SAYANTAN RAHA

Roll # : BAI09056

IIMB - BAI09 - Assignment 3

Out[4]: Toggle on/off Code

Q1 - 1

- The optimal values for the Problem are as follows. The following has been obtained by solving the set of following LP formulation:

Maximize Objective: 50 x_dh + 45 x_fn + 10 x_kbh + 60 x_kh

Decision variables:

- x_fn: quantity of FN
- x_dh: quantity of DH
- x_kh: quantity of KH
- x_kbh: quantity of KBH

Subject To

- Demand_1: x_fn <= 250
- Demand_2: x_dh <= 120
- Demand_3: x_kh <= 90
- Demand_4: x_kbh <= 550
- Non_Zero_Constraint_1: x_fn >= 0
- Non_Zero_Constraint_2: x_dh >= 0
- Non_Zero_Constraint_3: x_kh >= 0
- Non_Zero_Constraint_4: x_kbh >= 0
- Supply_1: 0.2 x_dh + 0.3 x_fn + 0.3 x_kbh + 0.4 x_kh <= 500
- Supply_2: 0.5 x_dh + 0.4 x_fn + 0.8 x_kbh + 0.8 x_kh <= 450
- Supply_3: 0.3 x_dh + 0.4 x_fn + 0.1 x_kbh + 0.1 x_kh <= 75
- Supply_4: 0.3 x_dh + 0.2 x_fn + 0.5 x_kbh + 0.6 x_kh <= 300
- Supply_5: 0.5 x_dh + 0.5 x_fn + 0.8 x_kbh + 0.4 x_kh <= 200

```
 fn :: 75.0 ::
 dh :: 120.0 ::
 kh :: 90.0 ::
 kbh :: 0.0 ::
Objective 14775.0
```

Q1 - 2

- From the **SENSITIVITY ANALYSIS REPORT** report above we can clearly see that amount of KBH to be produced to maximize profit under current profit rates is **Zero (0)**

The **Reduced Cost / Marginal Cost for Objective Coefficient** for KBH is -1.25 implying the profit will reduce by 1.25 units if AH are to produce one unit of KBH. It is a Non-Basic variable with coefficient equal to Zero

Q1 - 3

- From the **SENSITIVITY ANALYSIS REPORT** report above we can clearly see that **Supply_1 Constraint (Availability of Maida = 500 Kg)** is not a binding constraint. There is already 417 Kg of extra (SLACK) Maida available with the supplier. Hence he should not be procuring the extra Maida from his friend.

Unless his friend provides him the Maida free of cost (as cost of 50 Kg of Maida will become Zero and will reduce the cost price of items using Maida) there is no change in the value of the objective function of the current solution

Q1 - 4

Assuming this question is for KH

- AH is producing 90Kg of KH @ Profit of 60/unit. Hence he can accept the order of 20Kg from the Halva Shop

Assuming this question is for KBH

- From the Sensitivity Report we can see that, Reduced Cost / Marginal Cost for Objective Coefficient for KBH is -1.25. In order for him to accept any orders for KBH the minimum value of Profit from KBH should be 11.25/unit. Hence he should increase the Profit on KBH by 1.25/unit, if he is to accept this order

Q1 - 5

- From the Sensitivity Report we can see that the Profit on DH can be reduced by max of 16.25/unit for the current solution to remain optimal. Hence providing a discount of 10 INR/unit of DH **does not change** the optimal production plan

Q1 - 6

- ASSUMPTIONS for the following solution
  - We are increasing the profit amounts by 20% implying the Profit of KBH will increase from 10 to 12

```
Increased Profit for DH = 54.0
Increased Profit for FN = 60.0
Increased Profit for KH = 72.0
Increased Profit for KBH = 12.0
```

- Since the simultaneous increase in the coefficients of the Non-Basic Variables (non Zero) are withing the permissible ranges (as seen from the Sensitivity Report) and the sum of percentage increase is less than 100%, hence as per the 100% Rule there is **no change in the optimal Solution**

```
Current Profit due to change in Profit Values = 17730
```

