c_u_Metal_sum_limit_Z(1)_
      NU      .      -Inf      -
140.00000      .      1.66081e+06 Z(1_A)
225.69697      +Inf   1.66081e+06 Z(3_A)
      7 c_u_Metal_sum_limit_Z(2)_
      NU      .      -Inf      -
150.00000      .      1.66081e+06 Z(2_A)
225.69697      +Inf   1.66081e+06 Z(3_A)
      8 c_u_Metal_sum_limit_Z(3)_
      BS      -576.96970      576.96970      -Inf      -
648.96970      .      1.66081e+06 C(3_A)
436.96970      .      1.66081e+06 c_u_Metal_sum_limit_Z(1)_
      9 c_u_Metal_sum_limit_F(1)_
      NU      .      -Inf
-28.00000      -1097.37778      1.63009e+06 F(1_B)
      1097.37778      .
21.42857      +Inf   1.68433e+06 C(2_A)
      10 c_u_Metal_sum_limit_F(2)_
      NU      .      -Inf      -
139.63636      -1097.37778      1.50758e+06 t(B_Main_Isfahan)
      1097.37778      .
21.42857      +Inf   1.68433e+06 C(2_A)

```

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 2

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj	coef	Obj value at Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
-----					
11	c_u_Metal_sum_limit_F(3)_				
		NU	.		-Inf
-72.00000	-1097.37778		1.5818e+06 F(3_B)		
			1097.37778	.	



21.42857	+Inf	1.68433e+06	C(2_A)		
12 c_u_Metal_sum_limit_C(1)_					
	NU	.	.	-Inf	-
168.00000	.	1.66081e+06	C(1_A)		
225.69697	+Inf	1.66081e+06	Z(3_A)		
13 c_u_Metal_sum_limit_C(2)_					
	NU	.	.	-Inf	
-16.66667	.	1.66081e+06	C(2_A)		
225.69697	+Inf	1.66081e+06	Z(3_A)		
14 c_u_Metal_sum_limit_C(3)_					
	BS	-72.00000	72.00000	-Inf	
-72.00000	-Inf	+Inf	.		
153.69697	.	1.66081e+06	C(3_A)		
15 c_u_Metal_sum_limit_A(1)_					
	NU	.	.	-Inf	
-31.00000	-570.58182	1.64313e+06			
c_u_max_market_demand_limit(Mashhad_B)_			570.58182	.	
24.00000	+Inf	1.67451e+06	g(B_Isfahan_Mashhad)		
16 c_u_Metal_sum_limit_A(2)_					
	NU	.	.	-Inf	
-31.00000	-570.58182	1.64313e+06			
c_u_max_market_demand_limit(Mashhad_B)_			570.58182	.	
24.00000	+Inf	1.67451e+06	g(B_Isfahan_Mashhad)		
17 c_u_Metal_sum_limit_A(3)_					
	NU	.	.	-Inf	
-31.00000	-570.58182	1.64313e+06			
c_u_max_market_demand_limit(Mashhad_B)_			570.58182	.	
24.00000	+Inf	1.67451e+06	g(B_Isfahan_Mashhad)		
18 c_u_Metal_in_alloy_limit_A_Z_f_					
	BS	-515.69697	515.69697	-Inf	-
683.69697	.	1.66081e+06	c_u_Metal_sum_limit_C(1)_		
443.69697	.	1.66081e+06	C(3_A)		-
19 c_u_Metal_in_alloy_limit_A_Z_t_					
	BS	-729.39394	729.39394	-Inf	-
801.39394	.	1.66081e+06	C(3_A)		

561.39394 . 1.66081e+06 c\_u\_Metal\_sum\_limit\_C(1)\_ -

20 c\_u\_Metal\_in\_alloy\_limit\_A\_C\_f\_  
BS -184.66667 184.66667 -Inf -  
256.66667 . 1.66081e+06 C(3\_A)

-16.66667 . 1.66081e+06 c\_u\_Metal\_sum\_limit\_C(1)\_

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 3

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at	Limiting		
range	range	break point	variable	Marginal	Upper bound

21 c\_u\_Metal\_in\_alloy\_limit\_A\_C\_t\_  
BS -1371.69697 1371.69697 -Inf -  
1539.69697 . 1.66081e+06 c\_u\_Metal\_sum\_limit\_C(1)\_  
1299.69697 . 1.66081e+06 C(3\_A)

22 c\_u\_Metal\_in\_alloy\_limit\_A\_A\_f\_  
NU . -Inf  
-31.00000 -570.58182 1.64313e+06  
c\_u\_max\_market\_demand\_limit(Mashhad\_B)\_  
570.58182 .  
24.00000 +Inf 1.67451e+06 g(B\_Isfahan\_Mashhad)

23 c\_u\_Metal\_in\_alloy\_limit\_A\_A\_t\_  
BS -700.36364 700.36364 -Inf -  
700.36364 -Inf +Inf  
644.00000 313.82000 1.44103e+06 c\_u\_Metal\_in\_alloy\_limit\_A\_A\_f\_

24 c\_u\_Metal\_in\_alloy\_limit\_A\_F\_f\_  
BS . -Inf  
-28.00000 -1097.37778 1.66081e+06 F(1\_A)  
+Inf 1.66081e+06

25 c\_u\_Metal\_in\_alloy\_limit\_A\_F\_t\_  
BS -389.09091 389.09091 -Inf -  
389.09091 -Inf +Inf

```

361.09091    1097.37778    1.23383e+06 F(1_A)

  26 c_u_Metal_in_alloy_limit_B_Z_f_
      BS      -133.33333      133.33333      -Inf      -
666.66667      .      1.66081e+06 c_u_Alloy_sum_limit(B)_
116.66667      .      1.66081e+06 c_u_Metal_in_alloy_limit_B_C_f_

  27 c_u_Metal_in_alloy_limit_B_Z_t_
      BS      -533.33333      533.33333      -Inf      -
550.00000      .      1.66081e+06 c_u_Metal_in_alloy_limit_B_C_f_
43.63636      .      1.66081e+06 c_u_Alloy_sum_limit(B)_

  28 c_u_Metal_in_alloy_limit_B_C_f_
      NU      .      .      -Inf
-16.66667      .      1.66081e+06 C(2_A)
225.69697      +Inf      1.66081e+06 Z(3_A)

  29 c_u_Metal_in_alloy_limit_B_C_t_
      BS      -433.33333      433.33333      -Inf      -
433.33333      -Inf      +Inf
416.66667      .      1.66081e+06 c_u_Metal_in_alloy_limit_B_C_f_

  30 c_u_Metal_in_alloy_limit_B_A_f_
      BS      .      .      -Inf
-31.00000      -570.58182      1.66081e+06 A(1_B)
.      +Inf      1.66081e+06

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 4

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No. Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at Limiting	Marginal	Upper bound
range	range	break point variable		
-----				
-----				
31 c_u_Metal_in_alloy_limit_B_A_t_	BS	-466.66667	466.66667	-Inf
466.66667	-Inf	+Inf		
435.66667	570.58182	1.39454e+06 A(1_B)		

```

32 c_u_Metal_in_alloy_limit_B_F_f_
    NU      .      -Inf
-92.30769   -1097.37778   1.55952e+06 c_u_Metal_in_alloy_limit_B_Z_f_
                                1097.37778   .
21.42857    +Inf    1.68433e+06 C(2_A)

33 c_u_Metal_in_alloy_limit_B_F_t_
    BS      -166.66667   166.66667   -Inf   -
166.66667   -Inf      +Inf

-23.07692    705.45714   1.54324e+06 c_u_Metal_in_alloy_limit_B_F_f_

34 c_u_Export_from_main_fac_limit(A)_
    NU      .      -Inf
-56.36364    -313.82000   1.64313e+06
c_u_max_market_demand_limit(Mashhad_B)_
                                313.82000   .
43.63636    +Inf    1.67451e+06 g(B_Isfahan_Mashhad)

35 c_u_Export_from_main_fac_limit(B)_
    NU      .      -Inf   -
310.30303    -493.82000   1.50758e+06 t(B_Main_Isfahan)
                                493.82000   .
76.96345    +Inf    1.69882e+06 c_u_discount_limit_4(1)_

36 c_u_buy_from_fac_limit_f(1)_
    BS      -1517.97980   1517.97980   -Inf   -
1517.97980    -.42000   1.66145e+06 c_u_container_limit(2_Isfahan)_
                                .      .
1414.54545    .06000   1.66072e+06
c_u_transportation_limit_t(1_Tehran)_

37 c_u_buy_from_fac_limit_f(2)_
    BS      .      -Inf
.      -.26400   1.66081e+06 t(B_2_Tehran)
                                .
.      .03600   1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_

38 c_u_buy_from_fac_limit_t(1)_
    BS      -70.50505   70.50505   -Inf   -
329.09091    -.02400   1.66082e+06
c_u_transportation_limit_t(1_Tehran)_
-70.50505    .16800   1.6608e+06 c_u_container_limit(2_Isfahan)_

39 c_u_buy_from_fac_limit_t(2)_
    BS      .      -Inf
.      -.01500   1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_

```

```

.          .11000    1.66081e+06 t(B_2_Tehran)
.
40 c_u_container_limit(1_Tehran)_
    NU          .          -Inf -
310.30303      -1.82000    1.66025e+06 t(B_Main_Isfahan)
                                1.82000
84.60606          +Inf    1.66097e+06 c_u_buy_from_fac_limit_t(1)_

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 5

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at Limiting	Marginal	Upper bound	
range	range	break point variable			
-----					
41	c_u_container_limit(1_Isfahan)_				
	BS	-600.00000	600.00000	-Inf	-
1223.03030	-2.12000	1.66209e+06			
r_l_transportation_limit2(Isfahan)_					
600.00000	.14000	1.66073e+06			
c_u_container_limit(2_Isfahan)_					
42	c_u_container_limit(2_Tehran)_				
	NU	.	.	-Inf	
.	-2.40000	1.66081e+06			
c_u_transportation_limit_t(2_Tehran)_			2.40000	.	
.	+Inf	1.66081e+06	B(2_Tehran)		
43	c_u_container_limit(2_Isfahan)_				
	NU	.	.	-Inf	
.	-.14000	1.66081e+06	c_u_buy_from_fac_limit_f(2)_		
			.14000	.	
.	+Inf	1.66081e+06	c_u_buy_from_fac_limit_t(2)_		
44	c_u_container_limit(Main_Tehran)_				
	NU	.	.	-Inf	-
1188.48485	-2.00000	1.65844e+06			
c_u_transportation_limit_t(Main_Tehran)_			2.00000	.	
311.51515	+Inf	1.66144e+06			
c_u_transportation_limit_f(Main_Tehran)_					
45	c_u_container_limit(Main_Isfahan)_				

```

      NU      .      .      -Inf
-84.60606      -.18000      1.6608e+06 c_u_buy_from_fac_limit_t(1)_
      .18000      .
623.03030      +Inf      1.66093e+06
c_u_transportation_limit_f(Main_Tehran)_
      46 c_u_transportation_limit_t(1_Tehran)_
      NU      .      .      -Inf
-3.10303      -2.00000      1.66081e+06 t(B_Main_Isfahan)
      2.00000      .
.84606      +Inf      1.66082e+06 c_u_buy_from_fac_limit_t(1)_
      47 c_u_transportation_limit_t(1_Isfahan)_
      NU      .      .      -Inf
-6.23030      -2.00000      1.6608e+06
c_u_transportation_limit_f(Main_Tehran)_
      2.00000      .
.84606      +Inf      1.66082e+06 c_u_buy_from_fac_limit_t(1)_
      48 c_u_transportation_limit_t(2_Tehran)_
      BS      .      .      -Inf
.      -3.60000      1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_
      .
.      16.50000      1.66081e+06 t(B_2_Tehran)
      49 c_u_transportation_limit_t(2_Isfahan)_
      NU      .      .      -Inf
.      -6.00000      1.66081e+06
c_u_transportation_limit_f(2_Isfahan)_
      6.00000      .
.      +Inf      1.66081e+06 c_u_buy_from_fac_limit_t(2)_
      50 c_u_transportation_limit_t(Main_Tehran)_
      BS      -11.88485      11.88485      -Inf
-15.00000      -4.00000      1.66086e+06
c_u_transportation_limit_t(1_Isfahan)_
      .
-6.00000      .      1.66081e+06 h(Main)

```

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 6

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at	Limiting		
			Marginal	Upper bound	
range	range	break point variable			

```

-----
51 c_u_transportation_limit_t(Main_Isfahan)_
      BS      -5.88485      5.88485      -Inf
-7.43636      -4.00000      1.66084e+06
c_u_transportation_limit_t(1_Tehran)_
      .
6.00000      .      1.66081e+06 h(Main)

52 c_u_transportation_limit_f(1_Tehran)_
      BS      -10.58990      10.58990      -Inf
-12.66667      -6.00000      1.66088e+06
c_u_transportation_limit_t(1_Isfahan)_
      .
-8.52121      3.00000      1.66078e+06
c_u_transportation_limit_t(1_Tehran)_

53 c_u_transportation_limit_f(1_Isfahan)_
      BS      -10.58990      10.58990      -Inf
-11.62424      -6.00000      1.66088e+06
c_u_transportation_limit_t(1_Tehran)_
      .
-6.43636      3.00000      1.66078e+06
c_u_transportation_limit_t(1_Isfahan)_

