**Production Planning**

In an analysis of the UniCitrus production planning in order to maximize profit from producing orange juices, Standard and Dairy, there are various conditions that not only effects to the amount of production, but it also needs to concern varieties between the market demand and supply as well as a capacity of owned plant. Consequently, in the first part of this report will focus on a fundamental condition which returns an optimal solution in a case of an infinite machine capability. Alternatively, the second part will identify the limited machine capability condition and an unequal capacity in each types of juice that effect to a designing juice proportion. In the last session, it is essential to evaluate an impact of the excess fruit supply that influences to a production decision. However, all of these analytics bases on primary assumptions, therefore in some cases, the results might not be accurate or the most fitted optimization.

To begin with the optimization in production planning of the UniCitrus, a list of constraints and an objective function are shown by the first table, as a side, formulas are listed on the second point. Overview, this analysis divides products as two types; the intermediate goods (Hamlin, Pera, Valencia juices) and the final goods (Standard and Dairy juices), subsequently models in this analysis will focus on these two of production. An inventory system will preserve only the intermediate products only following an assumption that it would be flexible to change in any final productions. As a side, it is important to manipulate a production line in each single month because of a maximization of utilization in machines and labor, plus it might be insufficient in a real circumstance if managers decide to skip a manufacturing for only month in order to increase their short-term profit and ignoring real business routine. Therefore, this model will operate the factory every month. Furthermore, a unit in this analysis uses a tonne (1000kg.) to calculate the results. For example, a machine capability operates in the tonne or ordering orange stocks as the tonne. In summary, it can conclude that there are three main assumption as previously mentioned above or the table1.1.

Table1.1: Primary assumptions in this analysis

|  |
| --- |
| **Primary assumptios** |
| Separate an intermediate goods and final goods production |
| An inventory stock keeps only an intermediate goods |
| Initiating line production every month. |
| A tonne is a unit in this calculate |

**Table 2.1: Variables in an environment**

|  |  |  |
| --- | --- | --- |
| **Variables Name** | **Description** | **Type** |
| Decision Variables |  |  |
| Supply\_Standard | Total production of Standard for each month. (Tonne) | mpvar |
| Supply\_Dairy | Total production of Dairy for each month. (Tonne) | mpvar |
| All\_Supply | Total production of each month. (Tonne) | mpvar |
| Orange in uses for each type |  |  |
| Hamlin\_Use\_Standard | A number of Hamlin that blends in Standard. (Tonne) | mpvar |
| Pera\_Use\_Standard | A number of Pera that blends in Standard. (Tonne) | mpvar |
| Valen\_Use\_Standard | A number of Valencia that blends in Standard. (Tonne) | mpvar |
| Hamlin\_Use\_Dairy | A number of Hamlin that blends in Dairy. (Tonne) | mpvar |
| Pera\_Use\_Dairy | A number of Pera that blends in Dairy. (Tonne) | mpvar |
| Valen\_Use\_Dairy | A number of Valencia that blends in Dairy. (Tonne) | mpvar |
| Demand |  |  |
| Demand\_Standard | Demand of Standard for each month. (Tonne) | Real number |
| Demand\_Dairy | Demand of Dairy for each month. (Tonne) | Real number |
| Ordering size |  |  |
| Hamlin\_Buy | A number of Hamlin ordering size for each month. (Tonne) | mpvar |
| Pera\_Buy | A number of Pera ordering size for each month. (Tonne) | mpvar |
| Valen\_Buy | A number of Valencia ordering size for each month. (Tonne) | mpvar |
| Availability of Orange |  |  |
| Hamlin\_avia | A number of expected Hamlin available. (Tonne) | Real number |
| Pera\_avia | A number of expected Pera available. (Tonne) | Real number |
| Valen\_avia | A number of expected Valencia available. (Tonne) | Real number |
| Proportion constraints |  |  |
| P\_Hamlin\_Standard | A specification for Hamlin that are produced in Standard. (Tonne) | mpvar |
| P\_Pera\_Standard | A specification for Pera that are produced in Standard. (Tonne) | mpvar |
| P\_Valen\_Standard | A specification for Valencia that are produced in Standard. (Tonne) | mpvar |
| P\_Hamlin\_Dairy | A specification for Hamlin that are produced in Diary. (Tonne) | mpvar |
| P\_Pera\_Dairy | A specification for Pera that are produced in Diary. (Tonne) | mpvar |
| P\_Valencia\_Dairy | A specification for Valencia that are produced in Diary. (Tonne) | mpvar |