Q1 - 7

- As per the sensitivity Report the Constraint Supply_3 is binding and has the highest Marginal Cost (112.5). This constraint corresponds to the Supply Constraints for Fruits and Nuts.
- What this implies:
  - Increasing availability of Fruits and Nuts from 75 Kg by one unit increases profit by 112.5 INR
  - The above is valid only in the amount of Fruits and Nuts are increases from current available level of 75 till 128. Beyond this range if availability is increased then the current optimal solution will not hold true

Q1 - 8

- From the Sensitivity Report we can see that the Profit on DH can be reduced to 33.75/unit for the current solution to remain optimal.
- From the Sensitivity Report we can see that the Profit on KH can be reduced to 11.25/unit for the current solution to remain optimal.

- As per the problem statement the reduction in DH is 8/Unit and reduction in KH is 24/Unit
- We will compute the % change and use the 100% Rules to check if the changes are below 100% or not. Since both are Non-Basic Variables hence if the allowed change is less than 100% hence from using the 100% Rule for change in Objective coefficients we know the Optimal Solution will remain unchanged

```
% change in DH in the allowed direction = 0.49230769230769234
% change in KH in the allowed direction = 0.49230769230769234

Sum of the % changes in the allowed directions is < 100%, hence there is no change in the Optimal Solution
```

Q2 - a

Decision Variables: 15 Binary Variables for each compartment and each Fule Type. This will indicate which type of Fuel should be carried in which container.

e.g. ys1 = 1, will indicate Fuel S is being carried in Container 1 We use Python (PULP and GLPK Solver to solve the solution). Hence the DV will be as follows:

compartments = ['1', '2', '3', '4', '5']

- ys = LpVariable.dicts("Ys_", compartments, 0, None, cat = LpBinary) # 5 Variables for Fuel S
- yr = LpVariable.dicts("Yr_", compartments, 0, None, cat = LpBinary) # 5 Variables for Fuel R
- yu = LpVariable.dicts("Yu_", compartments, 0, None, cat = LpBinary) # 5 Variables for Fuel U

15 Binary Variables for quantity of fuel carried in each compartment. If corresponding binary indicator is 1, this will have non zero value.

e.g. ys1 = 1, s1 = 2800

In Python we will define these variables as follows:

- S = pulp.LpVariable.dict('S_%s', compartments, lowBound = 0)
- R = pulp.LpVariable.dict('R_%s', compartments, lowBound = 0)
- U = pulp.LpVariable.dict('U_%s', compartments, lowBound = 0)

2 - b

Constraints

- Demand_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 <= 4000
- Demand_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 <= 2900
- Demand_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 <= 4900
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_1: Yr**1 + Ys**1 + Yu__1 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_2: Yr**2 + Ys**2 + Yu__2 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_3: Yr**3 + Ys**3 + Yu__3 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_4: Yr**4 + Ys**4 + Yu__4 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_5: Yr**5 + Ys**5 + Yu__5 <= 1
- Maximum_Capacity_of_R_Fuel_if_Container_1_has_R: R_1 - 2700 Yr__1 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_2_has_R: R_2 - 2800 Yr__2 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_3_has_R: R_3 - 1100 Yr__3 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_4_has_R: R_4 - 1800 Yr__4 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_5_has_R: R_5 - 3400 Yr__5 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_1_has_S: S_1 - 2700 Ys__1 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_2_has_S: S_2 - 2800 Ys__2 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_3_has_S: S_3 - 1100 Ys__3 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_4_has_S: S_4 - 1800 Ys__4 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_5_has_S: S_5 - 3400 Ys__5 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_1_has_U: U_1 - 2700 Yu__1 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_2_has_U: U_2 - 2800 Yu__2 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_3_has_U: U_3 - 1100 Yu__3 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_4_has_U: U_4 - 1800 Yu__4 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_5_has_U: U_5 - 3400 Yu__5 <= 0
- Maximum*Shortfall*(500)_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 >= 3500
- Maximum*Shortfall*(500)_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 >= 2400
- Maximum*Shortfall*(500)_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 >= 4400
- Binaries Yr**1 Yr**2 Yr**3 Yr**4 Yr**5 Ys**1 Ys**2 Ys**3 Ys**4 Ys**5 Yu**1 Yu**2 Yu**3 Yu**4 Yu__5
- Non Zero: S1, S2, S3, S4, S5, R1, R2, R3, R4, R5, U1,U2, U3, U4, U5 >= 0

