

# ALGO Projet AMPL

*Production de nourriture*

*Binôme :*  
**Abdurrahman Yasar**  
**Djebien Tarik**  
*Groupe : 2*  
*Date: Novembre 2010*

# 1. Le problème

De la nourriture est produite en raffinant et en mélangeant plusieurs types d'huiles brutes. Les huiles brutes appartiennent aux catégories suivantes :

huiles végétales    VEG1  
                               VEG2

huiles animales    ANI1  
                               ANI2  
                               ANI3

Les prix d'achat de ces huiles varient au cours des mois. Les voici en euros par tonne.

	VEG1	VEG2	ANI1	ANI2	ANI3
janvier	110	120	130	110	115
février	130	130	110	90	115
mars	110	140	130	100	95
avril	120	110	120	120	125
mai	100	120	150	110	105
juin	90	100	140	80	135

Figure 1.1

La nourriture produite est vendue à 150 euros la tonne.

Les huiles brutes sont raffinées dans des ateliers distincts. On ne peut raffiner au maximum que 200 tonnes d'huiles végétales et 250 tonnes d'huiles animales par mois. Il n'y a pas de perte de poids lors du raffinage. Le coût du raffinage est négligeable.

On peut stocker jusqu'à 1000 tonnes de chaque huile pour usage futur. Le coût du stockage est estimé à 5 euros par tonne et par mois. La nourriture produite ne peut pas être stockée.

Il y a une limitation technologique sur ce qu'on appelle la « dureté » du produit final, qui doit être comprise entre 3 et 6. La dureté du produit final est égale à la moyenne des duretés des huiles brutes, pondérées par les quantités. Voici les duretés des huiles brutes.

VEG1	VEG2	ANI1	ANI2	ANI3
8.8	6.1	2.0	4.2	5.0

Figure 1.2

Aujourd'hui, les stocks sont de 500 tonnes d'huiles de chaque type. On souhaite disposer des mêmes stocks à la fin du mois de juin.

**Objectif:** *Quelle stratégie l'entreprise devrait-elle suivre pour maximiser son profit ?*

## 2. Modélisation

*Ce problème principal de production de nourriture se révèle être un ensemble de sous problèmes secondaires qui peut être résolu en utilisant la programmation linéaire.*

*Le problème présenté ici a trois aspects:*

- *Premièrement c'est un problème simple de mélange d'huiles de types différents.*
- *Deuxièmement il y a un problème de dureté et de stockage des huiles.*
- *Troisièmement il y a un problème d'évolution mensuel des prix d'achats des huiles.*

*Tout au long du sujet, nous admettrons par hypothèse qu'il n'y a pas de perte de poids lors du raffinage et que le coût du raffinage dans chacun des ateliers est négligeable.*

*Pour comprendre comment ce problème peut être formulé, il est commode de considérer d'abord le problème de mélange, raffinage des huiles, production de nourriture, dureté et stockage sur une période d'un mois, choisissons par exemple le mois de janvier.*

*Nous verrons ensuite dans la partie 3 une modélisation générale sur une période semestrielle.*

### **La Production de nourriture sur une durée d'un mois seul (exemple: Janvier):**

*L'objectif est de maximiser le bénéfice de l'entreprise caractérisé par:*

- *Soit  $\omega(\max)$  le nom de l'objectif de maximisation du profit (en euros) de l'entreprise.*
  - *On le calcule à partir du revenu résultant de ces deux opérations :*
    - *la vente de nourriture produite (paramétré par son prix de vente, ici 150 euros).*
    - *A laquelle on soustrait le coût d'achat de chacune des huiles brutes nécessaire à la fabrication de la nourriture (paramétré par leur prix d'achats au cours du mois).*
- (voir Figure 1.1 ).*

*Soit  $(x_1, x_2, x_3, x_4, x_5) \in (\mathbb{R}^+)^5$  les variables qui représentent les quantités des huiles brutes qui devraient être achetées (en tonnes).*

*Respectivement, on a en fonction des huiles définies dans le sujet :*

- $x_1$  : *quantité de l'huile VEG1 achetée (en tonnes).*
- $x_2$  : *quantité de l'huile VEG2 achetée (en tonnes).*
- $x_3$  : *quantité de l'huile ANI1 achetée (en tonnes).*
- $x_4$  : *quantité de l'huile ANI2 achetée (en tonnes).*
- $x_5$  : *quantité de l'huile ANI3 achetée (en tonnes).*

*Soit  $Y \in \mathbb{R}^+$  la variable qui représente la quantité totale de nourriture produite (en tonnes) à partir des huiles brutes.*