|  |  |  |
| --- | --- | --- |
| **Variable** | **Description** | **Type** |
| Orange Keep |  |  |
| K\_Hamlin | A number of Hamlin inventory. (Tonne) | mpvar |
| K\_Pera | A number of Pera inventory. (Tonne) | mpvar |
| K\_Valen | A number of Valencia inventory. (Tonne) | mpvar |
| K\_all | A number of an aggregate inventory. (Tonne) | mpvar |
| Addition |  |  |  |
| Production\_Cap | A machine production Capability. (Tonne) | Integer |
| Hamlin\_Production | An amount of Hamlin used in a production. (Tonne) | mpvar |
| Pera\_Production | An amount of Pear used in a production. (Tonne) | mpvar |
| Valen\_Production | An amount of Valencia used in a production. (Tonne) | mpvar |
| Inventory\_Hamlin | An amount of Hamlin inventory. (Tonne) | linctr |
| Inventory\_Pera | An amount of Pera inventory. (Tonne) | linctr |
| Inventory\_Valen | An amount of Valencia invenrory. (Tonne) | linctr |
| Bia | A binary value in considering initiate production | mpvar |
| Old\_Standard\_Valen | An amount of Valencia that produce Standard. | mpvar |
| New\_Standard\_Valen | An amount of new Valencia that produce Standard | mpvar |
| Old\_Dairy\_Valen | An amount of new Valencia that produce Dairy | mpvar |
| New\_Dairy\_Valen | An amount of new Valencia that produce Dairy | mpvar |
| Old\_Valen | An amount of Valencia from the first supplier | mpvar |
| New\_Valen | An amount of Valencia from the North supplier | mpvar |

**Table 2.2: Numeric Value**

Note: Production\_Cap are calculated into the tonne in order to compare to a fruit supply in the similar term.

(1850 = 500000 boxes of Pera equal 1850 tonne of orange.)

Therefore, machine capability is 1850 tonne for every fruits

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | Month1 | Month2 | Month3 |
| Hamlin\_avia | 1050 | 1225 | 350 |
| Pera\_avia | 925 | 1225 | 1750 |
| Valen\_avia | 0 | 175 | 350 |
| Demand\_Standard | 500 | 1500 | 700 |
| Demand\_Dairy | 200 | 100 | 100 |
| Production\_Cap | 1850 | 1850 | 1850 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Hamlin | Pera | Valencia | Standard Price | Dairy Price |
| Juice (kg/box) for each variety | 3.5 | 3.7 | 3.4 | 1000 | 1100 |

**Table 2: Summary of all constraint in the optimization**

**Objective Function**

Note: **λ is 1 in the first and second scenarios**

**Description:**

**Constraints**

**2.1 Proportion of each juice Description:**

Note: FUseStandard = Each fruit juice in uses for producing Standard Juice

Ai = Juice (kg/box) for each variety

Si = Proportion for each product

For Standard:

For Diary:

**2.2 Ordering amount for each fruit**

Note: These variables describe the amount of ordering each fruit.

**Description:**

**2.3 Comparing Order stock to Availability**

**Description:**

Note: Buying fruits responding to an availability of fruit in each month.

**2.4 Supply function**

**2.5 Limitation on selling**

**Description:**

Note: it is a condition that production must following

a demand

**2.6 Smoothing Optimization Description:**

Note: control a model uses Hamlin at least 1 tonne. However, it will buy a max Hamlin at 35.7 tonne, apart from that it is unbounded.