2 - c

Objective

Minimize Objective: - 8 R_1 - 8 R_2 - 8 R_3 - 8 R_4 - 8 R_5 - 10 S_1 - 10 S_2 - 10 S_3 - 10 S_4 - 10 S_5 - 6 U_1 - 6 U_2 - 6 U_3 - 6 U_4 - 6 U_5

1. 8 = Penalty on not fulfilling R typer Fuel / Litre
2. 10 = Penalty on not fulfilling S typer Fuel / Litre
3. 6 = Penalty on not fulfilling U typer Fuel / Litre

- Decision Variables : Fuel of Type S/R/U in Container 1/2/3/4/5: S1, S2, S3, S4, S5, R1, R2, R3, R4, R5, U1,U2, U3, U4, U5 >= 0

2 - d

In order to incorporate the new Penalty Structure, I will modify the objective and the constarints in the following manner:

New Decision Variables:

- D1_0 = Non Zero value means S1 deficit is more than 250 by D1_0 amount. D1_0 will zero if Total deficit is less 250. e.g. If D1_0 = 10, defict = 250+10 =260
- D1_1 = Non Zero value means S1 deficit is less than 250 by D1_1 amount. D1_1 will zero if Total deficit is more 250. e.g. If D1_1 = 10, defict = 250-10 =240
- D2_0 = Same logic for quantity > 250 for R
- D2_1 = Same logic for quantity < 250 for R
- D3_0 = Same logic for quantity > 250 for U
- D3_1 = Same logic for quantity < 250 for U

New Objective

- Minimize Objective: 10 *(250 - d1[1]) + 11* d1[0]+ 8 *(250 - d2[1]) + 8.8* d2[0] + 6 *(250 - d3[1]) + 6.6* d3[0]

Logic: If quantity > 250: # assume 260 for S1 D1_1 = 10, D1_0 = 0, Penalty = 10 *(250-D1_0) + 11* D1_1 Logic: If quantity < 250: # assume 240 for S1 D1_1 = 0, D1_0 = 10, Penalty = 10 *(250-D1_0) + 11* D1_1

Subject To

- New Constraints
  - _C1: S_1 + S_2 + S_3 + S_4 + S_5 + 250 + d1_0 - d1_1 <= 2900
  - _C2: R_1 + R_2 + R_3 + R_4 + R_5 + 250 + d2_0 - d2_1 <= 4000
  - _C3: U_1 + U_2 + U_3 + U_4 + U_5 + 250 + d3_0 - d3_1 <= 4900

Earlier Constraints

- Demand_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 == 4000
- Demand_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 == 2900
- Demand_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 == 4900
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_1: Yr**1 + Ys**1 + Yu__1 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_2: Yr**2 + Ys**2 + Yu__2 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_3: Yr**3 + Ys**3 + Yu__3 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_4: Yr**4 + Ys**4 + Yu__4 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_5: Yr**5 + Ys**5 + Yu__5 <= 1
- Maximum_Capacity_of_R_Fuel_if_Container_1_has_R: R_1 - 2700 Yr__1 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_2_has_R: R_2 - 2800 Yr__2 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_3_has_R: R_3 - 1100 Yr__3 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_4_has_R: R_4 - 1800 Yr__4 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_5_has_R: R_5 - 3400 Yr__5 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_1_has_S: S_1 - 2700 Ys__1 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_2_has_S: S_2 - 2800 Ys__2 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_3_has_S: S_3 - 1100 Ys__3 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_4_has_S: S_4 - 1800 Ys__4 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_5_has_S: S_5 - 3400 Ys__5 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_1_has_U: U_1 - 2700 Yu__1 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_2_has_U: U_2 - 2800 Yu__2 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_3_has_U: U_3 - 1100 Yu__3 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_4_has_U: U_4 - 1800 Yu__4 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_5_has_U: U_5 - 3400 Yu__5 <= 0
- Maximum*Shortfall*(500)_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 >= 3500
- Maximum*Shortfall*(500)_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 >= 2400
- Maximum*Shortfall*(500)_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 >= 4400
- Binaries: Yr**1 Yr**2 Yr**3 Yr**4 Yr**5 Ys**1 Ys**2 Ys**3 Ys**4 Ys**5 Yu**1 Yu**2 Yu**3 Yu**4 Yu__5 End