54 c_u_transportation_limit_f(2_Tehran)_
      BS      .      .      -Inf
.      -33.00000      1.66081e+06 t(B_2_Tehran)
      .
.      18.00000      1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_

55 c_u_transportation_limit_f(2_Isfahan)_
      BS      .      .      -Inf
.      -21.00000      1.66081e+06 c_u_container_limit(2_Isfahan)_
      .
.      18.00000      1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_

56 c_u_transportation_limit_f(Main_Tehran)_
      BS      -3.11515      3.11515      -Inf
-4.58636      .      1.66081e+06 h(Main)
      .
2.00000      4.00000      1.6608e+06
c_u_transportation_limit_t(1_Isfahan)_

57 c_u_transportation_limit_f(Main_Isfahan)_
      BS      -9.11515      9.11515      -Inf
-10.58636      .      1.66081e+06 h(Main)
      .

```

```

-7.56364      4.00000    1.66078e+06
c_u_transportation_limit_t(1_Tehran)_

58 r_l_transportation_limit2(Tehran)_
      BS      24.00000    -4.00000    20.00000
23.56364 -18318.00000    1.22118e+06 l(Tehran_Tabriz)
      .      +Inf
27.10303    182.00000    1.66518e+06 c_u_container_limit(1_Tehran)_

59 r_u_transportation_limit2(Tehran)_
      BS      24.00000    41.00000    -Inf
23.56364 -18318.00000    1.22118e+06 l(Tehran_Tabriz)
      .      65.00000
27.10303    182.00000    1.66518e+06 c_u_container_limit(1_Tehran)_

60 r_l_transportation_limit2(Isfahan)_
      NL      30.00000    .      30.00000
29.15394    -Inf    1.66099e+06 c_u_buy_from_fac_limit_t(1)_
      -212.00000    +Inf
36.23030    212.00000    1.65949e+06
c_u_transportation_limit_f(Main_Tehran)_

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 7

Problem:  
Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
61	r_u_transportation_limit2(Isfahan)_	BS	30.00000	40.00000	-Inf
30.00000	-Inf	-Inf			70.00000
36.23030	212.00000	1.66717e+06			
r_l_transportation_limit2(Isfahan)_					
62	c_u_transp_from_dep_to_market_limit(A_Tehran)_	NU	.	.	-Inf
-56.36364	-315.82000	1.64301e+06			
c_u_max_market_demand_limit(Mashhad_B)_					
43.63636	+Inf	1.67459e+06	g(B_Isfahan_Mashhad)		
63	c_u_transp_from_dep_to_market_limit(A_Isfahan)_	NU	.	.	-Inf



```

-56.36364      -314.00000      1.64312e+06
c_u_max_market_demand_limit(Mashhad_B)_
                                     314.00000      .
43.63636              +Inf      1.67452e+06 g(B_Isfahan_Mashhad)

64 c_u_transp_from_dep_to_market_limit(B_Tehran)_
      NU      .      .      -Inf      -
310.30303      -495.82000      1.50696e+06 t(B_Main_Isfahan)
                                     495.82000      .
76.96345              +Inf      1.69897e+06 c_u_discount_limit_4(1)_

65 c_u_transp_from_dep_to_market_limit(B_Isfahan)_
      NU      .      .      -Inf
-70.50505      -494.00000      1.62598e+06 c_u_buy_from_fac_limit_t(1)_
                                     494.00000      .
76.96345              +Inf      1.69883e+06 c_u_discount_limit_4(1)_

66 c_u_container_limit2(Tehran_Mashhad)_
      NU      .      .      -Inf
-76.96345      -204.18000      1.6451e+06 c_u_discount_limit_4(1)_
                                     204.18000      .
56.36364              +Inf      1.67232e+06
c_u_max_market_demand_limit(Mashhad_B)_

67 c_u_container_limit2(Tehran_Kerman)_
      NU      .      .      -Inf
-76.96345      -194.18000      1.64587e+06 c_u_discount_limit_4(1)_
                                     194.18000      .
310.30303              +Inf      1.72107e+06 t(B_Main_Isfahan)

68 c_u_container_limit2(Tehran_Ahvaz)_
      NU      .      .      -Inf
-76.96345      -234.18000      1.64279e+06 c_u_discount_limit_4(1)_
                                     234.18000      .
.              +Inf      1.66081e+06
c_u_max_market_demand_limit(Ahvaz_B)_

69 c_u_container_limit2(Tehran_Tabriz)_
      NU      .      .      -Inf
-43.63636      -184.18000      1.65278e+06 g(B_Isfahan_Mashhad)
                                     184.18000      .
56.36364              +Inf      1.67119e+06
c_u_max_market_demand_limit(Mashhad_B)_

70 c_u_container_limit2(Isfahan_Mashhad)_
      NU      .      .      -Inf
-43.63636      -206.00000      1.65182e+06 g(B_Isfahan_Mashhad)
                                     206.00000      .
56.36364              +Inf      1.67242e+06
c_u_max_market_demand_limit(Mashhad_B)_

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 8

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj	coef	Obj value at Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
71	c_u_container_limit2(Isfahan_Kerman)_				
		NU	.	.	-Inf
-76.96345	-196.00000	1.64573e+06	c_u_discount_limit_4(1)_	196.00000	.
70.50505		+Inf	1.67463e+06	c_u_buy_from_fac_limit_t(1)_	
72	c_u_container_limit2(Isfahan_Ahvaz)_				
		NU	.	.	-Inf
-76.96345	-236.00000	1.64265e+06	c_u_discount_limit_4(1)_	236.00000	.
.		+Inf	1.66081e+06		
c_u_max_market_demand_limit(Ahvaz_B)_					
73	c_u_container_limit2(Isfahan_Tabriz)_				
		NU	.	.	-Inf
-43.63636	-186.00000	1.6527e+06	g(B_Isfahan_Mashhad)	186.00000	.
56.36364		+Inf	1.6713e+06		
c_u_max_market_demand_limit(Mashhad_B)_					
74	c_u_market_sell_limit_f(Tehran_Mashhad)_				
		BS	.	.	-Inf
.		-Inf	1.66081e+06		
.76963	20308.00000	1.66081e+06			
c_u_market_sell_limit_t(Tehran_Mashhad)_					
75	c_u_market_sell_limit_f(Tehran_Kerman)_				
		BS	.	.	-Inf
.		-Inf	1.66081e+06		
.76963	19333.00000	1.66081e+06			
c_u_market_sell_limit_t(Tehran_Kerman)_					
76	c_u_market_sell_limit_f(Tehran_Ahvaz)_				
		BS	.	.	-Inf
.		-Inf	1.66081e+06		

```

.76963    23298.00000    1.66081e+06
c_u_market_sell_limit_t(Tehran_Ahvaz)_

77 c_u_market_sell_limit_f(Tehran_Tabriz)_
    BS . -Inf
    -Inf 1.66081e+06

.43636    18318.00000    1.66081e+06
c_u_market_sell_limit_t(Tehran_Tabriz)_

78 c_u_market_sell_limit_f(Isfahan_Mashhad)_
    BS . -Inf
    -Inf 1.66081e+06

.43636    20500.00000    1.66081e+06
c_u_market_sell_limit_t(Isfahan_Mashhad)_

79 c_u_market_sell_limit_f(Isfahan_Kerman)_
    BS . -Inf
    -Inf 1.66081e+06

.76963    19500.00000    1.66081e+06
c_u_market_sell_limit_t(Isfahan_Kerman)_

80 c_u_market_sell_limit_f(Isfahan_Ahvaz)_
    BS . -Inf
    -Inf 1.66081e+06

.76963    23490.00000    1.66081e+06
c_u_market_sell_limit_t(Isfahan_Ahvaz)_

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 9

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No. Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at Limiting	Marginal	Upper bound
range	range	break point variable		
-----				
81 c_u_market_sell_limit_f(Isfahan_Tabriz)_	BS	.	.	-Inf
.	-Inf	1.66081e+06	.	.
-----				
.43636    18510.00000    1.66081e+06				
c_u_market_sell_limit_t(Isfahan_Tabriz)_				

```

82 c_u_market_sell_limit_t(Tehran_Mashhad)_
    NU . . -Inf
. -20308.00000 1.66081e+06
c_u_market_sell_limit_f(Tehran_Mashhad)_
    20308.00000 .
.56364 +Inf 1.67226e+06
c_u_max_market_demand_limit(Mashhad_B)_

83 c_u_market_sell_limit_t(Tehran_Kerman)_
    NU . . -Inf
. -19333.00000 1.66081e+06
c_u_market_sell_limit_f(Tehran_Kerman)_
    19333.00000 .
3.10303 +Inf 1.7208e+06 t(B_Main_Isfahan)

84 c_u_market_sell_limit_t(Tehran_Ahvaz)_
    NU . . -Inf
. -23298.00000 1.66081e+06
c_u_market_sell_limit_f(Tehran_Ahvaz)_
    23298.00000 .
. +Inf 1.66081e+06
c_u_max_market_demand_limit(Ahvaz_B)_

85 c_u_market_sell_limit_t(Tehran_Tabriz)_
    NU . . -Inf
. -18318.00000 1.66081e+06
c_u_market_sell_limit_f(Tehran_Tabriz)_
    18318.00000 .
.56364 +Inf 1.67114e+06
c_u_max_market_demand_limit(Mashhad_B)_

86 c_u_market_sell_limit_t(Isfahan_Mashhad)_
    NU . . -Inf
. -20500.00000 1.66081e+06
c_u_market_sell_limit_f(Isfahan_Mashhad)_
    20500.00000 .
.56364 +Inf 1.67237e+06
c_u_max_market_demand_limit(Mashhad_B)_

87 c_u_market_sell_limit_t(Isfahan_Kerman)_
    NU . . -Inf
. -19500.00000 1.66081e+06
c_u_market_sell_limit_f(Isfahan_Kerman)_
    19500.00000 .
.70505 +Inf 1.67456e+06 c_u_buy_from_fac_limit_t(1)_

88 c_u_market_sell_limit_t(Isfahan_Ahvaz)_
    NU . . -Inf
. -23490.00000 1.66081e+06

```

```

c_u_market_sell_limit_f(Isfahan_Ahvaz)_
23490.00000 .
. +Inf 1.66081e+06
c_u_max_market_demand_limit(Ahvaz_B)_
89 c_u_market_sell_limit_t(Isfahan_Tabriz)_
NU . -Inf
. -18510.00000 1.66081e+06
c_u_market_sell_limit_f(Isfahan_Tabriz)_
18510.00000 .
.56364 +Inf 1.67125e+06
c_u_max_market_demand_limit(Mashhad_B)_
90 c_u_max_market_demand_limit(Mashhad_A)_
BS 356.36364 243.63636 -Inf
300.00000 -30.00000 1.65012e+06
c_u_max_market_demand_limit(Tabriz_B)_
. 600.00000
400.00000 30.00000 1.6715e+06
c_u_max_market_demand_limit(Kerman_A)_
GLPK 5.0 - SENSITIVITY ANALYSIS REPORT
Page 10