**2.7 Production Occurrence**

**2.8 Production Constraints Description:**

Note: Adjustment of the Hamlin that consumes more 10 percent of machine capability and less 10 percent for blending Valencia.

**2.9 Storing Stock**

**2.10 Model in considering open the production line**

**Description:**

Note:

**Table 4: Summary of an unlimited machine capability**

|  |  |  |  |
| --- | --- | --- | --- |
| Scenario: An unlimited machine capability | | | |
| **Production/month** | **Month: 1** | **Month: 2** | **Month: 3** |
| Standard\_Product | 500.00 | 1500.00 | 700.00 |
| Dairy\_Product | 0.00 | 100.00 | 100.00 |
| All Production | 500.00 | 1600.00 | 800.00 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Order** | **Month1** | **Month2** | **Month3** |
| Hamlin | 1.000 | 1.000 | 1.000 |
| Pera | 134.189 | 427.432 | 127.000 |
| Valencia | 0.000 | 4.412 | 96.029 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Uses** | **Month: 1** | **Month: 2** | **Month: 3** |
| **For Standard** | | | |
| Hamlin | 1.000 | 1.000 | 1.000 |
| Pera | 134.189 | 404.460 | 113.514 |
| Valencia | 0.000 | 0.000 | 81.323 |
| **For Dairy** | | | |
| Hamlin | 0.000 | 0.000 | 8.571 |
| Pera | 0.000 | 22.973 | 13.513 |
| Valencia | 0.000 | 4.412 | 14.705 |

|  |  |
| --- | --- |
| **Summary Production** | **Total** |
| Net Profit | £ 1,417,100.00 |
| Revenue | £ 2,920,000.00 |
| Fixed Cost | £ 1,500,000.00 |
| Operating Cost | £ 2,900.00 |
| **Production Detail** | |
| Machine Capacity | Infinite |
| Total Production | 2900.000 |

**Table 5: A limited machine Capability**

|  |  |  |  |
| --- | --- | --- | --- |
| **Scenario: A limited machine capability = 1850** | | | |
| **Production/month** | **Month: 1** | **Month: 2** | **Month: 3** |
| Standard\_Product | 500.00 | 1500.00 | 700.00 |
| Dairy\_Product | 0.00 | 100.00 | 100.00 |
| All Production | 500.00 | 1600.00 | 800.00 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Uses** | **Month: 1** | **Month: 2** | **Month: 3** |
| **For Standard** | | | |
| Hamlin | 35.714 | 43.514 | 13.203 |
| Pera | 101.350 | 364.243 | 176.699 |
| Valencia | 0.000 | 0.000 | 0.000 |
| **For Dairy** | | | |
| Hamlin | 0.000 | 0.000 | 8.571 |
| Pera | 0.000 | 22.973 | 14.864 |
| Valencia | 0.000 | 4.412 | 4.412 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Order** | **Month1** | **Month2** | **Month3** |
| Hamlin | 35.714 | 43.514 | 21.775 |
| Pera | 101.350 | 387.216 | 191.564 |
| Valencia | 0.000 | 4.412 | 4.412 |

|  |  |
| --- | --- |
| **Summary Production** | **Total** |
| Net Profit | £ 1,417,100.00 |
| Revenue | £ 2,920,000.00 |
| Fixed Cost | £ 1,500,000.00 |
| Operating Cost | £ 2,900.00 |
| **Production Detail** | |
| Machine Capacity | 1850 |
| Total Production | 2900.000 |