Q2 - e

New Objective:

Minimize Objective: - 10 *(250 - d1[1]) + 11* (250 + d1[0]) - 10 *250* (1 - y1[0]) - 11 *250* (y1[0]) +\ 8 *(250 - d2[1]) + 8.8* (250 + d2[0]) - 8 *250* (1 - y1[1]) - 8.8 *250* (y1[1]) +\ 6 *(250 - d3[1]) + 6.6* (250 + d3[0]) - 6 *250* (1 - y1[2]) - 6.6 *250* (y1[2])

Logic:

- If S deficit = 240,
  - d1_1 = 10, Y[1] = 1, hence Loss = 10 *240 + 11* 250 - 0 - 11 * 250 ...
- If S deficit = 260,
  - d1_0 = 10, Y[1] = 0, hence Loss = 10 *250 + 11* 260 - 10 * 250 - 8 ...

Subject To

New Constraints

- _C4: - 300 Y1__0 + d1_1 <= 0
- _C5: - 300 Y1__1 + d2_1 <= 0
- _C6: - 300 Y1__2 + d3_1 <= 0

Setting an binary variable to 1 if D1_1 (implies deficit is less 250), else 0. This will be used in the objective function

Older COnstraints

- _C1: S_1 + S_2 + S_3 + S_4 + S_5 + 250 + d1_0 - d1_1 <= 2900
- _C2: R_1 + R_2 + R_3 + R_4 + R_5 + 250 + d2_0 - d2_1 <= 4000
- _C3: U_1 + U_2 + U_3 + U_4 + U_5 + 250 + d3_0 - d3_1 <= 4900
- Demand_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 <= 4000
- Demand_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 <= 2900
- Demand_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 <= 4900
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_1: Yr**1 + Ys**1 + Yu__1 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_2: Yr**2 + Ys**2 + Yu__2 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_3: Yr**3 + Ys**3 + Yu__3 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_4: Yr**4 + Ys**4 + Yu__4 <= 1
- Integer_Constraint_for_One_Type_of_Fuel_in_Container_5: Yr**5 + Ys**5 + Yu__5 <= 1
- Maximum_Capacity_of_R_Fuel_if_Container_1_has_R: R_1 - 2700 Yr__1 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_2_has_R: R_2 - 2800 Yr__2 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_3_has_R: R_3 - 1100 Yr__3 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_4_has_R: R_4 - 1800 Yr__4 <= 0
- Maximum_Capacity_of_R_Fuel_if_Container_5_has_R: R_5 - 3400 Yr__5 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_1_has_S: S_1 - 2700 Ys__1 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_2_has_S: S_2 - 2800 Ys__2 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_3_has_S: S_3 - 1100 Ys__3 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_4_has_S: S_4 - 1800 Ys__4 <= 0
- Maximum_Capacity_of_S_Fuel_if_Container_5_has_S: S_5 - 3400 Ys__5 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_1_has_U: U_1 - 2700 Yu__1 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_2_has_U: U_2 - 2800 Yu__2 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_3_has_U: U_3 - 1100 Yu__3 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_4_has_U: U_4 - 1800 Yu__4 <= 0
- Maximum_Capacity_of_U_Fuel_if_Container_5_has_U: U_5 - 3400 Yu__5 <= 0
- Maximum*Shortfall*(500)_Constraint_for_R_Fuel: R_1 + R_2 + R_3 + R_4 + R_5 >= 3500
- Maximum*Shortfall*(500)_Constraint_for_S_Fuel: S_1 + S_2 + S_3 + S_4 + S_5 >= 2400
- Maximum*Shortfall*(500)_Constraint_for_U_Fuel: U_1 + U_2 + U_3 + U_4 + U_5 >= 4400
- Binaries: Yr**1 Yr**2 Yr**3 Yr**4 Yr**5 Ys**1 Ys**2 Ys**3 Ys**4 Ys**5 Yu**1 Yu**2 Yu**3 Yu**4 Yu__5 y1_0 y1_1 y2_0 y2_1 y3_0 y3_1 End