*Respectivement, on a en fonction des huiles définies dans le sujet :*

- $y_1$  : *quantité de l'huile VEG1 utilisée pour produire la nourriture (en tonnes).*
- $y_2$  : *quantité de l'huile VEG2 utilisée pour produire la nourriture (en tonnes).*
- $y_3$  : *quantité de l'huile ANI1 utilisée pour produire la nourriture (en tonnes).*
- $y_4$  : *quantité de l'huile ANI2 utilisée pour produire la nourriture (en tonnes).*
- $y_5$  : *quantité de l'huile ANI3 utilisée pour produire la nourriture (en tonnes).*

*On a alors:*

$$Y = y_1 + y_2 + y_3 + y_4 + y_5$$

Si on n'a permis aucun stockage d'huiles brutes, le problème d'achat, de mélange et de production pour le mois de janvier pourrait être formulé par ce système linéaire PL1 :

$$\begin{array}{rcl}
 -110x_1 - 120x_2 - 130x_3 - 110x_4 - 115x_5 + 150Y & = & \omega(\max) \\
 x_1 + x_2 & \leq & 200 \\
 x_3 + x_4 + x_5 & \leq & 250 \\
 x_1 + x_2 + x_3 + x_4 + x_5 - Y & = & 0
 \end{array}
 \quad (PL1):$$

Les prix de vente en euros par tonne des huiles pour le mois de janvier sont indiqués dans le tableau Figure 1.1 .

Les deux premières contraintes représentent les capacités limitées pour raffiner les Huiles végétales et animales. En effet, on ne peut raffiner au maximum que 200 tonnes d'huiles végétales et 250 tonnes d'huiles animales par mois .

Finalement, il est nécessaire de s'assurer que le poids en tonnes de la production de nourriture finale est égal au poids en tonnes des huiles brutes utilisées pour sa fabrication.

$$(Equation\ de\ continuité): \quad x_1 + x_2 + x_3 + x_4 + x_5 = Y$$

C'est fait par la dernière contrainte qui impose la continuité du poids en tonne avant et après le raffinage et mélange des huiles brutes pour la fabrication de la nourriture.

### **La Production de nourriture sur une durée d'un mois seul** **avec la « dureté » du produit final** **(exemple: Janvier):**

D'après le sujet, la «dureté» de la nourriture doit être bornée entre sa limite supérieure de 6 et sa limite inférieure de 3.

La dureté de la nourriture est égale à la moyenne des duretés des huiles brutes, pondérées par les quantités. (Voir les duretés des huiles brutes Figure 1.2 ).

Les deux inéquations suivantes imposent la limitation technologique sur la dureté de la nourriture produite à partir de la dureté de chacune des huiles intervenant dans la fabrication de la nourriture.

$$\begin{array}{l}
 \text{Limitation technologique supérieure :} \\
 ( 8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5 ) \leq 6Y \\
 \text{Limitation technologique inférieure :} \\
 ( 8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5 ) \geq 3Y
 \end{array}$$

On obtient alors un nouveau problème prenant en charge la dureté qui pourrait être formulé par ce système linéaire PL2:

$$\begin{array}{rcl}
 -110x_1 - 120x_2 - 130x_3 - 110x_4 - 115x_5 + 150Y & = & \omega(\max) \\
 x_1 + x_2 & \leq & 200 \\
 x_3 + x_4 + x_5 & \leq & 250 \\
 8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5 - 6Y & \leq & 0 \\
 8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5 - 3Y & \geq & 0 \\
 x_1 + x_2 + x_3 + x_4 + x_5 - Y & = & 0
 \end{array}
 \quad (PL2):$$

**La Production de nourriture sur une durée d'un mois seul avec la «dureté» et la gestion de stock des huiles brutes non utilisées**  
**(exemple: Janvier):**

Soit  $Z \in \mathbb{R}^+$  la variable qui représente la quantité totale des huiles stockées (en tonnes) à partir des huiles brutes non utilisées à la fin du mois.

Respectivement, on a en fonction des huiles définies dans le sujet :

- $z_1$  : quantité de l'huile VEG1 non utilisée et stockée pour le prochain mois (en tonnes).
- $z_2$  : quantité de l'huile VEG2 non utilisée et stockée pour le prochain mois (en tonnes).
- $z_3$  : quantité de l'huile ANI1 non utilisée et stockée pour le prochain mois (en tonnes).
- $z_4$  : quantité de l'huile ANI2 non utilisée et stockée pour le prochain mois (en tonnes).
- $z_5$  : quantité de l'huile ANI3 non utilisée et stockée pour le prochain mois (en tonnes).