**Table 6: An excess supply of Valencia oranges**

|  |  |  |  |
| --- | --- | --- | --- |
| **Scenario: An excess supply of Valencia oranges** | | | |
| **Production/month** | **Month: 1** | **Month: 2** | **Month: 3** |
| Standard\_Product | 500.000 | 1500.000 | 700.000 |
| Dairy\_Product | 200.000 | 100.000 | 100.000 |
| All Production | 700.000 | 1600.000 | 800.000 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Order** | **Month1** | **Month2** | **Month3** |
| Hamlin | 1.000 | 0.000 | 1.000 |
| Pera | 162.162 | 417.973 | 202.703 |
| Valencia | 14.191 | 7.353 | 6.838 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Uses** | **Month: 1** | **Month: 2** | **Month: 3** |
| **For Standard** | | | |
| Hamlin | 0.000 | 0.000 | 0.000 |
| Pera | 135.135 | 404.459 | 189.189 |
| Valencia | 0.000 | 0.000 | 0.000 |
| **For Dairy** | | | |
| Hamlin | 1.000 | 0.000 | 1.000 |
| Pera | 27.027 | 13.514 | 13.513 |
| Valencia | 14.191 | 7.353 | 6.838 |

|  |  |
| --- | --- |
| **Summary Production** | **Total** |
| Net Profit | £ 1,640,000.00 |
| Revenue | £ 3,140,000.00 |
| Fixed Cost | £ 1,500,000.00 |
| Operating Cost | **λ** |
| **Production Detail** | |
| Machine Capacity | Unlimited |
| Total Production | 3100.000 |

**Suggestion**

|  |  |  |  |
| --- | --- | --- | --- |
| **Orange Store** | **Month: 1** | **Month: 2** | **Month: 3** |
| K\_Hamlin | 0.000 | 0.000 | 0.000 |
| K\_Pera | 0.000 | 0.000 | 0.000 |
| K\_Valen | 0.000 | 0.000 | 0.000 |

**Appendix**

**Code Block**

model Tour

uses "mmxprs";

declarations

!Objecyive function and month

month = 1..3

!Decision Variable to objective, Supply

Supply\_Standard : array(month) of mpvar

Supply\_Dairy : array(month) of mpvar

All\_Supply : array(month) of mpvar

!Orange in uses for each type, minor decision var

Hamlin\_Use\_Standard : array(month) of mpvar

Pera\_Use\_Standard : array(month) of mpvar

Valen\_Use\_Standard : array(month) of mpvar

Hamlin\_Use\_Dairy : array(month) of mpvar

Pera\_Use\_Dairy : array(month) OF mpvar

Valen\_Use\_Dairy : array(month) of mpvar

!Demand

Demand\_Standard : array(month) of integer

Demand\_Dairy : array(month) of integer

!Var to check with aviability, buy for each month

Hamlin\_Buy : array(month) of mpvar

Pera\_Buy : array(month) of mpvar

Valen\_Buy : array(month) of mpvar

!Aviability in each month

Hamlin\_avia : array(month) of integer

Pera\_avia : array(month) of integer

Valen\_avia : array(month) of integer

!Proportion constraints

P\_Hamlin\_Standard : array(month) of mpvar

P\_Pera\_Standard : array(month) of mpvar

P\_Valen\_Standard : array(month) of mpvar

P\_Hamlin\_Dairy : array(month) of mpvar

P\_Valen\_Dairy : array(month) of mpvar

P\_Pera\_Dairy : array(month) of mpvar

!Orange keep

K\_Hamlin : array(month) of mpvar

K\_Pera : array(month) of mpvar

K\_Valen : array(month) of mpvar

K\_all : array(month) of mpvar

!Addition

Production\_Cap : array(month) of integer

Hamlin\_Production :array(month) of mpvar

Pera\_Production :array(month) of mpvar

Valen\_Production:array(month) of mpvar

m : array(month) of integer

Inventory\_Hamlin :array(month) of linctr

Inventory\_Pera :array(month) of linctr

Inventory\_Valen :array(month) of linctr

New\_Standard\_Valen :array(month) of mpvar

Old\_Standard\_Valen :array(month) of mpvar

Old\_Dairy\_Valen :array(month) of mpvar

New\_Dairy\_Valen :array(month) of mpvar

Old\_Valen :array(month) of mpvar

New\_Valen :array(month) of mpvar

New\_Valen\_avai:array(month) of mpvar

lambda : real

New\_Valen\_Buy :array(month) of mpvar

Valen\_Buy\_Total:array(month) of mpvar

end-declarations

!List numeric array

!Aviability Constaints Value

Hamlin\_avia :: [1050,1225,350]