Q3 - 1

- Optimal Production PLan
  - Alloy 1 - 0
  - Alloy 2 - 1500
  - Alloy 3 - 500
  - Alloy 4 - 1000

These can be derived from the Constraints Section of the Sensitivity Report

```
Objective Function Value = 495000
```

Q3 - 2 - Check / Object will definitely increase

As per the sesitivity Report Maximum Allowable increase in Metal 3 is 100 GM. Hence if 200 GM of metal is imported then this exceeds allowable increase for the Optimal Solution. This will change the current Optimal Solution, unfortunately the sensitivity report does not tell how it will change.

From Solving the dual solution we see that the Shadow Price for Y3 is 555 (Please see ans to 3-4, details are provided). Hence even if we procure 200 Gm of Metal 3, cost incurred is 20000, while for the 100 GM we know a additional revnue of 55500 can be made, even if they decide not to use the additional 100 GM as it will change the optimal solution.

Hence we can procure the amount.

Q3 - 3

- From the sensitivity report we can see that the Demand for Alloy 3 can be increased by 750 units without changing the Optimal Solution. So if the demand increases by 200, it will not change the optimal solution.
- If we solve the Dual Formulation, the Dual Price for Y6 (Corresponding Variable for the Demand Constraint for Alloy 3) is 59 (See 3-4 for Details below). Hence they will add 5 * 200 = 11,800. Overall Objective = 506800
- However, we know the profit per unit of Alloy 3 is 281. Ideally as per my understanding the Dual Price should have been same as the Profit (Objective Value), may be the numbers have been changed from the actual formuation. I will consider the increase in objective to be equal to the increase due to profit from selling the additional 200 units of Alloy 3. Hence the Objective will increase by 281 * 200 = 56200,

Final Objective = 551200

Q 3 - 4

- The corresponding Dual formulations are:

0.2y1 + 0.4y2 + 0.4y3 + y4 + 0 + 0 + 0 >= 186

0.2y1 + 0.6y2 + 0.2y3 + 0 + y5 + 0 + 0 >= 111

0.3y1 + 0.3y2 + 0.4y3 + 0 + 0 + y6 + 0 >= 281

0.5y1 + 0.5y2 + 0 + 0 + 0 + 0 + y7 >= 188

- Constraints 3, 6, 7 are binding in Primal, hence the corresponding Dual Variables are non-Zero. Hence y3, y6 and y7 are non zero.

Solving above equations we get: y3 = 555 y6 = 59 y7 = 188

Substituting the value of Y3 in equation 1 we get the Surplus = 36. This is the Reduced Cost for x1. hence the profit needs to be increased by 36 (Profit = 222) for Sedon to accept the order

Q 3 - 5

- As we can clearly see that the increase in the profit values for all are well outside the range of the permissible values as per the sensitivity report. The Solution will no longer remain optimal with these revised objective values. We need to resolve to find new Optimal Solution with these Profits

Q 3 - 6

- From sensitivity report we see that maximum Allowed decrease of Alloy 2 is 500 (Actual Value = 1000), which does not alter the optimal solution.
- The government restriction which reduces sales of alloy 2 value to 1000, hence does not impact the optimal Solution

Q 3 - 7

- Maximum Permissible change (increase) for Alloy 3 = 500. Actual change = 500. Hence percentage increase is 100%
- Maximum Permissible change (decrease) for Alloy 2 = 500. Actual change = 500. Hence percentage increase is 100%

Since the simultatneous change is more than 100%, hence this violates the 100% Rule for change. The solution may no longer remain optimal. We need to resolve the problem before making any conclusions.