On a alors:

$$Z = z_1 + z_2 + z_3 + z_4 + z_5$$

Maintenant que l'on a autorisé le stockage des huiles brutes non utilisée à la fin du mois, pour une future utilisation au mois prochain, on se retrouve dans l'obligation de modifier notre objectif principal  $\omega(\max)$  .

Le nouvel objectif est de maximiser le bénéfice de l'entreprise caractérisé par:

- Soit  $\Omega(\max)$  le nom du nouvel objectif de maximisation du profit (en euros) de l'entreprise.
- On le calcule à partir du revenu résultant de ces trois opérations :
  - la vente de nourriture produite (paramétré par son prix de vente, ici 150 euros).
  - A laquelle on soustrait le coût d'achat de chacune des huiles brutes nécessaire à la fabrication de la nourriture (paramétré par leur prix d'achats au cours du mois). (voir Figure 1.1 ).
  - A laquelle on soustrait également le coût de stockage de chacune des huiles brutes en fonction de leur quantités (paramétré par le prix de stockage qui est estimé dans le sujet à 5 euros par tonne et par mois).
- En résumé, on a:

$$\Omega(\max) = \omega(\max) - 5 * Z$$

Dans le sujet, il est dit «Aujourd'hui, les stocks sont de 500 tonnes d'huiles de chaque type ».

Supposons que «Aujourd'hui» est le premier jour du mois de Janvier.

On a alors:

$$z_1 = z_2 = z_3 = z_4 = z_5 = 500$$

Introduisons cette notation pour nous situer dans le temps à un mois  $m$  donné:

- $(\dots)_1 \stackrel{\text{def}}{=} \text{Au cours du premier mois } m \in \text{MOIS de production de nourriture.}$
- $(\dots)_m \stackrel{\text{def}}{=} \text{Au cours du mois } m \in \text{MOIS de production de nourriture.}$
- $(\dots)_{m+1} \stackrel{\text{def}}{=} \text{Au cours du mois suivant de } m \in \text{MOIS de production de nourriture.}$

On obtient alors le problème mensuel final prenant en charge la dureté et le stockage qui pourrait être formulé par ce système linéaire (PL3):

$$\begin{array}{rcl}
-110(x_1)_m - 120(x_2)_m - 130(x_3)_m - 110(x_4)_m - 115(x_5)_m + 150(Y)_m - 5(Z)_m & = & \Omega(max) \\
(x_1)_m + (x_2)_m & \leq & 200 \\
(x_3)_m + (x_4)_m + (x_5)_m & \leq & 250 \\
8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 6(Y)_m & \leq & 0 \\
8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 3(Y)_m & \geq & 0 \\
(x_1)_m + (x_2)_m + (x_3)_m + (x_4)_m + (x_5)_m + (Z)_m & = & (Y)_m + (Z)_{m+1}
\end{array}$$

**La Production de nourriture sur une durée de six mois avec la gestion de la «dureté», des stock et de l'inflation des prix des huiles brutes (exemple: période de janvier à juin):**

La stratégie adoptée pour la résolution du problème de période mensuel précédente serait adaptable pour chaque mois inclus dans un calendrier débutant au mois de janvier jusqu'au mois de juin en ajoutant quelques modifications.

Introduisons deux ensembles:

Soit  $HUILE = \{VEG1, VEG2, ANI1, ANI2, ANI3\}$ ,  
l'ensemble ordonné des huiles brutes disponibles.

Soit  $MOIS = \{n \in \llbracket 1, 6 \rrbracket \mid (\forall n_1, n_2 \in MOIS, n_1 \neq n_2) \wedge (n=1 \rightarrow \text{janvier}, \dots, n=6 \rightarrow \text{juin})\}$ ,  
l'ensemble ordonné décrivant les mois de la période de production de nourriture.

Les premiers changements que l'on doit effectuer sont les coefficients intervenants dans l'objectif de notre programme linéaire (PL3) associant les prix d'achats des huiles brutes en fonction du mois d'achats ( Figure 1.1 ).