Pera\_avia :: [925, 1225,1750]

Valen\_avia :: [0,175,350]

!Demand Constrain Value

Demand\_Standard :: [500,1500,700]

Demand\_Dairy :: [200,100,100]

m ::[1,1,1]

**Question 1**

!Objective function

profit := (1000\*(SUM(i in month) Supply\_Standard(i))) + (1100\*(SUM(i in month) Supply\_Dairy(i))) - (500000\*(sum(i in month) m(i))) - (10\*(SUM(i in month) K\_all(i))) -((SUM(i in month) All\_Supply(i)))

!!!!!!Contraints!!!!!!!!

!!!!!!!!!!!Create supply!!!!!!!!!!!

!Proportion created and specification for each products!

forall(i in month)do

3.5\*Hamlin\_Use\_Standard(i) <= 0.25\*Supply\_Standard(i)

3.7\*Pera\_Use\_Standard(i) >= 0.6\*Supply\_Standard(i)

3.4\*Valen\_Use\_Standard(i) <= 0.4\*Supply\_Standard(i)

end-do

forall(i in month) do

3.5\*Hamlin\_Use\_Dairy(i) <= 0.3\*Supply\_Dairy(i)

3.7\*Pera\_Use\_Dairy(i) >= 0.5\*Supply\_Dairy(i)

3.4\*Valen\_Use\_Dairy(i) <= 0.5\*Supply\_Dairy(i)

3.4\*Valen\_Use\_Dairy(i) >= 0.15\*Supply\_Dairy(i)

end-do

!Buy fruit check

forall(i in month) do

Hamlin\_Use\_Standard(i) + Hamlin\_Use\_Dairy(i) + K\_Hamlin(i) + K\_Hamlin(i)\*3.5 + K\_Pera(i)\*3.7 + K\_Valen(i)\*3.4 = Hamlin\_Buy(i)

Pera\_Use\_Standard(i) + Pera\_Use\_Dairy(i) + K\_Pera(i) + K\_Hamlin(i)\*3.5 + K\_Pera(i)\*3.7 + K\_Valen(i)\*3.4 = Pera\_Buy(i)

Valen\_Use\_Standard(i) + Valen\_Use\_Dairy(i)+ K\_Valen(i) + K\_Hamlin(i)\*3.5 + K\_Pera(i)\*3.7 + K\_Valen(i)\*3.4 = Valen\_Buy(i)

end-do

!Compare Buy and Aviability in mrk

forall(i in month) do

Hamlin\_Buy(i) <= Hamlin\_avia(i)

Pera\_Buy(i) <= Pera\_avia(i)

Valen\_Buy(i) <= Valen\_avia(i)

end-do

!Supply function

forall(i in month) do

Hamlin\_Use\_Standard(i)\*3.5 + Pera\_Use\_Standard(i)\*3.7 + Valen\_Use\_Standard(i)\*3.4 = Supply\_Standard(i)

Hamlin\_Use\_Dairy(i)\*3.5 + Pera\_Use\_Dairy(i)\*3.7 + Valen\_Use\_Dairy(i)\*3.4 = Supply\_Dairy(i)

Supply\_Standard(i)+Supply\_Dairy(i) = All\_Supply(i)

end-do

!Demand = Supply constraints

forall(i in month) do

Supply\_Standard(i) <= Demand\_Standard(i)

Supply\_Dairy(i) <= Demand\_Dairy(i)

end-do

!Production Condition

!Set for smoothing optimization

forall(i in month)do

Hamlin\_Buy(i) >= 1

end-do

!Keep stock

forall(i in month)do

K\_Hamlin(i)+K\_Pera(i)+K\_Valen(i) = K\_all(i)

end-do

!No inventory last month

K\_Hamlin(3) = 0

K\_Pera(3) = 0

K\_Valen(3) = 0

Supply\_Standard(3) = Demand\_Standard(3)