Q 3 - 8

- Alloy 5 is introduced

The new sets of Primal equations are:

- Objective:

186x1 + 111x2 + 281x3 + 188x4 + 220x5

- Constraints:

0.2x1 +0.2x2 + 0.3x3 + 0.5x4 + 0.5x5 <= 2000

0.4x1 +0.6x2 + 0.3x3 + 0.5x4 + 0.4x5 <= 3000

0.2x1 +0.2x2 + 0.4x3 + 0x4 + 0.1x5 <= 500

x1 <= 1000

x2 <= 2000

x3 <= 500

x4 <= 1000

x5 <= 1500

Let us assume that x5 = 0, then the formulation remains optimal if the following dual constraint is feasible and non-binding:

0.5y1 + 0.4y2 + 0.1y3 + 0 + 0 + 0 + 0 + 0 >= 220

We know : y1, y2 = 0

y3 = 555,

hence 55.5 >= 220, which is infeasible. Hence x5 != 0, and with new alloy the old solution is no longer optimal

Q4

Decision Variables

- hiredEmpl_i, i = 1,2,3,4. Employees to be hired for MOnths Septemner, October, January, February
- transfrEmpl_i, i= 1..6, Employees to be trasferred from othe location all 6 months
- Binary Variables: isReqHire_1 - (Hiring in March), isReqHire_2 - (Hiring in December)
- analyst = 79, as of First Day of September

Minimize Objective: 36000 analyst_1 + 34800 hiredEmpl_1 + 29100 hiredEmpl_2 + 12000 hiredEmpl_3 + 6000 hiredEmpl_4 + 20000 isReqHire_1 + 20000 isReqHire_2 + 8000 transfrEmpl_1 + 8000 transfrEmpl_2 + 8000 transfrEmpl_3 + 8000 transfrEmpl_4 + 8000 transfrEmpl_5 + 8000 transfrEmpl_6

Subject To

- Demand_Dec: analyst_1 + 0.95 hiredEmpl_1 + 0.95 hiredEmpl_2 + 0.8 transfrEmpl_4 >= 65
- Demand_Feb: analyst_1 + 0.95 hiredEmpl_1 + 0.95 hiredEmpl_2 + hiredEmpl_3 + hiredEmpl_4 + 0.8 transfrEmpl_6 >= 90
- Demand_Jan: analyst_1 + 0.95 hiredEmpl_1 + 0.95 hiredEmpl_2 + hiredEmpl_3 + 0.8 transfrEmpl_5 >= 80
- Demand_Nov: analyst_1 + 0.95 hiredEmpl_1 + hiredEmpl_2 + 0.8 transfrEmpl_3 >= 90
- Demand_Oct: analyst_1 + hiredEmpl_1 + hiredEmpl_2 + 0.8 transfrEmpl_2 >= 105
- Demand_Sept: analyst_1 + hiredEmpl_1 + 0.8 transfrEmpl_1 >= 110
- Initial_Number_of_confirmed_Analyst_as_of_Sept01: analyst_1 = 79
- Transferred_Employee_20%_Constraint: 1.2 analyst_1 + 1.16 hiredEmpl_1 + 0.97 hiredEmpl_2 + 0.4 hiredEmpl_3 + 0.2 hiredEmpl_4 - transfrEmpl_1 - transfrEmpl_2 - transfrEmpl_3 - transfrEmpl_4 - transfrEmpl_5 - transfrEmpl_6 >= 0
- Hired Employees per month _C1: hiredEmpl_1 - 50 isReqHire_1 <= 0
- _C2: hiredEmpl_2 - 50 isReqHire_1 <= 0
- _C3: hiredEmpl_3 - 50 isReqHire_2 <= 0
- _C4: hiredEmpl_4 - 50 isReqHire_2 <= 0
- 0 <= hiredEmpl_1
- 0 <= hiredEmpl_2
- 0 <= hiredEmpl_3
- 0 <= hiredEmpl_4
- 0 <= transfrEmpl_1
- 0 <= transfrEmpl_2
- 0 <= transfrEmpl_3
- 0 <= transfrEmpl_4
- 0 <= transfrEmpl_5
- 0 <= transfrEmpl_6