```

Problem:  
Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
91	c_u_max_market_demand_limit(Mashhad_B)_	BS	343.63636	56.36364	-Inf
300.00000	-30.00000	1.6505e+06			
c_u_max_market_demand_limit(Kerman_A)_					400.00000
700.00000	30.00000	1.67112e+06			
c_u_max_market_demand_limit(Tabriz_B)_					
92	c_u_max_market_demand_limit(Kerman_A)_	NU	800.00000		-Inf
756.36364	-30.00000	1.6595e+06	g(B_Isfahan_Mashhad)	30.00000	800.00000
856.36364	+Inf	1.6625e+06			
c_u_max_market_demand_limit(Mashhad_B)_					
93	c_u_max_market_demand_limit(Kerman_B)_	BS	300.00000	900.00000	-Inf

```

223.03655      -193.33000      1.60281e+06 l(Tehran_Kerman)
.              1200.00000
343.63636      30.00000      1.66981e+06
c_u_max_market_demand_limit(Kerman_A)_
94 c_u_max_market_demand_limit(Ahvaz_A)_
      BS      .      1500.00000      -Inf
.      -Inf      1.66081e+06
.              1500.00000
56.36364      60.00000      1.66081e+06 g(A_Isfahan_Ahvaz)
95 c_u_max_market_demand_limit(Ahvaz_B)_
      BS      1500.00000      .      -Inf
1443.63636     -60.00000      1.57081e+06 g(A_Isfahan_Ahvaz)
.              1500.00000
1500.00000      +Inf      +Inf
96 c_u_max_market_demand_limit(Tabriz_A)_
      BS      400.00000      1000.00000      -Inf
356.36364     -183.18000      1.58754e+06 l(Tehran_Tabriz)
.              1400.00000
456.36364      30.00000      1.67281e+06
c_u_max_market_demand_limit(Tabriz_B)_
97 c_u_max_market_demand_limit(Tabriz_B)_
      NU      1100.00000      .      -Inf
1043.63636     -30.00000      1.65912e+06
c_u_max_market_demand_limit(Mashhad_B)_
      30.00000      1100.00000
1143.63636      +Inf      1.66212e+06 g(B_Isfahan_Mashhad)
98 c_u_discount_limit_1(1)_
      NU      2500.00000      .      -Inf
76.96970      .      1.66081e+06 d(1)
.              2500.00000
2576.96325      +Inf      1.66081e+06 c_u_discount_limit_4(1)_
99 c_u_discount_limit_1(2)_
      BS      .      3000.00000      -Inf
.      -Inf      1.66081e+06
.              3000.00000
.      19.14006      1.66081e+06 c_u_discount_limit_2(2)_
100 c_u_discount_limit_2(1)_
      BS      -2500.00000      2500.00000      -Inf      -
2500.00000      -Inf      +Inf
.              .
-76.96970      .      1.66081e+06 c_u_discount_limit_1(1)_

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 11

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Row name	St	Activity	Slack	Lower bound
Activity	Obj	coef	Obj value at Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
101	c_u_discount_limit_2(2)_	NU	.	.	-Inf
.	-19.14006	1.66081e+06	c_u_buy_from_fac_limit_f(2)_	19.14006	.
.	+Inf	1.66081e+06	c_u_buy_from_fac_limit_t(2)_	.	.
102	c_u_discount_limit_3(1_A)_	NU	.	.	-Inf
.	-18.75000	1.66081e+06	R(A_1)	18.75000	.
3.07853e+07		+Inf	5.78885e+08	c_u_discount_limit_4(1)_	.
103	c_u_discount_limit_3(1_B)_	NU	.	.	-Inf
2576.96970	-26.00000	1.59381e+06	R(B_1)	26.00000	.
3.07853e+07		+Inf	8.02079e+08	c_u_discount_limit_4(1)_	.
104	c_u_discount_limit_3(2_A)_	NU	.	.	-Inf
.	-19.49994	1.66081e+06	R(A_2)	19.49994	.
.	+Inf	1.66081e+06	c_u_buy_from_fac_limit_f(2)_	.	.
105	c_u_discount_limit_3(2_B)_	NU	.	.	-Inf
.	-26.99994	1.66081e+06	R(B_2)	26.99994	.
.	+Inf	1.66081e+06	c_u_buy_from_fac_limit_f(2)_	.	.
106	c_u_discount_limit_4(1)_	BS	-3.07853e+07	3.07853e+07	-Inf
9.99997e+08	.	1.66081e+06	c_u_discount_limit_1(1)_	.	.
3.07853e+07	.00005	1.65934e+06	c_u_discount_limit_4(2)_	.	.
107	c_u_discount_limit_4(2)_	NU	.	.	-Inf
.	-.00006	1.66081e+06	c_u_buy_from_fac_limit_f(2)_	.	.

.000006  
 . +Inf 1.66081e+06 c\_u\_buy\_from\_fac\_limit\_t(2)\_

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
 Page 12

Problem:  
 Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
1	g(A_Tehran_Mashhad)	NL	.	520.00000	.
-43.63636	-Inf	1.66081e+06	g(B_Isfahan_Mashhad)	.	+Inf
300.00000	520.00000	1.66081e+06	g(B_Tehran_Mashhad)	.	
2	g(A_Tehran_Kerman)	BS	300.00000	540.00000	.
256.36364	510.00000	1.65181e+06	c_u_max_market_demand_limit(Kerman_A)_	.	+Inf
600.00000	540.00000	1.66081e+06	g(B_Isfahan_Kerman)	.	
3	g(A_Tehran_Ahvaz)	NL	.	490.00000	.
.	-Inf	1.66081e+06	c_u_max_market_demand_limit(Ahvaz_B)_	.	+Inf
56.36364	550.00000	1.65743e+06	c_u_max_market_demand_limit(Mashhad_B)_	-60.00000	
4	g(A_Tehran_Tabriz)	NL	.	500.00000	.
155.15152	-Inf	1.66081e+06	t(B_Main_Isfahan)	.	-
400.00000	500.00000	1.66081e+06	g(A_Isfahan_Tabriz)	.	+Inf
5	g(A_Isfahan_Mashhad)	BS	356.36364	520.00000	.
56.36364	520.00000	1.66081e+06	g(A_Tehran_Mashhad)	.	+Inf
400.00000	550.00000	1.6715e+06	c_u_max_market_demand_limit(Kerman_A)_	.	
6	g(A_Isfahan_Kerman)				



	BS	500.00000	540.00000	.
200.00000	540.00000	1.66081e+06	g(B_Isfahan_Kerman)	.
				+Inf
500.00000	+Inf	+Inf		

7	g(A_Isfahan_Ahvaz)			
	NL	.	490.00000	.
.	-Inf	1.66081e+06		
c_u_max_market_demand_limit(Ahvaz_B)_			-60.00000	+Inf
56.36364	550.00000	1.65743e+06		
c_u_max_market_demand_limit(Mashhad_B)_				

8	g(A_Isfahan_Tabriz)			
	BS	400.00000	500.00000	.
100.00000	500.00000	1.66081e+06	g(A_Tehran_Tabriz)	-
				+Inf
456.36364	530.00000	1.67281e+06		
c_u_max_market_demand_limit(Tabriz_B)_				

9	g(B_Tehran_Mashhad)			
	BS	300.00000	700.00000	.
-56.36364	700.00000	1.66081e+06	g(A_Tehran_Mashhad)	.
				+Inf
300.00000	+Inf	+Inf		

10	g(B_Tehran_Kerman)			
	BS	300.00000	690.00000	.
200.00000	690.00000	1.66081e+06	g(B_Isfahan_Kerman)	-
				+Inf
343.63636	720.00000	1.66981e+06		
c_u_max_market_demand_limit(Kerman_A)_				

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 13

Problem:  
Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		

11	g(B_Tehran_Ahvaz)				
	BS	1000.00000	730.00000	.	
943.63636	670.00000	1.60081e+06	g(A_Tehran_Ahvaz)		
					+Inf
1000.00000	+Inf	+Inf			

12 g(B_Tehran_Tabriz)				
	BS	500.00000	710.00000	.
100.00000	710.00000	1.66081e+06	g(A_Tehran_Tabriz)	.
				+Inf
500.00000	+Inf	+Inf		
13 g(B_Isfahan_Mashhad)				
	BS	43.63636	700.00000	.
111.51515	670.00000	1.6595e+06		-
c_u_max_market_demand_limit(Kerman_A)_				.
				+Inf
343.63636	700.00000	1.66081e+06	g(A_Tehran_Mashhad)	
14 g(B_Isfahan_Kerman)				
	NL	.	690.00000	.
155.15152	-Inf	1.66081e+06	t(B_Main_Isfahan)	-
				.
				+Inf
300.00000	690.00000	1.66081e+06	g(B_Tehran_Kerman)	
15 g(B_Isfahan_Ahvaz)				
	BS	500.00000	730.00000	.
443.63636	670.00000	1.63081e+06	g(A_Isfahan_Ahvaz)	.
				+Inf
500.00000	+Inf	+Inf		
16 g(B_Isfahan_Tabriz)				
	BS	600.00000	710.00000	.
543.63636	680.00000	1.64281e+06		.
c_u_max_market_demand_limit(Tabriz_B)_				.
				+Inf
1000.00000	710.00000	1.66081e+06	g(A_Tehran_Tabriz)	
17 Extracted_ore(1)				
	BS	560.00000	-45.00000	.
471.42857	-254.57253	1.54345e+06	c_u_Max_extracted_ore_limit(1)_	.
				+Inf
560.00000	+Inf	+Inf		
18 Extracted_ore(2)				
	BS	1000.00000	-65.00000	.
896.66667	-390.65010	1.33516e+06	c_u_Max_extracted_ore_limit(2)_	.
				+Inf
1000.00000	+Inf	+Inf		
19 Extracted_ore(3)				
	BS	1440.00000	-70.00000	.
1316.00000	-197.51434	1.47719e+06	c_u_Max_extracted_ore_limit(3)_	.
				+Inf
1440.00000	+Inf	+Inf		

```

20 t(A_1_Tehran)
      NL      .      -375.00000      .
      -Inf      1.66081e+06 R(A_1)
      -42.25000      +Inf
43.63636      -332.75000      1.65897e+06 g(B_Isfahan_Mashhad)

```

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 14

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		

```

-----
21 t(A_1_Isfahan)
      NL      .      -375.00000      .
      -Inf      1.66081e+06 R(A_1)
      -42.25000      +Inf
43.63636      -332.75000      1.65897e+06 g(B_Isfahan_Mashhad)

```

```

22 t(A_2_Tehran)
      NL      .      -390.00000      .
      -Inf      1.66081e+06 R(A_2)
      -37.94000      +Inf
      -352.06000      1.66081e+06 c_u_buy_from_fac_limit_t(2)_

```

```

23 t(A_2_Isfahan)
      NL      .      -390.00000      .
      -Inf      1.66081e+06 R(A_2)
      -37.50000      +Inf
      -352.50000      1.66081e+06 t(B_2_Isfahan)

```

```

24 t(B_1_Tehran)
      BS      1588.48485      -520.00000      .
1433.33333      -520.04000      1.66075e+06
c_u_transportation_limit_t(1_Tehran)_
      .      +Inf
1900.00000      -519.96000      1.66088e+06
c_u_transportation_limit_t(1_Isfahan)_

```

```

25 t(B_1_Isfahan)
      BS      988.48485      -520.00000      .
676.96970      -520.04000      1.66077e+06
c_u_transportation_limit_t(1_Isfahan)_
      .      +Inf

```

```

1143.63636    -519.96000    1.66085e+06
c_u_transportation_limit_t(1_Tehran)_

26 t(B_2_Tehran)
      NL      .      -540.00000      .
      -Inf    1.66081e+06 B(2_Tehran)
      .      -540.44000      +Inf
      -539.56000    1.66081e+06 c_u_buy_from_fac_limit_t(2)_

27 t(B_2_Isfahan)
      BS      .      -540.00000      .
      -540.44000    1.66081e+06 t(B_2_Tehran)
      .      +Inf
      -520.86000    1.66081e+06 c_u_discount_limit_4(2)_

28 h(1)      BS      .52949    -120.00000      .
.52949    -960.00000    1.66037e+06 c_u_container_limit(2_Isfahan)_
      .      1.00000
.58121      .      1.66088e+06
c_u_transportation_limit_t(1_Tehran)_

29 h(2)      BS      .      -90.00000      .
      -750.00000    1.66081e+06 t(B_2_Tehran)
      .      1.00000
      1.66081e+06
c_u_transportation_limit_t(2_Isfahan)_

30 B(1_Tehran) BS      15.88485    -180.00000      .
14.33333    -184.00000    1.66075e+06
c_u_transportation_limit_t(1_Tehran)_
      .      +Inf
19.00000    -176.00000    1.66088e+06
c_u_transportation_limit_t(1_Isfahan)_

```

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 15

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
31	B(1_Isfahan)	BS	15.88485	-210.00000	.
12.76970	-214.00000	1.66075e+06			
c_u_transportation_limit_t(1_Isfahan)_					+Inf

```

17.43636      -206.00000      1.66088e+06
c_u_transportation_limit_t(1_Tehran)_

32 B(2_Tehran)  BS          .      -240.00000      .
.      -Inf      1.66081e+06
.      .      +Inf
.      -196.00000      1.66081e+06 t(B_2_Tehran)

33 B(2_Isfahan) BS          .      -220.00000      .
.      -264.00000      1.66081e+06 t(B_2_Tehran)
.      .      +Inf
.      -206.00000      1.66081e+06 c_u_container_limit(2_Isfahan)_

34 B(Main_Tehran)
BS          8.11515      -200.00000      .
5.00000      -204.00000      1.66078e+06
c_u_transportation_limit_t(1_Isfahan)_
.      .      +Inf
9.66667      -196.00000      1.66085e+06
c_u_transportation_limit_t(1_Tehran)_

35 B(Main_Isfahan)
BS          14.11515      -230.00000      .
12.56364      -234.00000      1.66076e+06
c_u_transportation_limit_t(1_Tehran)_
.      .      +Inf
17.23030      -226.00000      1.66087e+06
c_u_transportation_limit_t(1_Isfahan)_

36 G(Tehran_Mashhad)
BS          3.00000      -110.00000      .
2.23037      -20418.00000      1.59989e+06 l(Tehran_Mashhad)
.      .      +Inf
3.00000      +Inf      +Inf

37 G(Tehran_Kerman)
BS          6.00000      -85.00000      .
5.23037      -19418.00000      1.54482e+06 l(Tehran_Kerman)
.      .      +Inf
6.00000      +Inf      +Inf

38 G(Tehran_Ahvaz)
BS          10.00000      -120.00000      .
9.23037      -23418.00000      1.42783e+06 l(Tehran_Ahvaz)
.      .      +Inf
10.00000      +Inf      +Inf