Supply\_Dairy(3) = Demand\_Dairy(3)

!Improve production

!Create Var by each orange type to store how many tonne used in a production

forall(i in month)do

Hamlin\_Production(i) = Hamlin\_Use\_Standard(i) + Hamlin\_Use\_Dairy(i)

Pera\_Production(i) = Pera\_Use\_Standard(i) + Pera\_Use\_Dairy(i)

Valen\_Production(i) = Valen\_Use\_Standard(i) + Valen\_Use\_Dairy(i)

end-do

!To keep Hamlin, Pera, Valencia

Inventory\_Hamlin(1) := Hamlin\_Buy(1) - Hamlin\_Production(1) = K\_Hamlin(1)

Inventory\_Hamlin(2) := Hamlin\_Buy(2) - Hamlin\_Production(2) + K\_Hamlin(1) = K\_Hamlin(2)

Inventory\_Hamlin(3) := Hamlin\_Buy(3) - Hamlin\_Production(3) + K\_Hamlin(2) = K\_Hamlin(3)

Inventory\_Pera(1) := Pera\_Buy(1) - Pera\_Production(1) = K\_Pera(1)

Inventory\_Pera(2) := Pera\_Buy(2) - Pera\_Production(2) + K\_Pera(1) = K\_Pera(2)

Inventory\_Pera(3) := Pera\_Buy(3) - Pera\_Production(3) + K\_Pera(2) = K\_Valen(3)

Inventory\_Valen(1) := Valen\_Buy(1) - Valen\_Production(1) = K\_Valen(1)

Inventory\_Valen(2) := Valen\_Buy(2) - Valen\_Production(2) + K\_Valen(1) = K\_Valen(2)

Inventory\_Valen(3) := Valen\_Buy(3) - Valen\_Production(3) + K\_Valen(2) = K\_Valen(3)

**Question: 2**

!!!!!!!!Question2!!!!!!!!

Production\_Cap :: [1850,1850,1850] !! in each month, capacity = 1850 tonne

!Create Var by each orange type to store how many tonne used in a production

forall(i in month)do

Hamlin\_Production(i) = Hamlin\_Use\_Standard(i) + Hamlin\_Use\_Dairy(i)

Pera\_Production(i) = Pera\_Use\_Standard(i) + Pera\_Use\_Dairy(i)

Valen\_Production(i) = Valen\_Use\_Standard(i) + Valen\_Use\_Dairy(i)

end-do

!Production Constraints

!Machine constraints

forall(i in month) do

1.1\*Hamlin\_Production(i) <= Production\_Cap(i)

Pera\_Production(i) <= Production\_Cap(i)

0.9\*Valen\_Production(i) <= Production\_Cap(i)

end-do

!Binary to check if opening the production line or not

forall(i in month)do

Hamlin\_Production(i) + Pera\_Production(i) + Valen\_Production(i) <= 1850\*Bia(i)

Bia(i) <= 1

Hamlin\_Production(i) >= 0.1\*1850\*Bia(i)

Valen\_Production(i) <= 0.1\*1850\*Bia(i)

Hamlin\_Production(i) + Pera\_Production(i) + Valen\_Production(i) <= 10000000\*Bia(i)

end-do

!!!!!!!!!!!!!END Q2!!!!!!!!!!!!!

**Question: 3**

!!!!!!Q3!!!!!!!!!!!

!Condition for Question 3

forall(i in month)do

New\_Standard\_Valen(i) + Old\_Standard\_Valen(i) = Valen\_Use\_Standard(i)

New\_Dairy\_Valen(i) + Old\_Dairy\_Valen(i) = Valen\_Use\_Dairy(i)

Old\_Standard\_Valen(i) + Old\_Dairy\_Valen(i) = Old\_Valen(i)

New\_Standard\_Valen(i) + New\_Dairy\_Valen(i) = New\_Valen(i)

Old\_Valen(i) + New\_Valen(i) = Valen\_Use\_Standard(i)+Valen\_Use\_Dairy(i)