End

```
Transferred Employee by Month 1 :: 25 ::
Transferred Employee by Month 2 :: 19 ::
Transferred Employee by Month 3 :: 1 ::
Transferred Employee by Month 4 :: 0 ::
Transferred Employee by Month 5 :: 0 ::
Transferred Employee by Month 6 :: 1 ::
Is Hiring required Period 1 :: 1 ::
Is Hiring required Period 2 :: 0 ::
Nu mber of Hired Employees by Month1 :: 11 ::
Nu mber of Hired Employees by Month2 :: 0 ::
Nu mber of Hired Employees by Month3 :: 0 ::
Nu mber of Hired Employees by Month4 :: 0 ::
Number of Analyst on Month 1 (September) 1 :: 79.0 ::
Objective 3614800.0
```

Q5

- Problem is not solving if we have to match monthly demand. For Month 5 Demand == 2000, this cannot be achieved (Max that can be achived is 1600 for month 5)
- LP will solve if monthly constraints does not have to be met. Hence we will design a solution to meet overall demand of 7000 Stones

Decision variables (DV)

- Q1_Stone_i, i = 1..6
- Q2_Stone_i, i = 1..6
- carryFwd_i, i = 1..6 Carry for each period (Extra Stones)
- unMetDemand_i, i = 1..6 Unmet Demand for each period (Less stones Stones to be met in subsequent years)

Minimize Objective: 1000 carryFwd_0 + 1000 carryFwd_1 + 1000 carryFwd_2 + 1000 carryFwd_3 + 1000 carryFwd_4 + 1000 carryFwd_5 + 1000 carryFwd_6 + 215000 q1Stones_1 + 215000 q1Stones_2 + 215000 q1Stones_3 + 207500 q1Stones_4 + 207500 q1Stones_5 + 207500 q1Stones_6 + 240000 q2Stones_1 + 240000 q2Stones_2 + 240000 q2Stones_3 + 232500 q2Stones_4 + 232500 q2Stones_5 + 232500 q2Stones_6

Subject To

- All DV >0
- Stopped_Production_1: q2Stones_4 = 0
- Stopped_Production_2: q2Stones_5 = 0
- Overall Demand_Constraints_6: q1Stones_1 + q1Stones_2 + q1Stones_3 + q1Stones_4 + q1Stones_5 + q1Stones_6 + q2Stones_1 + q2Stones_2 + q2Stones_3 + q2Stones_4 + q2Stones_5 + q2Stones_6 >= 7000
- Production_Constraints_Q1_1: q1Stones_1 <= 800
- Production_Constraints_Q1_2: q1Stones_2 <= 800
- Production_Constraints_Q1_3: q1Stones_3 <= 800
- Production_Constraints_Q1_4: q1Stones_4 <= 800
- Production_Constraints_Q1_5: q1Stones_5 <= 800
- Production_Constraints_Q1_6: q1Stones_6 <= 800
- Production_Constraints_Q2_1: q2Stones_1 <= 1400
- Production_Constraints_Q2_2: q2Stones_2 <= 1400
- Production_Constraints_Q2_3: q2Stones_3 <= 1400
- Production_Constraints_Q2_4: q2Stones_4 <= 1400
- Production_Constraints_Q2_5: q2Stones_5 <= 1400
- Production_Constraints_Q2_6: q2Stones_6 <= 1400
- Stone_Storage_Limit_1: carryFwd_1 <= 1200
- Stone_Storage_Limit_2: carryFwd_2 <= 1200
- Stone_Storage_Limit_3: carryFwd_3 <= 1200
- Stone_Storage_Limit_4: carryFwd_4 <= 1200
- Stone_Storage_Limit_5: carryFwd_5 <= 1200
- Stone_Storage_Limit_6: carryFwd_6 <= 1200
- Demand_C1: - carryFwd_1 + q1Stones_1 + q2Stones_1 + unMetDemand_1 <= 700
- Demand_C2: carryFwd_1 - carryFwd_2 + q1Stones_2 + q2Stones_2 - unMetDemand_1 + unMetDemand_2 <= 700
- Demand_C3: carryFwd_2 - carryFwd_3 + q1Stones_3 + q2Stones_3 - unMetDemand_2 + unMetDemand_3 <= 1000
- Demand_C4: carryFwd_3 - carryFwd_4 + q1Stones_4 + q2Stones_4 - unMetDemand_3 + unMetDemand_4 <= 1200
- Demand_C5: carryFwd_4 - carryFwd_5 + q1Stones_5 + q2Stones_5 - unMetDemand_4 + unMetDemand_5 <= 2000
- Demand_C6: carryFwd_5 - carryFwd_6 + q1Stones_6 + q2Stones_6 - unMetDemand_5 + unMetDemand_6 <= 1400