On peut décrire la Figure 1.1 par la matrice suivante:

$$A = \begin{array}{ccccc} & VEG1 & VEG2 & ANI1 & ANI2 & ANI3 \\ \begin{array}{c} 110 \\ 130 \\ 110 \\ 10 \\ 100 \\ 90 \end{array} & \begin{array}{c} 120 \\ 130 \\ 140 \\ 110 \\ 120 \\ 100 \end{array} & \begin{array}{c} 130 \\ 110 \\ 130 \\ 120 \\ 150 \\ 140 \end{array} & \begin{array}{c} 110 \\ 90 \\ 100 \\ 120 \\ 110 \\ 80 \end{array} & \begin{array}{c} 115 \\ 115 \\ 95 \\ 125 \\ 105 \\ 35 \end{array} & \begin{array}{c} \text{janvier} \\ \text{février} \\ \text{mars} \\ \text{avril} \\ \text{mai} \\ \text{juin} \end{array}
\end{array}, A \in M_{\text{card}(MOIS) \times \text{card}(HUILE)}(\mathbb{N}),$$

On définit  $(a_{i,j}) \stackrel{\text{def}}{=} \text{le prix d'achat de l'huile brute de la colonne } j \text{ au mois de la ligne } i.$

Notre nouvel objectif est de maximiser le bénéfice de l'entreprise sur les six mois, il est donc caractérisé par:

- Soit  $\psi(max)$  le nom du nouvel objectif de maximisation du profit (en euros) de l'entreprise sur une période de six mois.
- On le calcule à partir du revenu résultant de ces trois opérations :
  - la vente de nourriture produite (paramétré par son prix de vente, ici 150 euros).
  - A laquelle on soustrait le coût d'achat de chacune des huiles brutes nécessaire à la fabrication de la nourriture (paramétré par leur prix d'achats  $(a_{i,j}) \in A$ ).
  - A laquelle on soustrait également le coût de stockage de chacune des huiles brutes en fonction de leur quantités (paramétré par le prix de stockage qui est estimé dans le sujet à 5 euros par tonne et par mois pour chaque mois).
- En résumé, on a:

$$\psi(max) = \sum_{m \in MOIS} (\Omega(max))_m$$

On obtient alors le problème semestrielle prenant en charge la dureté et le stockage qui pourrait être formulé par ce système linéaire (PL4):

$$\begin{aligned} \sum_{m \in MOIS} [-a_{m,1}(x_1)_m - a_{m,2}(x_2)_m - a_{m,3}(x_3)_m - a_{m,4}(x_4)_m - a_{m,5}(x_5)_m + 150(Y)_m - 5(Z)_m] &= \psi(max) \\ \forall m \in MOIS, (x_1)_m + (x_2)_m &\leq 200 \\ \forall m \in MOIS, (x_3)_m + (x_4)_m + (x_5)_m &\leq 250 \\ \forall m \in MOIS, 8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 6(Y)_m &\leq 0 \\ \forall m \in MOIS, 8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 3(Y)_m &\geq 0 \\ \forall i \in \llbracket 1, card(HUILE) \rrbracket, (z_i)_1 &= 500 \\ \forall i \in \llbracket 1, card(HUILE) \rrbracket, (z_i)_{card(MOIS)+1} &= 500 \\ \forall m \in MOIS, \forall i \in \llbracket 1, card(HUILE) \rrbracket, (z_i)_m &\leq 1000 \\ \forall m \in MOIS, (x_1)_m + (x_2)_m + (x_3)_m + (x_4)_m + (x_5)_m + (Z)_m &= (Y)_m + (Z)_{m+1} \end{aligned}$$

Pour terminer, nous devons prendre en compte l'évolution des prix d'achats des huiles brutes.

En effet, sur une période de plusieurs mois, les prix d'achats des huiles brutes évoluent selon un taux d'inflation décrit par le tableau suivant:

	février	mars	avril	mai	juin
huiles végétales	$x\%$	$2x\%$	$3x\%$	$4x\%$	$5x\%$
huiles animales	$2x\%$	$4x\%$	$6x\%$	$8x\%$	$10x\%$

Figure 3.1

Ici, les changements que l'on doit effectuer sont les coefficients  $(a_{i,j})$  intervenants dans l'objectif de notre programme linéaire (PL4) associant les prix d'achats des huiles brutes en fonction du mois d'achats.

En effet, on doit maintenant augmenter ce prix de base d'un pourcentage dépendant de deux facteurs:

- une constante d'inflation selon le mois d'achat de l'huile brute.( Figure 3.1 )
- un taux d'inflation paramétré ici par un entier  $x$ .