39 G(Tehran_Tabriz)
BS          5.00000      -100.00000      .
4.56364      -18418.00000      1.56922e+06 l(Tehran_Tabriz)

```

5.00000	+Inf	+Inf	.	+Inf
---------	------	------	---	------

40 G(Isfahan_Mashhad)				
	BS	4.00000	-100.00000	.
3.56364	-20600.00000	1.57881e+06	l(Isfahan_Mashhad)	.
4.00000	+Inf	+Inf	.	+Inf

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 16

Problem:  
Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					

41 G(Isfahan_Kerman)					
	BS	5.00000	-100.00000	.	
4.23037	-19600.00000	1.56331e+06	l(Isfahan_Kerman)	.	
5.00000	+Inf	+Inf	.		+Inf

42 G(Isfahan_Ahvaz)					
	BS	5.00000	-110.00000	.	
4.23037	-23600.00000	1.54336e+06	l(Isfahan_Ahvaz)	.	
5.00000	+Inf	+Inf	.		+Inf

43 G(Isfahan_Tabriz)					
	BS	10.00000	-90.00000	.	
9.56364	-18600.00000	1.47571e+06	l(Isfahan_Tabriz)	.	
10.00000	+Inf	+Inf	.		+Inf

44	R(A_1)	BS	.	18.75000	.
-Inf	.	1.66081e+06	c_u_discount_limit_3(1_A)_	.	
43.63636	61.00000	1.66081e+06	t(A_1_Tehran)	.	+Inf

45	R(A_2)	BS	.	19.50000	.
.	.000006	1.66081e+06	c_u_discount_limit_3(2_A)_	.	
.	57.00000	1.66081e+06	t(A_2_Isfahan)	.	+Inf

46	R(B_1)	BS	2576.96970	26.00000	.
----	--------	----	------------	----------	---

2576.96970	6.86000	1.61149e+06	c_u_discount_limit_4(2)_	.	.
				.	+Inf
2647.11111	251.42714	2.24173e+06	c_u_Max_extracted_ore_limit(3)_	.	.
47 R(B_2)	BS	27.00000	.	.	.
.	.00014	1.66081e+06	c_u_discount_limit_3(2_B)_	.	.
				.	+Inf
.	46.14000	1.66081e+06	c_u_discount_limit_4(2)_	.	.
48 U(A)	BS	1556.36364	.	.	.
1500.00000	-280.53156	1.2242e+06	c_u_Max_extracted_ore_limit(3)_	.	.
				.	+Inf
1556.36364	+Inf	+Inf	.	.	.
49 Z(1_A)	BS	140.00000	.	.	.
436.96970	.	1.66081e+06	c_u_Metal_sum_limit_Z(1)_	.	.
				.	+Inf
140.00000	+Inf	+Inf	.	.	.
50 Z(2_A)	BS	150.00000	.	.	.
16.66667	.	1.66081e+06	Z(2_B)	.	.
				.	+Inf
150.00000	+Inf	+Inf	.	.	.

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 17

Problem:  
Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
51 Z(3_A)	BS	225.69697	.	.	.
153.69697	.	1.66081e+06	C(3_A)	.	.
				.	+Inf
365.69697	.	1.66081e+06	c_u_Metal_sum_limit_Z(1)_	.	.
52 C(1_A)	BS	168.00000	.	.	.
-65.33333	.	1.66081e+06	C(1_B)	.	.
				.	+Inf
168.00000	+Inf	+Inf	.	.	.
53 C(2_A)	BS	16.66667	.	.	.
168.00000	.	1.66081e+06	c_u_Metal_sum_limit_C(2)_	.	.
				.	+Inf
184.66667	.	1.66081e+06	C(1_B)	.	.

54 C(3_A)	NL	.	.	.	-
184.66667	-Inf	1.66081e+06	c_u_Metal_in_alloy_limit_A_C_f_	.	.
				+Inf	
72.00000	.	1.66081e+06	c_u_Metal_sum_limit_C(3)_	.	.
55 A(1_A)	BS	196.00000	.	.	.
165.00000	-570.58182	1.54898e+06	A(1_B)	.	.
				+Inf	
196.00000	+Inf	+Inf	.	.	.
56 A(2_A)	BS	300.00000	.	.	.
269.00000	-570.58182	1.48964e+06	A(2_B)	.	.
				+Inf	
300.00000	+Inf	+Inf	.	.	.
57 A(3_A)	BS	360.00000	.	.	.
329.00000	-510.05737	1.47719e+06	c_u_Max_extracted_ore_limit(3)_	.	.
				+Inf	
360.00000	+Inf	+Inf	.	.	.
58 F(1_A)	NL	.	.	.	.
.	-Inf	1.66081e+06	c_u_Metal_in_alloy_limit_A_F_f_	.	.
			-1097.37778	+Inf	
28.00000	1097.37778	1.63009e+06	F(1_B)	.	.
59 F(2_A)	NL	.	.	.	.
.	-Inf	1.66081e+06	c_u_Metal_in_alloy_limit_A_F_f_	.	.
			-1097.37778	+Inf	
126.95455	1097.37778	1.5215e+06	Z(3_A)	.	.
60 F(3_A)	NL	.	.	.	.
.	-Inf	1.66081e+06	c_u_Metal_in_alloy_limit_A_F_f_	.	.
			-1097.37778	+Inf	
72.00000	1097.37778	1.5818e+06	F(3_B)	.	.

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 18

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting		
range	range	break point	variable	Marginal	Upper bound
61 U(B)	BS	666.66667	.	.	.
604.44444	-493.82000	1.3316e+06	F(1_A)	.	.



666.66667	+Inf	+Inf	.	+Inf
62 Z(1_B)	NL	.	.	-
225.69697	-Inf	1.66081e+06 Z(3_A)	.	
133.33333	.	1.66081e+06 Z(3_B)	.	+Inf
63 Z(2_B)	NL	.	.	-
225.69697	-Inf	1.66081e+06 Z(3_A)	.	
133.33333	.	1.66081e+06 Z(3_B)	.	+Inf
64 Z(3_B)	BS	133.33333	.	.
-6.66667	.	1.66081e+06 Z(1_B)	.	
666.66667	.	1.66081e+06 c_u_Alloy_sum_limit(B)_	.	+Inf
65 C(1_B)	NL	.	.	.
-16.66667	-Inf	1.66081e+06 C(2_A)	.	
168.00000	.	1.66081e+06 C(1_A)	.	+Inf
66 C(2_B)	BS	233.33333	.	.
65.33333	.	1.66081e+06 C(1_B)	.	
250.00000	.	1.66081e+06 c_u_Metal_in_alloy_limit_B_C_f_	.	+Inf
67 C(3_B)	NL	.	.	.
-16.66667	-Inf	1.66081e+06 C(2_A)	.	
72.00000	.	1.66081e+06 c_u_Metal_sum_limit_C(3)_	.	+Inf
68 A(1_B)	NL	.	.	.
.	-Inf	1.66081e+06 c_u_Metal_in_alloy_limit_B_A_f_	.	
31.00000	570.58182	1.64313e+06	-570.58182	+Inf
c_u_max_market_demand_limit(Mashhad_B)_				
69 A(2_B)	NL	.	.	.
.	-Inf	1.66081e+06 c_u_Metal_in_alloy_limit_B_A_f_	.	
31.00000	570.58182	1.64313e+06	-570.58182	+Inf
c_u_max_market_demand_limit(Mashhad_B)_				
70 A(3_B)	NL	.	.	.
.	-Inf	1.66081e+06 c_u_Metal_in_alloy_limit_B_A_f_	.	
31.00000	570.58182	1.64313e+06	-570.58182	+Inf
c_u_max_market_demand_limit(Mashhad_B)_				

# GLPK 5.0 - SENSITIVITY ANALYSIS REPORT

Page 19

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
-----					
71	F(1_B)	BS	28.00000	.	.
-98.95455	-1097.37778	1.63009e+06	F(1_A)	.	+Inf
28.00000	+Inf	+Inf			
72	F(2_B)	BS	200.00000	.	.
73.04545	-1097.37778	1.44134e+06	F(2_A)	.	+Inf
200.00000	+Inf	+Inf			
73	F(3_B)	BS	72.00000	.	.
-54.95455	-1097.37778	1.5818e+06	F(3_A)	.	+Inf
72.00000	+Inf	+Inf			
74	t(A_Main_Tehran)	BS	300.00000	.	.
256.36364	-30.00000	1.65181e+06		.	+Inf
c_u_max_market_demand_limit(Kerman_A)_		1.66081e+06	g(A_Tehran_Mashhad)		
600.00000	.				
75	t(A_Main_Isfahan)	BS	1256.36364	.	.
956.36364	.	1.66081e+06	g(A_Tehran_Mashhad)	.	+Inf
1300.00000	30.00000	1.6985e+06			
c_u_max_market_demand_limit(Kerman_A)_					
76	t(B_Main_Tehran)	BS	511.51515	.	.
211.51515	.	1.66081e+06	g(A_Tehran_Mashhad)	.	+Inf
666.66667	.04000	1.66083e+06			
c_u_transportation_limit_t(1_Tehran)_					
77	t(B_Main_Isfahan)				

```

        BS      155.15152      .      .      -
639.09091      -.04000      1.66081e+06
c_u_transportation_limit_t(1_Tehran)_
        .      +Inf
455.15152      .      1.66081e+06 g(A_Tehran_Mashhad)

78 h(Main)      NU      1.00000      .      .
.70576      .      1.66081e+06
c_u_transportation_limit_t(Main_Isfahan)_
        .      1.00000
1.62303      +Inf      1.66081e+06
c_u_transportation_limit_f(Main_Tehran)_

79 l(Tehran_Mashhad)
        NU      1.00000      .      .
.74346 -60924.00000      1.64518e+06 c_u_discount_limit_4(1)_
        60924.00000      1.00000
1.18788      +Inf      1.67226e+06
c_u_max_market_demand_limit(Mashhad_B)_

80 l(Tehran_Kerman)
        NU      1.00000      .      .
.87173 -115998.00000      1.64593e+06 c_u_discount_limit_4(1)_
        115998.00000      1.00000
1.51717      +Inf      1.7208e+06 t(B_Main_Isfahan)

```

GLPK 5.0 - SENSITIVITY ANALYSIS REPORT  
Page 20

Problem:

Objective: revenue = 1660813.37 (MAXimum)

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting		
range	range	break point	variable	Marginal	Upper bound
-----					
-----					

```

81 l(Tehran_Ahvaz)
        NU      1.00000      .      .
.92304 -232980.00000      1.64288e+06 c_u_discount_limit_4(1)_
        232980.00000      1.00000
1.00000      +Inf      1.66081e+06
c_u_max_market_demand_limit(Ahvaz_B)_

82 l(Tehran_Tabriz)
        NU      1.00000      .      .
.91273 -91590.00000      1.65282e+06 g(B_Isfahan_Mashhad)
        91590.00000      1.00000
1.11273      +Inf      1.67114e+06

```

```

c_u_max_market_demand_limit(Mashhad_B)_
      83 l(Isfahan_Mashhad)
          NU      1.00000      .      .
.89091  -82000.00000      1.65187e+06 g(B_Isfahan_Mashhad)
                                82000.00000      1.00000
1.14091      +Inf      1.67237e+06
c_u_max_market_demand_limit(Mashhad_B)_
      84 l(Isfahan_Kerman)
          NU      1.00000      .      .
.84607  -97500.00000      1.64581e+06 c_u_discount_limit_4(1)_
                                97500.00000      1.00000
1.14101      +Inf      1.67456e+06 c_u_buy_from_fac_limit_t(1)_
      85 l(Isfahan_Ahvaz)
          NU      1.00000      .      .
.84607  -117450.00000      1.64273e+06 c_u_discount_limit_4(1)_
                                117450.00000      1.00000
1.00000      +Inf      1.66081e+06
c_u_max_market_demand_limit(Ahvaz_B)_
      86 l(Isfahan_Tabriz)
          NU      1.00000      .      .
.95636  -185100.00000      1.65274e+06 g(B_Isfahan_Mashhad)
                                185100.00000      1.00000
1.05636      +Inf      1.67125e+06
c_u_max_market_demand_limit(Mashhad_B)_
      87 d(1)      BS      .03079      .      .
.03079  -47849.99991      1.65934e+06 c_u_discount_limit_4(2)_
                                .      1.00000
1.03079      .      1.66081e+06 c_u_discount_limit_1(1)_
      88 d(2)      BS      .      .      .
.      -1.94999e+10      1.66081e+06 c_u_discount_limit_3(2_A)_
                                .      1.00000
.      57419.99989      1.66081e+06 c_u_discount_limit_4(2)_
End of report

```

## Analysis D

here in problem D, we wanna see how much change we should make in cost of converting Ore to Alloy to stop Main fac from producing product:

### Code for Analysis D

the related python code for this problem is shown below, but you should run the model\_runner.py to get the result.