End

```
1
Carry Forward :: Period ::  0 :: 0.0 ::
Carry Forward :: Period ::  1 :: 100.0 ::
Carry Forward :: Period ::  2 :: 200.0 ::
Carry Forward :: Period ::  3 :: 800.0 ::
Carry Forward :: Period ::  4 :: 400.0 ::
Carry Forward :: Period ::  5 :: 0.0 ::
Carry Forward :: Period ::  6 :: 0.0 ::
Quarry1 :: Period :: 1 :: 800.0 ::
Quarry1 :: Period :: 2 :: 800.0 ::
Quarry1 :: Period :: 3 :: 800.0 ::
Quarry1 :: Period :: 4 :: 800.0 ::
Quarry1 :: Period :: 5 :: 800.0 ::
Quarry1 :: Period :: 6 :: 800.0 ::
Quarry2 :: Period ::  1 :: 0.0 ::
Quarry2 :: Period ::  2 :: 0.0 ::
Quarry2 :: Period ::  3 :: 800.0 ::
Quarry2 :: Period ::  4 :: 0.0 ::
Quarry2 :: Period ::  5 :: 0.0 ::
Quarry2 :: Period ::  6 :: 1400.0 ::
Unmet Demand :: Period ::  1 :: 0.0 ::
Unmet Demand :: Period ::  2 :: 0.0 ::
Unmet Demand :: Period ::  3 :: 0.0 ::
Unmet Demand :: Period ::  4 :: 0.0 ::
Unmet Demand :: Period ::  5 :: 800.0 ::
Unmet Demand :: Period ::  6 :: 0.0 ::
Objective 1533000000.0
```

Q - 6

Decision Variables:

- xi, i=1..5. Mins for each type of advertisement
- d1_0: if this takes positive value, then GRP Overall (Goal 1) >=100
- d1_1: if this takes positive value, then GRP Overall (Goal 1)<=100
- d2_0: if this takes positive value, then GRP Sport (Goal 2)>=20
- d2_1: if this takes positive value, then GRP Sport (Goal 2)<=20
- d3_0: if this takes positive value, then GRP Eng Chnl (Goal 2)>=5
- d3_1: if this takes positive value, then GRP Eng Chnl (Goal 2)<=5

Minimize Objective: d1_1 + d2_1 + d3_0

Subject To

- Goal1_C1: - d1_0 + d1_1 + 4.2 x1 + 3.5 x2 + 2.8 x3 + 2.5 x4 + 0.2 x5 >= 100
- Goal2_C2: - d2_0 + d2_1 + 4.2 x1 + 3.5 x2 >= 20
- Goal3_C3: d3_0 - d3_1 - 0.2 x5 >= -5
- Budget_C4: 120000 x1 + 85000 x2 + 70000 x3 + 60000 x4 + 25000 x5 <= 2000000
- All Variables >= 0

End

```
Cricket :: 0 ::
Oth Sport :: 8 ::
Hindi Serial :: 0 ::
Hindi Movie :: 22 ::
English News :: 0 ::
D1 0 :: 0.0
D1 1 :: 17.0
D2 0 :: 0.0
D2 1 :: 0.0
D3 0 :: 0.0
D3 1 :: 0.0
Objective 17.0::
```