Valen\_Use\_Standard(i)+Valen\_Use\_Dairy(i) <= Valen\_avia(i) + New\_Valen\_avai(i)

New\_Valen(i) <= New\_Valen\_avai(i)

Old\_Valen(i) <= Valen\_avia(i)

end-do

!Supply function, Production function

forall(i in month) do

Hamlin\_Use\_Standard(i)\*3.5 + Pera\_Use\_Standard(i)\*3.7 + Valen\_Use\_Standard(i)\*3.4 + New\_Standard\_Valen(i)\*3.4 = Supply\_Standard(i)

Hamlin\_Use\_Dairy(i)\*3.5 + Pera\_Use\_Dairy(i)\*3.7 + Valen\_Use\_Dairy(i)\*3.4 + New\_Dairy\_Valen(i)\*3.4 = Supply\_Dairy(i)

Supply\_Standard(i)+Supply\_Dairy(i) = All\_Supply(i)

end-do

!Buy fruit check, to make New Valencia

forall(i in month) do

Hamlin\_Use\_Standard(i) + Hamlin\_Use\_Dairy(i) + K\_Hamlin(i) = Hamlin\_Buy(i)

Pera\_Use\_Standard(i) + Pera\_Use\_Dairy(i) + K\_Pera(i) = Pera\_Buy(i)

Valen\_Use\_Standard(i) + Valen\_Use\_Dairy(i)+ K\_Valen(i)+ New\_Valen\_Buy(i) = Valen\_Buy\_Total(i)

New\_Standard\_Valen(i) + New\_Dairy\_Valen(i) = New\_Valen\_Buy(i)

end-do

!Lambda value

lambda := 1100

forall(i in month) do

Valen\_Buy(i) + New\_Valen\_Buy(i) =Valen\_Buy\_Total(i)

end-do

!Compare Buy and Aviability in mrk, unlock problem of month 1

forall(i in month) do

Hamlin\_Buy(i) <= Hamlin\_avia(i)

Pera\_Buy(i) <= Pera\_avia(i)

end-do

Valen\_Buy\_Total(1) >= Valen\_avia(1)

Valen\_Buy\_Total(2) <= Valen\_avia(2)

Valen\_Buy\_Total(3) <= Valen\_avia(3)

!!!!!!END Q3!!!!!!

maximize(profit)

writeln("Profit is: ", getobjval, " GBP")

FORALL(i in month) DO

writeln("-----------------:")

writeln("Month: ",i)

writeln("Machine Capacity: ",getsol(Production\_Cap(i)))

writeln("Supply\_Standard= ", getsol(Supply\_Standard(i)))

writeln("Supply\_Dairy= ", getsol(Supply\_Dairy(i)))

writeln("Production Capacity= ", getsol(All\_Supply(i)))

writeln("Hamlin\_Use\_Standard is: ", getsol(Hamlin\_Use\_Standard(i)))

writeln("Hamlin\_Use\_Dairy is: ", getsol(Hamlin\_Use\_Dairy(i)))

writeln("K\_Hamlin is: ", getsol(K\_Hamlin(i)))

writeln("Hamlin\_Buy is: ", getsol(Hamlin\_Buy(i)))

writeln("Pera\_Use\_Standard is: ", getsol(Pera\_Use\_Standard(i)))

writeln("Pera\_Use\_Dairy is: ", getsol(Pera\_Use\_Dairy(i)))

writeln("K\_Pera is: ", getsol(K\_Pera(i)))

writeln("Pera\_Buy is: ", getsol(Pera\_Buy(i)))

writeln("Valen\_Use\_Standard is: ", getsol(Valen\_Use\_Standard(i)))

writeln("Valen\_Use\_Dairy is: ", getsol(Valen\_Use\_Dairy(i)))

writeln("K\_Valen is: ", getsol(K\_Valen(i)))

writeln("Valen\_Buy is: ", getsol(Valen\_Buy(i)))

writeln("------")

END-DO

end-model