```
def d():
    for cost in range(300, 400):
        instance.Price_of_ore_to_alloy = cost
        solver.solve(instance)
        print(f"Price of ore to alloy:
{instance.Price_of_ore_to_alloy()}, Use Main?: {instance.h['Main']
()}")
```

## Output

run the command below to see the result:

```
!python model_runner.py -d
results for problem: -d
Price of ore to alloy: 300, Use Main?: 1.0
Price of ore to alloy: 301, Use Main?: 1.0
Price of ore to alloy: 302, Use Main?: 1.0
Price of ore to alloy: 303, Use Main?: 1.0
Price of ore to alloy: 304, Use Main?: 1.0
Price of ore to alloy: 305, Use Main?: 1.0
Price of ore to alloy: 306, Use Main?: 1.0
Price of ore to alloy: 307, Use Main?: 1.0
Price of ore to alloy: 308, Use Main?: 1.0
Price of ore to alloy: 309, Use Main?: 1.0
Price of ore to alloy: 310, Use Main?: 1.0
Price of ore to alloy: 311, Use Main?: 1.0
Price of ore to alloy: 312, Use Main?: 1.0
Price of ore to alloy: 313, Use Main?: 1.0
Price of ore to alloy: 314, Use Main?: 1.0
Price of ore to alloy: 315, Use Main?: 1.0
Price of ore to alloy: 316, Use Main?: 1.0
Price of ore to alloy: 317, Use Main?: 1.0
Price of ore to alloy: 318, Use Main?: 1.0
Price of ore to alloy: 319, Use Main?: 1.0
Price of ore to alloy: 320, Use Main?: 1.0
Price of ore to alloy: 321, Use Main?: 1.0
Price of ore to alloy: 322, Use Main?: 1.0
Price of ore to alloy: 323, Use Main?: 1.0
Price of ore to alloy: 324, Use Main?: 1.0
Price of ore to alloy: 325, Use Main?: 1.0
Price of ore to alloy: 326, Use Main?: 1.0
Price of ore to alloy: 327, Use Main?: 1.0
Price of ore to alloy: 328, Use Main?: 1.0
Price of ore to alloy: 329, Use Main?: 1.0
Price of ore to alloy: 330, Use Main?: 1.0
Price of ore to alloy: 331, Use Main?: 1.0
Price of ore to alloy: 332, Use Main?: 1.0
Price of ore to alloy: 333, Use Main?: 1.0
```

[illegible]

```
Price of ore to alloy: 383, Use Main?: 0.0
Price of ore to alloy: 384, Use Main?: 0.0
Price of ore to alloy: 385, Use Main?: 0.0
Price of ore to alloy: 386, Use Main?: 0.0
Price of ore to alloy: 387, Use Main?: 0.0
Price of ore to alloy: 388, Use Main?: 0.0
Price of ore to alloy: 389, Use Main?: 0.0
Price of ore to alloy: 390, Use Main?: 0.0
Price of ore to alloy: 391, Use Main?: 0.0
Price of ore to alloy: 392, Use Main?: 0.0
Price of ore to alloy: 393, Use Main?: 0.0
Price of ore to alloy: 394, Use Main?: 0.0
Price of ore to alloy: 395, Use Main?: 0.0
Price of ore to alloy: 396, Use Main?: 0.0
Price of ore to alloy: 397, Use Main?: 0.0
Price of ore to alloy: 398, Use Main?: 0.0
Price of ore to alloy: 399, Use Main?: 0.0
```

Problem:

- Name: unknown
- Lower bound: 980595.0
- Upper bound: 980595.0
- Number of objectives: 1
- Number of constraints: 107
- Number of variables: 88
- Number of nonzeros: 342
- Sense: maximize

Solver:

- Status: ok
- Termination condition: optimal
- Statistics:
  - Branch and bound:
    - Number of bounded subproblems: 3
    - Number of created subproblems: 3
  - Error rc: 0
  - Time: 0.007172584533691406

Solution:

- number of solutions: 0
- number of solutions displayed: 0

no results saved.

## Explanation

As we see from the output, after cost of 352 the main factory will not produce any Alloy.

## Analysis E

Here we wanna see how to change contract cost to see when a factory becomes unworthy to have contract with,

first we should be aware that for **Fac2**, we won't have contract in any cost, so we do analysis just for **Fac1**:

### Code for Analysis E

the related python code for this problem is shown below, but you should run the `model_runner.py` to get the result.

```
def e():
    instance.Price_of_ore_to_alloy = 0
    for cost in range(400, 550):
        instance.contract_cost[1] = 100 * cost
        solver.solve(instance)
        print(f"Fac1 contract cost: {instance.contract_cost[1]()},
Buy from Fac1: {instance.h[1]()}, Buy from Fac2: {instance.h[2]()}")
```

### Output

run the command below to see the result:

```
!python model_runner.py -e

results for problem: -e
Fac1 contract cost: 110000, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110100, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110200, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110300, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110400, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110500, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110600, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110700, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110800, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 110900, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111000, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111100, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111200, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111300, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111400, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111500, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111600, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111700, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111800, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 111900, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112000, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112100, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112200, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112300, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112400, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112500, Buy from Fac1: 1.0, Buy from Fac2: 0.0
Fac1 contract cost: 112600, Buy from Fac1: 1.0, Buy from Fac2: 0.0
```



[illegible]

[illegible]

[illegible]

```

Fac1 contract cost: 127400, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 127500, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 127600, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 127700, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 127800, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 127900, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128000, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128100, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128200, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128300, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128400, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128500, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128600, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128700, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128800, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 128900, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129000, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129100, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129200, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129300, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129400, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129500, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129600, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129700, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129800, Buy from Fac1: 0.0, Buy from Fac2: 1.0
Fac1 contract cost: 129900, Buy from Fac1: 0.0, Buy from Fac2: 1.0
no results saved.

```

## Explanation

As we see, in contract cost of 119600 we won't have contract with Fac1 and instead we will buy from Fac2,

if we analyzed this for Fac2, 'Buy from Fac2' was always 0, that is because costs of buying and sending from Fac2 are higher.

## Analysis F

### Output

The part related to problem is extracted from sensit.sen:

No.	Column name	St	Activity	Obj coef	Lower bound
Activity	Obj coef	Obj value at	Limiting	Marginal	Upper bound
range	range	break point	variable		
-----	-----	--	-----	-----	-----
-----	-----	-----	-----	-----	-----

```

22 t(A_1_Tehran)
      NL      .      -375.00000      .
      -Inf    1.10712e+06 R(A_1)
      -12.25000      +Inf
      -362.75000    1.10712e+06 t(A_Main_Tehran)

23 t(A_1_Isfahan)
      NL      .      -375.00000      .
      -Inf    1.10712e+06 R(A_1)
      -12.25000      +Inf
298.18182    -362.75000    1.10347e+06 g(B_Isfahan_Kerman)

24 t(A_2_Tehran)
      NL      .      -390.00000      .
      -Inf    1.10712e+06 R(A_2)
      -7.94000      +Inf
      -382.06000    1.10712e+06 c_u_buy_from_fac_limit_t(2)_

25 t(A_2_Isfahan)
      NL      .      -390.00000      .
      -Inf    1.10712e+06 R(A_2)
      -7.50000      +Inf
      -382.50000    1.10712e+06 t(B_2_Isfahan)

26 t(B_1_Tehran)
      BS      2034.90909    -520.00000      .
1956.60606    -520.00000    1.10712e+06
c_u_transportation_limit_t(1_Tehran)_
      .      +Inf
2118.30303    -519.96000    1.10721e+06 c_u_buy_from_fac_limit_t(1)_

27 t(B_1_Isfahan)
      BS      1356.60606    -520.00000      .
1273.21212    -520.04000    1.10707e+06 c_u_buy_from_fac_limit_t(1)_
      .      +Inf
1434.90909    -520.00000    1.10712e+06
c_u_transportation_limit_t(1_Tehran)_

28 t(B_2_Tehran)
      NL      .      -540.00000      .
      -Inf    1.10712e+06 c_u_buy_from_fac_limit_f(2)_
      -.44000      +Inf
      -539.56000    1.10712e+06 c_u_buy_from_fac_limit_t(2)_

29 t(B_2_Isfahan)
      BS      .      -540.00000      .
      -540.44000    1.10712e+06 t(B_2_Tehran)
      .      +Inf
      -520.86400    1.10712e+06 c_u_discount_limit_4(2)_

```

## Explanation

To make it clear, information showed above is related to variable  $t$ , which shows how much alloy we send to which depots in what amount (value is showed in **Activity** column).

The **NL** shows that variable is noneBasic, and **BS** shows that variable is Basic.

The **Obj coef** column shows the coefficient of that variable, it is obvious that for  $t(A, F, D)$  it is equal to  $priceOf Alloy Fac(A, F)$ , the **Obj coef Range** shows that in what coef range the **BS** does not change, we will use this to determine the range of coefficients:

to buy **A** from **Fac1**: the coef range is in range (362.750 , inf) and also (362.750 , inf), so it is in range (362.750 , inf)

to buy **A** from **Fac2**: the coef range is in range (382.060 , inf) and also (382.50 , inf), so it is in range (382.50 , inf)

to buy **B** from **Fac1**: the coef range is in range (519.96 , 520.0) and also (520.0 , 520.4), so it is in range (520.0 , 520.0)

to buy **B** from **Fac2**: the coef range is in range (539.56 , inf) and also (520.864 , 540.44), so it is in range (539.56 , 540.44)

## Analysis G

Here we wanna see how the revenue changes by changing a coefficient and a right-hand side.

we have two parts, first we analyze the problem considering values of coefficients change separately, then we analyze the problem considering values of coefficients can change together.

in first part, we choose **depots\_min\_to\_receive** for **Tehran** as right-hand side value and **Container\_cost\_to\_be\_sent\_depot** for **MainFac** to **Tehran** as our coefficient.

in second part, For right-hand side, we chose **Max\_ore** of **Fac2** and for constraint coefficient we chose **price\_of\_alloy\_fac** for Alloy B in **Fac2** and plot in 3D.

## Code for Analysis G

the related python code for this problem is shown below, but you should run the `model_runner.py` to get the result.

```
# first part right-hand side.
def g_2():
    depot_Tehran_min_to_recieve_change_list = []
    revenue_depend_on_tehran_min_to_recieve = []

    for capacity in range(20,65):
        instance.depots_min_to_receive['Tehran'] = capacity
        solver.solve(instance)
        revenue = instance.revenue()
        print(f"Tehran minimum receive: {capacity}, Revenue
```

```

{instance.revenue()}, buy from Fac2: {instance.h[2]()})"
    revenue_depend_on_tehran_min_to_recieve.append(revenue)
    depot_Tehran_min_to_recieve_change_list.append(capacity)
    depot_Tehran_min_to_recieve_change_np =
np.array(depot_Tehran_min_to_recieve_change_list)
    revenue_depend_on_tehran_max_to_recieve_np =
np.array(revenue_depend_on_tehran_min_to_recieve)

    plt.xlabel("Tehran minimum receive")
    plt.ylabel("Revenue")
    plt.plot(depot_Tehran_min_to_recieve_change_np,
revenue_depend_on_tehran_max_to_recieve_np)
    plt.show()

#first part coefficient.
def g_3():
    Container_cost_to_be_sent_depot_list=[]
    revenue_changing_Container_cost_to_be_sent_depot_list = []

    for cost in range (100):
        new_cost = 50*cost
        instance.Container_cost_to_be_sent_depot['Main', 'Tehran'] =
new_cost
        solver.solve(instance)
        revenue = instance.revenue()
        print(f"Container cost from Main to Tehran: {new_cost},
Revenue{instance.revenue()}")
        Container_cost_to_be_sent_depot_list.append(new_cost)

    revenue_changing_Container_cost_to_be_sent_depot_list.append(revenue)

    Container_cost_to_be_sent_depot_np =
np.array(Container_cost_to_be_sent_depot_list)
    revenue_changing_Container_cost_to_be_sent_depot_np =
np.array(revenue_changing_Container_cost_to_be_sent_depot_list)

    plt.plot(Container_cost_to_be_sent_depot_np,
revenue_changing_Container_cost_to_be_sent_depot_np)

    plt.xlabel("Container cost Main->Tehran")
    plt.ylabel("Revenue")
    plt.show()

#second part, both coefficient and right-hand side.
def g():
    price_of_alloy_fac_2_alloy_b_set = np.arange(0,
(instance.price_of_alloy_fac[2,'B']())*3,1)#for c
    max_ore_2_set = np.arange(0,(instance.Max_ore[2]()),100)#for b
    object =

```

```

np.zeros(shape=(len(price_of_alloy_fac_2_alloy_b_set),len(max_ore_2_set)))
    for i,V1 in enumerate(price_of_alloy_fac_2_alloy_b_set):
        for j,V2 in enumerate(max_ore_2_set):
            instance.price_of_alloy_fac[2,'B'] = V1
            instance.Max_ore[2] = V2
            solver.solve(instance)
            object[i,j] = instance.revenue()

    fig = go.Figure(data=[go.Surface(z=object, x=max_ore_2_set,
y=price_of_alloy_fac_2_alloy_b_set)])
    fig.update_layout(title='3D Surface Plot', autosize=True,
                        scene=dict(
                            xaxis_title='maximum of extractable ore number
2',
                            yaxis_title='price of alloy b from factory 2',
                            zaxis_title='Object'))

    fig.show()
    plt.show()#?

```

## Output

run the command below to see the result:

```

%matplotlib inline
%run model_runner.py -g

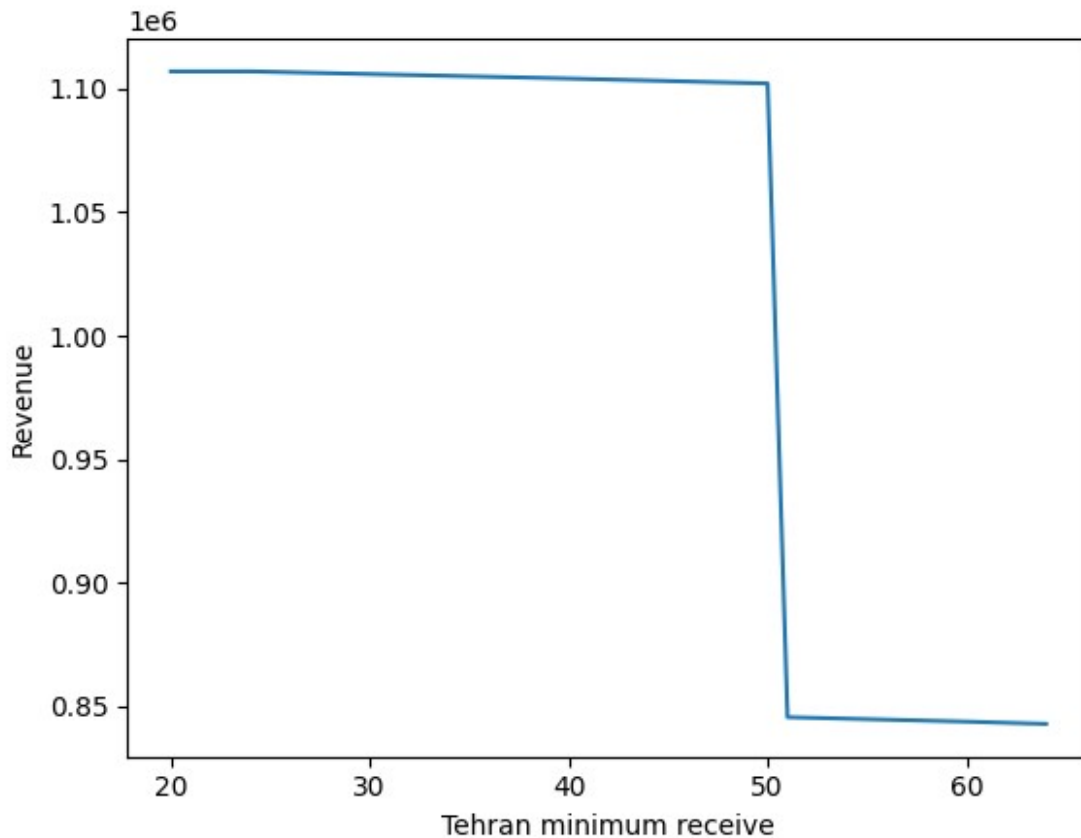
results for problem: -g
Tehran minimum receive: 20, Revenue 1107067.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 21, Revenue 1107067.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 22, Revenue 1107067.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 23, Revenue 1107067.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 24, Revenue 1107067.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 25, Revenue 1106887.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 26, Revenue 1106707.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 27, Revenue 1106527.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 28, Revenue 1106347.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 29, Revenue 1106167.5757575764, buy from Fac2:
0.0
Tehran minimum receive: 30, Revenue 1105987.5757575764, buy from Fac2:

```



0.0  
Tehran minimum receive: 31, Revenue 1105807.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 32, Revenue 1105627.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 33, Revenue 1105447.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 34, Revenue 1105267.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 35, Revenue 1105087.5757575766, buy from Fac2:  
0.0  
Tehran minimum receive: 36, Revenue 1104907.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 37, Revenue 1104727.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 38, Revenue 1104547.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 39, Revenue 1104367.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 40, Revenue 1104187.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 41, Revenue 1103987.5757575766, buy from Fac2:  
0.0  
Tehran minimum receive: 42, Revenue 1103787.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 43, Revenue 1103587.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 44, Revenue 1103387.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 45, Revenue 1103187.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 46, Revenue 1102987.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 47, Revenue 1102787.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 48, Revenue 1102587.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 49, Revenue 1102387.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 50, Revenue 1102187.5757575764, buy from Fac2:  
0.0  
Tehran minimum receive: 51, Revenue 845475.0, buy from Fac2: 1.0  
Tehran minimum receive: 52, Revenue 845275.0, buy from Fac2: 1.0  
Tehran minimum receive: 53, Revenue 845075.0, buy from Fac2: 1.0  
Tehran minimum receive: 54, Revenue 844875.0, buy from Fac2: 1.0  
Tehran minimum receive: 55, Revenue 844675.0, buy from Fac2: 1.0  
Tehran minimum receive: 56, Revenue 844475.0, buy from Fac2: 1.0  
Tehran minimum receive: 57, Revenue 844275.0, buy from Fac2: 1.0  
Tehran minimum receive: 58, Revenue 844075.0, buy from Fac2: 1.0

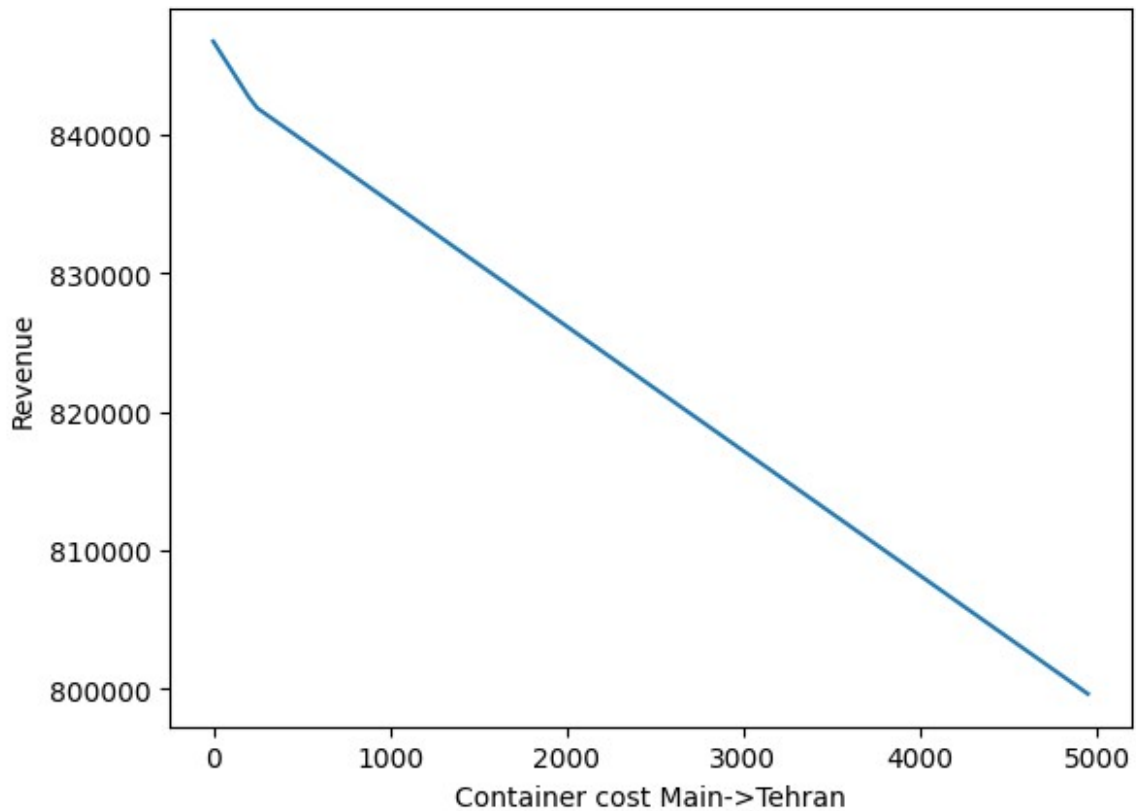
Tehran minimum receive: 59, Revenue 843875.0, buy from Fac2: 1.0  
Tehran minimum receive: 60, Revenue 843675.0, buy from Fac2: 1.0  
Tehran minimum receive: 61, Revenue 843445.0, buy from Fac2: 1.0  
Tehran minimum receive: 62, Revenue 843215.0, buy from Fac2: 1.0  
Tehran minimum receive: 63, Revenue 842985.0, buy from Fac2: 1.0  
Tehran minimum receive: 64, Revenue 842755.0, buy from Fac2: 1.0



Container cost from Main to Tehran: 0, Revenue846755.0  
Container cost from Main to Tehran: 50, Revenue845755.0  
Container cost from Main to Tehran: 100, Revenue844755.0  
Container cost from Main to Tehran: 150, Revenue843755.0  
Container cost from Main to Tehran: 200, Revenue842755.0  
Container cost from Main to Tehran: 250, Revenue841925.0  
Container cost from Main to Tehran: 300, Revenue841475.0  
Container cost from Main to Tehran: 350, Revenue841025.0  
Container cost from Main to Tehran: 400, Revenue840575.0  
Container cost from Main to Tehran: 450, Revenue840125.0  
Container cost from Main to Tehran: 500, Revenue839675.0  
Container cost from Main to Tehran: 550, Revenue839225.0  
Container cost from Main to Tehran: 600, Revenue838775.0  
Container cost from Main to Tehran: 650, Revenue838325.0  
Container cost from Main to Tehran: 700, Revenue837875.0  
Container cost from Main to Tehran: 750, Revenue837425.0

Container cost from Main to Tehran:	800,	Revenue836975.0
Container cost from Main to Tehran:	850,	Revenue836525.0
Container cost from Main to Tehran:	900,	Revenue836075.0
Container cost from Main to Tehran:	950,	Revenue835625.0
Container cost from Main to Tehran:	1000,	Revenue835175.0
Container cost from Main to Tehran:	1050,	Revenue834725.0
Container cost from Main to Tehran:	1100,	Revenue834275.0
Container cost from Main to Tehran:	1150,	Revenue833825.0
Container cost from Main to Tehran:	1200,	Revenue833375.0
Container cost from Main to Tehran:	1250,	Revenue832925.0
Container cost from Main to Tehran:	1300,	Revenue832475.0
Container cost from Main to Tehran:	1350,	Revenue832025.0
Container cost from Main to Tehran:	1400,	Revenue831575.0
Container cost from Main to Tehran:	1450,	Revenue831125.0
Container cost from Main to Tehran:	1500,	Revenue830675.0
Container cost from Main to Tehran:	1550,	Revenue830225.0
Container cost from Main to Tehran:	1600,	Revenue829775.0
Container cost from Main to Tehran:	1650,	Revenue829325.0
Container cost from Main to Tehran:	1700,	Revenue828875.0
Container cost from Main to Tehran:	1750,	Revenue828425.0
Container cost from Main to Tehran:	1800,	Revenue827975.0
Container cost from Main to Tehran:	1850,	Revenue827525.0
Container cost from Main to Tehran:	1900,	Revenue827075.0
Container cost from Main to Tehran:	1950,	Revenue826625.0
Container cost from Main to Tehran:	2000,	Revenue826175.0
Container cost from Main to Tehran:	2050,	Revenue825725.0
Container cost from Main to Tehran:	2100,	Revenue825275.0
Container cost from Main to Tehran:	2150,	Revenue824825.0
Container cost from Main to Tehran:	2200,	Revenue824375.0
Container cost from Main to Tehran:	2250,	Revenue823925.0
Container cost from Main to Tehran:	2300,	Revenue823475.0
Container cost from Main to Tehran:	2350,	Revenue823025.0
Container cost from Main to Tehran:	2400,	Revenue822575.0
Container cost from Main to Tehran:	2450,	Revenue822125.0
Container cost from Main to Tehran:	2500,	Revenue821675.0
Container cost from Main to Tehran:	2550,	Revenue821225.0
Container cost from Main to Tehran:	2600,	Revenue820775.0
Container cost from Main to Tehran:	2650,	Revenue820325.0
Container cost from Main to Tehran:	2700,	Revenue819875.0
Container cost from Main to Tehran:	2750,	Revenue819425.0
Container cost from Main to Tehran:	2800,	Revenue818975.0
Container cost from Main to Tehran:	2850,	Revenue818525.0
Container cost from Main to Tehran:	2900,	Revenue818075.0
Container cost from Main to Tehran:	2950,	Revenue817625.0
Container cost from Main to Tehran:	3000,	Revenue817175.0
Container cost from Main to Tehran:	3050,	Revenue816725.0
Container cost from Main to Tehran:	3100,	Revenue816275.0
Container cost from Main to Tehran:	3150,	Revenue815825.0
Container cost from Main to Tehran:	3200,	Revenue815375.0

Container cost from Main to Tehran:	3250,	Revenue814925.0
Container cost from Main to Tehran:	3300,	Revenue814475.0
Container cost from Main to Tehran:	3350,	Revenue814025.0
Container cost from Main to Tehran:	3400,	Revenue813575.0
Container cost from Main to Tehran:	3450,	Revenue813125.0
Container cost from Main to Tehran:	3500,	Revenue812675.0
Container cost from Main to Tehran:	3550,	Revenue812225.0
Container cost from Main to Tehran:	3600,	Revenue811775.0
Container cost from Main to Tehran:	3650,	Revenue811325.0
Container cost from Main to Tehran:	3700,	Revenue810875.0
Container cost from Main to Tehran:	3750,	Revenue810425.0
Container cost from Main to Tehran:	3800,	Revenue809975.0
Container cost from Main to Tehran:	3850,	Revenue809525.0
Container cost from Main to Tehran:	3900,	Revenue809075.0
Container cost from Main to Tehran:	3950,	Revenue808625.0
Container cost from Main to Tehran:	4000,	Revenue808175.0
Container cost from Main to Tehran:	4050,	Revenue807725.0
Container cost from Main to Tehran:	4100,	Revenue807275.0
Container cost from Main to Tehran:	4150,	Revenue806825.0
Container cost from Main to Tehran:	4200,	Revenue806375.0
Container cost from Main to Tehran:	4250,	Revenue805925.0
Container cost from Main to Tehran:	4300,	Revenue805475.0
Container cost from Main to Tehran:	4350,	Revenue805025.0
Container cost from Main to Tehran:	4400,	Revenue804575.0
Container cost from Main to Tehran:	4450,	Revenue804125.0
Container cost from Main to Tehran:	4500,	Revenue803675.0
Container cost from Main to Tehran:	4550,	Revenue803225.0
Container cost from Main to Tehran:	4600,	Revenue802775.0
Container cost from Main to Tehran:	4650,	Revenue802325.0
Container cost from Main to Tehran:	4700,	Revenue801875.0
Container cost from Main to Tehran:	4750,	Revenue801425.0
Container cost from Main to Tehran:	4800,	Revenue800975.0
Container cost from Main to Tehran:	4850,	Revenue800525.0
Container cost from Main to Tehran:	4900,	Revenue800075.0
Container cost from Main to Tehran:	4950,	Revenue799625.0



no results saved.

<Figure size 640x480 with 0 Axes>

the result for g() will be a html output in your browser, because the code takes long to generate output page, some images of output are shown below.

.

.

.

.

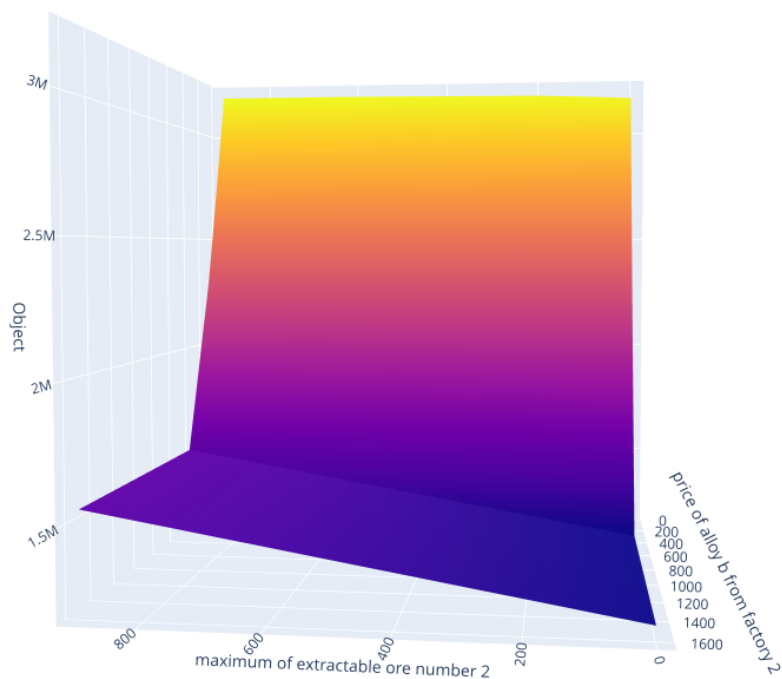
.

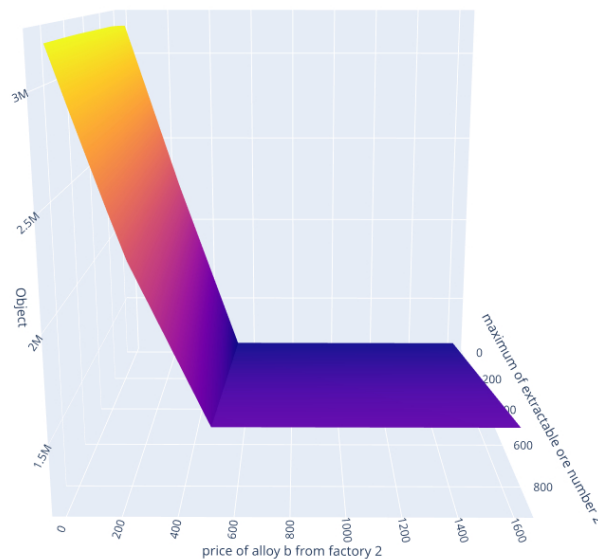
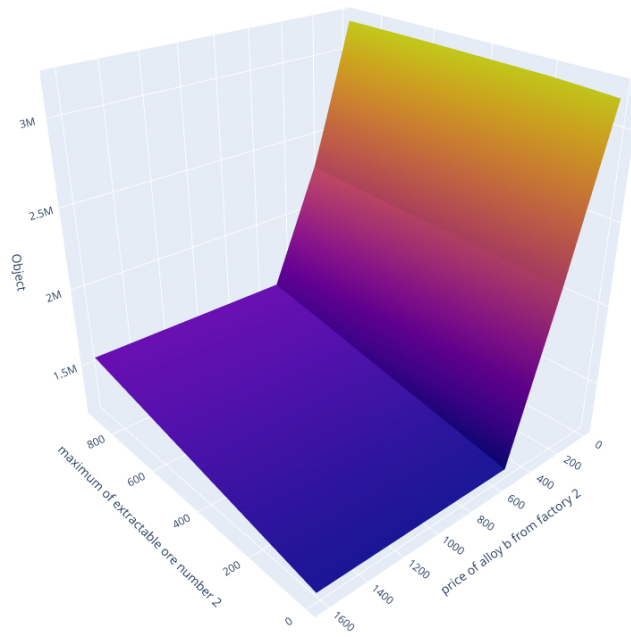
.

.

.

.





## Explanation

In first output, we have revenue based on `Depot_min_to_receive` of `Tehran`, as we see, the revenue is constantly reducing, and after a point when `Fac2` enters the equation (minimum receive of 51), we will have a huge amount of reduce in revenue, which is because of having a floor of buying Alloy from `Fac2`.

In second output, we have revenue based on cost of sending container from `MainFac` to `Tehran`. as we see and as expected, as the container cost increases, the revenue decreases constantly.

In third output we see that as the `price_of_alloy_fac` for Alloy B in `Fac2` increases, the revenue decreases and as the `Max_ore` of `Fac2` increases, the revenue increases, which is expected.

## Analysis H

we have three parts, first analyze for cost be 10 percent higher and lower.

and in second and third part we have costs range between 10 percent lower and 10 percent higher, for this parts, we chose `Container_cost_to_be_sent_depot` of `Fac1` to `Isfahan` and `Container_cost_to_be_sent_market` of `Isfahan` -> `Mashhad` as variables, and we analyze them separately.

### Code for Analysis H

the related python code for this problem is shown below, but you should run the `model_runner.py` to get the result.

```
def h():
    # Increase all prices
    solver.solve(instance)
    previous_revenue = instance.revenue()

    factories_list = instance.Factories()
    depots_list = instance.Depots()
    Markets_list = instance.Markets()

    for factory in factories_list:
        for depot in depots_list:
            instance.Container_cost_to_be_sent_depot[factory, depot] =
110 / 100 * instance.Container_cost_to_be_sent_depot[factory, depot]()

    for depot in depots_list:
        for market in Markets_list:
            instance.Container_cost_to_be_sent_market[depot, market] =
110 / 100 * instance.Container_cost_to_be_sent_market[depot, market]()

    solver.solve(instance)
    new_revenue = instance.revenue()
    print(f'For a 10% increase in costs, previous revenue is
{previous_revenue} and new revenue is {new_revenue}')
```

```
    # Decrease all prices
    instance = model.create_instance(data=data)
    solver.solve(instance)
    previous_revenue = instance.revenue()

    for factory in factories_list:
        for depot in depots_list:
            instance.Container_cost_to_be_sent_depot[factory, depot] =
(90 / 100) * instance.Container_cost_to_be_sent_depot[factory, depot]
()

    for depot in depots_list:
        for market in Markets_list:
```

```

        instance.Container_cost_to_be_sent_market[depot, market] =
(90 / 100) * instance.Container_cost_to_be_sent_market[depot, market]
()

solver.solve(instance)
new_revenue = instance.revenue()
print(f'For a 10% reduction in costs, previous revenue is
{previous_revenue} and new revenue is {new_revenue}')
```

def h\_1():

*# Single price analysis: shipping from factory 1 to depot Isfahan*

instance = model.create\_instance(data=data)

Container\_cost\_to\_be\_sent\_1\_to\_Isfahan = []

revenue\_changing\_Container\_cost\_to\_be\_sent\_1\_to\_Isfahan = []

the\_least\_cost = instance.Container\_cost\_to\_be\_sent\_depot[1, 'Isfahan']() \* (90 / 100)

cost\_step = instance.Container\_cost\_to\_be\_sent\_depot[1, 'Isfahan']() / 100

for cost in range(21):

    new\_cost = cost \* cost\_step + the\_least\_cost

    instance.Container\_cost\_to\_be\_sent\_depot[1, 'Isfahan'] =

new\_cost

    solver.solve(instance)

    revenue = instance.revenue()

    Container\_cost\_to\_be\_sent\_1\_to\_Isfahan.append(new\_cost)

revenue\_changing\_Container\_cost\_to\_be\_sent\_1\_to\_Isfahan.append(revenue)

Container\_cost\_to\_be\_sent\_1\_to\_Isfahan\_np =

np.array(Container\_cost\_to\_be\_sent\_1\_to\_Isfahan)

revenue\_changing\_Container\_cost\_to\_be\_sent\_1\_to\_isfahan\_np =

np.array(revenue\_changing\_Container\_cost\_to\_be\_sent\_1\_to\_Isfahan)

plt.plot(Container\_cost\_to\_be\_sent\_1\_to\_Isfahan\_np,

revenue\_changing\_Container\_cost\_to\_be\_sent\_1\_to\_isfahan\_np)

plt.show()

def h\_2():

*# Single price analysis: shipping from depot Isfahan to Mashhad*

instance = model.create\_instance(data=data)

Container\_cost\_to\_be\_sent\_isfahan\_to\_mashhad = []

revenue\_changing\_Container\_cost\_to\_be\_sent\_isfahan\_to\_mashhad = []

the\_least\_cost =

instance.Container\_cost\_to\_be\_sent\_market['Isfahan', 'Mashhad']() \* (90 / 100)

cost\_step = instance.Container\_cost\_to\_be\_sent\_market['Isfahan',



```

'Mashhad']() / 100

    for cost in range(21):
        new_cost = cost * cost_step + the_least_cost
        instance.Container_cost_to_be_sent_market['Isfahan',
'Mashhad'] = new_cost
        solver.solve(instance)
        revenue = instance.revenue()
        Container_cost_to_be_sent_isfahan_to_mashhad.append(new_cost)

revenue_changing_Container_cost_to_be_sent_isfahan_to_mashhad.append(r
evenue)

    Container_cost_to_be_sent_Isfahan_to_mashhad_np =
np.array(Container_cost_to_be_sent_isfahan_to_mashhad)
    revenue_changing_Container_cost_to_be_sent_isfahan_to_mashhad_np =
np.array(revenue_changing_Container_cost_to_be_sent_isfahan_to_mashhad
)

    plt.plot(Container_cost_to_be_sent_Isfahan_to_mashhad_np,
revenue_changing_Container_cost_to_be_sent_isfahan_to_mashhad_np)
    plt.show()

```

## Output

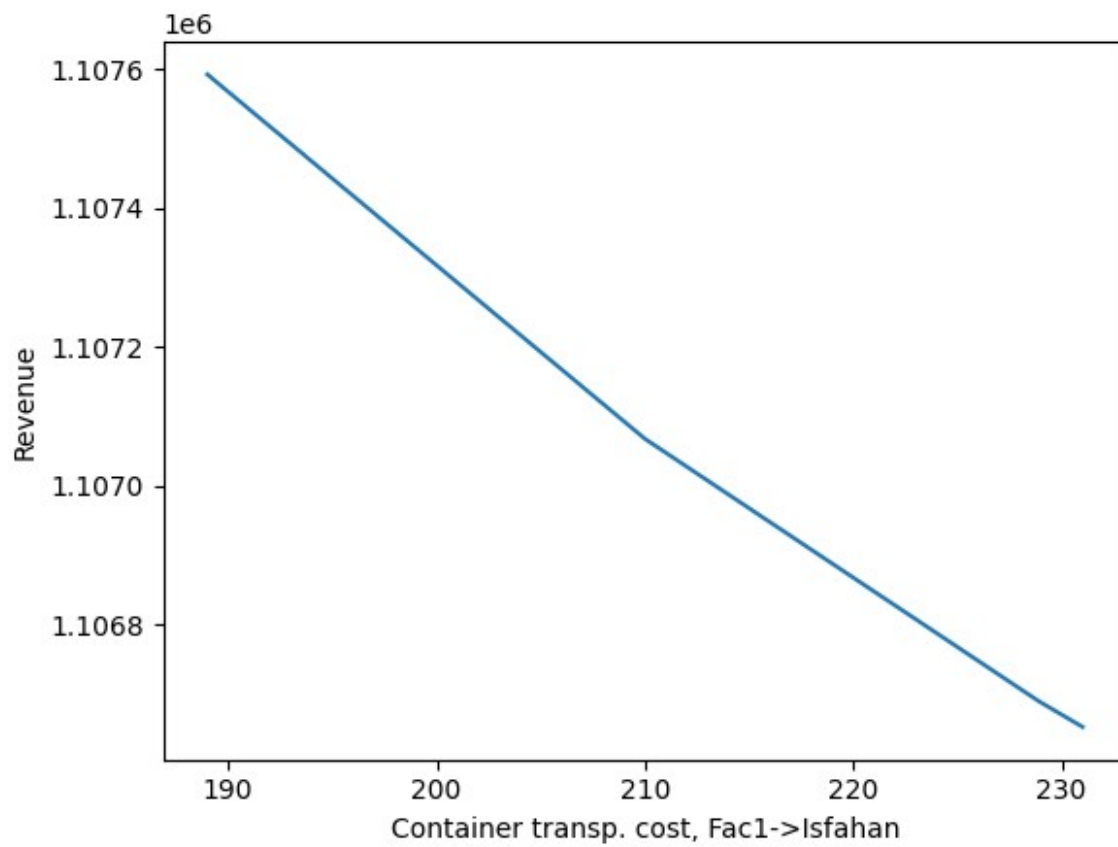
run the command below to see the result:

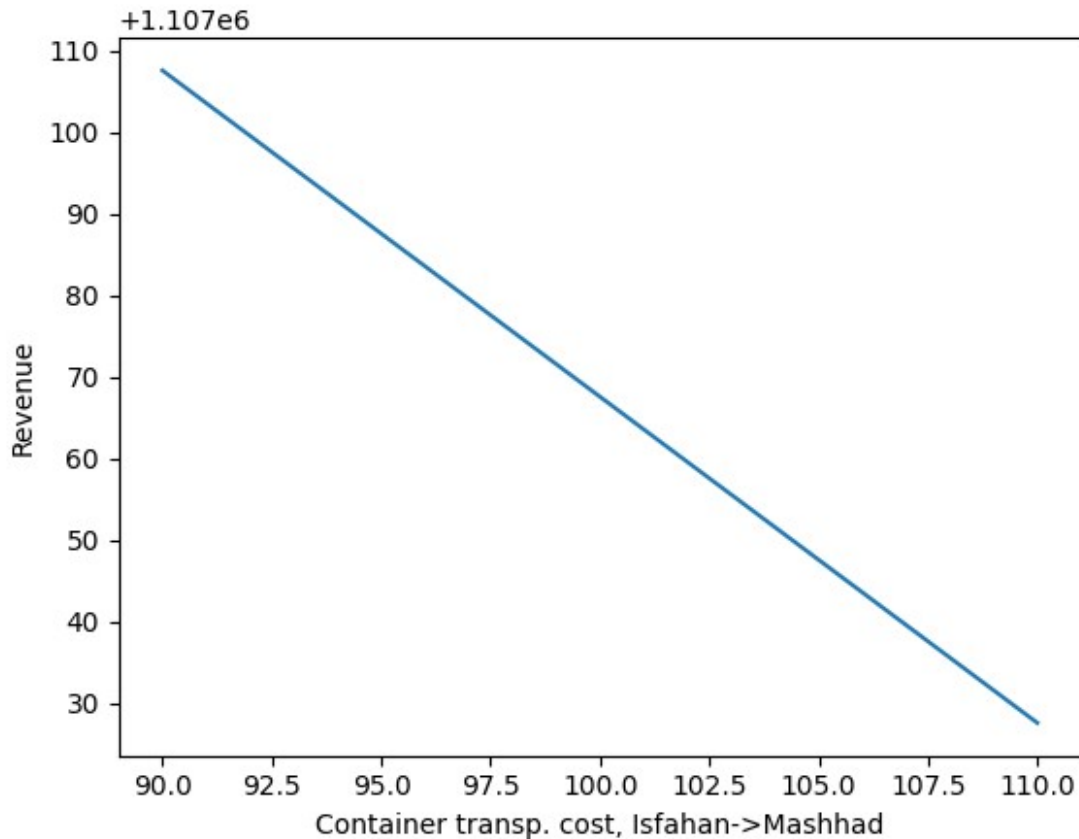
```

%matplotlib inline
%run model_runner.py -h

results for problem: -h
1107067.5757575764
For a 10% increase in costs, previous revenue is 1107067.5757575764
and new revenue is 1105486.5757575764
For a 10% reduction in costs, previous revenue is 1107067.5757575764
and new revenue is 1108648.5757575764

```





no results saved.

<Figure size 640x480 with 0 Axes>

### Explanation

For first part, we see that in case of cost being increased 10 percent, the revenue goes from 1107067.5757575764 to 1105486.5757575764, that shows we have 1581 **less** revenue in this case.

if we decrease costs for 10 percent, the revenue goes from 1107067.5757575764 to 1108648.5757575764, that shows we have made 1581 **more** revenue in this case.

For second part, in first graph we see that by increasing `Container_cost_to_be_sent_depot[1, 'Isfahan']` in the desired range, the revenue **decreases**, which is expected,

and in second graph, by increasing `Container_cost_to_be_sent_market['Isfahan', 'Mashhad']` in the desired range, the revenue **decreases**, which is expected.

### Analysis I

here we wanna see after adding **Abadan** as a new **Market**, in what prices for **Alloys A** and **B**, it is worthy to sell Alloy in the market.

## Code for Analysis I

the related python code for this problem is shown below, but you should run the model\_runner.py to get the result.

```
def i():
    print(list(instance.Markets))
    for p in range(300,500,5):
        instance.sell_prices_Abadan['A'] = p
        solver.solve(instance)
        print(f"Price of A: {p}, Sell A?
{sum(instance.Abadan_Alloys[i,'A']() for i in instance.Depots)})")
    instance.sell_prices_Abadan['A'] = 0
    for p in range(300,500,5):
        instance.sell_prices_Abadan['B'] = p
        solver.solve(instance)
        print(f"Price of B: {p}, Sell B?
{sum(instance.Abadan_Alloys[i,'B']() for i in instance.Depots)})")
```

## Output

run the command below to see the result:

```
!python model_runner.py -i
results for problem: -i
for A -----
Price of A: 300, Total sell A? 0.0
Price of A: 305, Total sell A? 0.0
Price of A: 310, Total sell A? 0.0
Price of A: 315, Total sell A? 0.0
Price of A: 320, Total sell A? 0.0
Price of A: 325, Total sell A? 0.0
Price of A: 330, Total sell A? 0.0
Price of A: 335, Total sell A? 0.0
Price of A: 340, Total sell A? 0.0
Price of A: 345, Total sell A? 0.0
Price of A: 350, Total sell A? 800.0
Price of A: 355, Total sell A? 837.857142857143
Price of A: 360, Total sell A? 2400.0
Price of A: 365, Total sell A? 2423.030303029303
Price of A: 370, Total sell A? 2423.030303029303
Price of A: 375, Total sell A? 6400.0
Price of A: 380, Total sell A? 6423.0303030293
Price of A: 385, Total sell A? 6423.0303030293
Price of A: 390, Total sell A? 6423.0303030293
Price of A: 395, Total sell A? 6423.0303030293
Price of A: 400, Total sell A? 6423.0303030293
Price of A: 405, Total sell A? 6423.030303029311
```

Price of A: 410, Total sell A? 6423.030303029311  
Price of A: 415, Total sell A? 6423.030303029311  
Price of A: 420, Total sell A? 6423.0303030293  
Price of A: 425, Total sell A? 6423.030303029311  
Price of A: 430, Total sell A? 6423.030303029311  
Price of A: 435, Total sell A? 6423.030303029311  
Price of A: 440, Total sell A? 6423.030303029311  
Price of A: 445, Total sell A? 6423.030303029311  
Price of A: 450, Total sell A? 6423.030303029311  
Price of A: 455, Total sell A? 6423.030303029311  
Price of A: 460, Total sell A? 6423.0303030293  
Price of A: 465, Total sell A? 6423.030303029311  
Price of A: 470, Total sell A? 6423.030303029311  
Price of A: 475, Total sell A? 6423.0303030293  
Price of A: 480, Total sell A? 6423.030303029311  
Price of A: 485, Total sell A? 6423.0303030293  
Price of A: 490, Total sell A? 6423.0303030293  
Price of A: 495, Total sell A? 6423.0303030293  
for B -----  
Price of B: 300, Total sell B? 0.0  
Price of B: 305, Total sell B? 0.0  
Price of B: 310, Total sell B? 0.0  
Price of B: 315, Total sell B? 0.0  
Price of B: 320, Total sell B? 0.0  
Price of B: 325, Total sell B? 0.0  
Price of B: 330, Total sell B? 0.0  
Price of B: 335, Total sell B? 0.0  
Price of B: 340, Total sell B? 0.0  
Price of B: 345, Total sell B? 0.0  
Price of B: 350, Total sell B? 0.0  
Price of B: 355, Total sell B? 0.0  
Price of B: 360, Total sell B? 0.0  
Price of B: 365, Total sell B? 0.0  
Price of B: 370, Total sell B? 0.0  
Price of B: 375, Total sell B? 0.0  
Price of B: 380, Total sell B? 0.0  
Price of B: 385, Total sell B? 0.0  
Price of B: 390, Total sell B? 0.0  
Price of B: 395, Total sell B? 0.0  
Price of B: 400, Total sell B? 0.0  
Price of B: 405, Total sell B? 0.0  
Price of B: 410, Total sell B? 0.0  
Price of B: 415, Total sell B? 0.0  
Price of B: 420, Total sell B? 0.0  
Price of B: 425, Total sell B? 0.0  
Price of B: 430, Total sell B? 0.0  
Price of B: 435, Total sell B? 0.0  
Price of B: 440, Total sell B? 0.0  
Price of B: 445, Total sell B? 0.0

Price of B: 450, Total sell B? 0.0  
Price of B: 455, Total sell B? 0.0  
Price of B: 460, Total sell B? 0.0  
Price of B: 465, Total sell B? 0.0  
Price of B: 470, Total sell B? 0.0  
Price of B: 475, Total sell B? 0.0  
Price of B: 480, Total sell B? 0.0  
Price of B: 485, Total sell B? 0.0  
Price of B: 490, Total sell B? 0.0  
Price of B: 495, Total sell B? 0.0  
Price of B: 500, Total sell B? 2300.0  
Price of B: 505, Total sell B? 2352.88888888789  
Price of B: 510, Total sell B? 2352.88888888789  
Price of B: 515, Total sell B? 2352.88888888789  
Price of B: 520, Total sell B? 6399.999999999  
Price of B: 525, Total sell B? 6400.0  
Price of B: 530, Total sell B? 6400.0  
Price of B: 535, Total sell B? 6423.0303030293  
Price of B: 540, Total sell B? 6423.0303030293  
Price of B: 545, Total sell B? 6423.0303030293  
Price of B: 550, Total sell B? 6423.0303030293  
Price of B: 555, Total sell B? 6423.0303030293  
Price of B: 560, Total sell B? 6423.0303030293  
Price of B: 565, Total sell B? 6423.0303030293  
Price of B: 570, Total sell B? 6423.0303030293  
Price of B: 575, Total sell B? 6423.0303030293  
Price of B: 580, Total sell B? 6423.0303030293  
Price of B: 585, Total sell B? 6423.0303030293  
Price of B: 590, Total sell B? 6423.0303030293  
Price of B: 595, Total sell B? 6423.0303030293  
Price of B: 600, Total sell B? 6423.0303030293  
Price of B: 605, Total sell B? 6423.0303030293  
Price of B: 610, Total sell B? 6423.0303030293  
Price of B: 615, Total sell B? 6423.0303030293  
Price of B: 620, Total sell B? 6423.0303030293  
Price of B: 625, Total sell B? 6423.0303030293  
Price of B: 630, Total sell B? 6423.0303030293  
Price of B: 635, Total sell B? 6423.0303030293  
Price of B: 640, Total sell B? 6423.0303030293  
Price of B: 645, Total sell B? 6423.0303030293  
Price of B: 650, Total sell B? 6423.0303030293  
Price of B: 655, Total sell B? 6423.0303030293  
Price of B: 660, Total sell B? 6423.030303029311  
Price of B: 665, Total sell B? 6423.0303030293  
Price of B: 670, Total sell B? 6423.0303030293  
Price of B: 675, Total sell B? 6423.0303030293  
Price of B: 680, Total sell B? 6499.999999999  
Price of B: 685, Total sell B? 6666.66666666567  
Price of B: 690, Total sell B? 6700.0

```
Price of B: 695, Total sell B? 6966.66666666567
Price of B: 700, Total sell B? 7000.0
Price of B: 705, Total sell B? 7066.66666666567
Price of B: 710, Total sell B? 7366.66666666667
Price of B: 715, Total sell B? 8166.666666665669
Price of B: 720, Total sell B? 8166.666666665669
Price of B: 725, Total sell B? 8166.666666665669
Price of B: 730, Total sell B? 9166.66666666567
Price of B: 735, Total sell B? 9666.66666666567
Price of B: 740, Total sell B? 9666.66666666567
Price of B: 745, Total sell B? 9666.66666666567
Price of B: 750, Total sell B? 9666.66666666567
Price of B: 755, Total sell B? 9666.66666666567
Price of B: 760, Total sell B? 9666.66666666567
Price of B: 765, Total sell B? 9666.66666666567
Price of B: 770, Total sell B? 9666.66666666567
Price of B: 775, Total sell B? 9666.66666666567
Price of B: 780, Total sell B? 9666.66666666567
Price of B: 785, Total sell B? 9666.66666666567
Price of B: 790, Total sell B? 9666.66666666567
Price of B: 795, Total sell B? 9666.66666666567
no results saved.
```

## Explanation

As we see, for Alloy A to be sold in Abadan, we should have a selling price above 350, and for B to be sold, we should have a price above 500, higher price for B is because prices in other markets are higher for B, thus we should sell B in Abadan for higher price to take some share of other markets which are currently profitable.