On peut décrire la Figure 3.1 par la matrice suivante:

$$B = \begin{matrix} & \begin{matrix} VEG1 & VEG2 & ANI1 & ANI2 & ANI3 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 2 & 2 \\ 2 & 2 & 4 & 4 & 4 \\ 3 & 3 & 6 & 6 & 6 \\ 4 & 4 & 8 & 8 & 8 \\ 5 & 5 & 10 & 10 & 10 \end{bmatrix} & \begin{matrix} \text{janvier} \\ \text{février} \\ \text{mars} \\ \text{avril} \\ \text{mai} \\ \text{juin} \end{matrix} \end{matrix}, \quad B \in M_{\text{card(MOIS)} \times \text{card(HUILE)}}(\mathbb{N}),$$

On définit  $(b_{i,j})_{\text{def}}$  la constante d'inflation sur le prix de l'huile brute de la colonne  $j$  au mois de la ligne  $i$ .

On cherche maintenant le prix final d'achat des huiles brutes donné par la formule suivante:

$$\text{prix}_{\text{final}} = \text{prix}_{\text{base}} + \left\{ \frac{\text{prix}_{\text{base}}}{100} \right\} * (\text{constante}_{\text{inflation}} * x)$$

On peut décrire le prix final d'achat des huiles brutes par la matrice suivante:

$$C = \begin{matrix} & \begin{matrix} VEG1 & VEG2 & ANI1 & ANI2 & ANI3 \end{matrix} \\ \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & (c_{i,j}) & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} & \begin{matrix} \text{janvier} \\ \text{février} \\ \text{mars} \\ \text{avril} \\ \text{mai} \\ \text{juin} \end{matrix} \end{matrix}, \quad C \in M_{\text{card(MOIS)} \times \text{card(HUILE)}}(\mathbb{R}^+),$$

$$\text{On définit } (c_{i,j})_{\text{def}} = (a_{i,j}) + \frac{[(a_{i,j})(b_{i,j})x]}{100}$$

, le prix final de l'huile brute de la colonne  $j$  au mois de la ligne  $i$ .



Notre objectif final est de maximiser le bénéfice de l'entreprise sur les six mois en tenant compte de l'évolution des prix d'achats des huiles brutes, il est donc caractérisé par:

- Soit  $\Theta(\max)$  le nom du nouvel objectif de maximisation du profit (en euros) de l'entreprise sur une période de six mois avec évolution des prix de ventes des huiles brutes.
- On le calcule à partir du revenu résultant de ces trois opérations :
  - la vente de nourriture produite (paramétré par son prix de vente, ici 150 euros).
  - A laquelle on soustrait le coût d'achat final prenant en considération l'inflation du prix d'achat au cours des mois de chacune des huiles brutes nécessaire à la fabrication de la nourriture (paramétré par leur prix d'achats final  $(c_{i,j}) \in C$ ).
  - A laquelle on soustrait également le coût de stockage de chacune des huiles brutes en fonction de leur quantités (paramétré par le prix de stockage qui est estimé dans le sujet à 5 euros par tonne et par mois pour chaque mois).

On obtient alors le problème semestrielle final prenant en charge la dureté et le stockage et l'inflation des prix qui pourrait être formulé par ce système linéaire  $(PL_{final})$  :

$$\begin{aligned}
 \sum_{m \in MOIS} [-c_{m,1}(x_1)_m - c_{m,2}(x_2)_m - c_{m,3}(x_3)_m - c_{m,4}(x_4)_m - c_{m,5}(x_5)_m + 150(Y)_m - 5(Z)_m] &= \Theta(\max) \\
 \forall m \in MOIS, \quad (x_1)_m + (x_2)_m &\leq 200 \\
 \forall m \in MOIS, \quad (x_3)_m + (x_4)_m + (x_5)_m &\leq 250 \\
 \forall m \in MOIS, \quad 8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 6(Y)_m &\leq 0 \\
 \forall m \in MOIS, \quad 8.8(x_1)_m + 6.1(x_2)_m + 2(x_3)_m + 4.2(x_4)_m + 5(x_5)_m - 3(Y)_m &\geq 0 \\
 \forall i \in \llbracket 1, \text{card}(HUILE) \rrbracket, \quad (z_i)_1 &= 500 \\
 \forall i \in \llbracket 1, \text{card}(HUILE) \rrbracket, \quad (z_i)_{\text{card}(MOIS)+1} &= 500 \\
 \forall m \in MOIS, \forall i \in \llbracket 1, \text{card}(HUILE) \rrbracket, \quad (z_i)_m &\leq 1000 \\
 \forall m \in MOIS, \quad (x_1)_m + (x_2)_m + (x_3)_m + (x_4)_m + (x_5)_m + (Z)_m &= (Y)_m + (Z)_{m+1}
 \end{aligned}$$

### 3. Réalisation en AMPL

*Dans cette partie nous allons décrire formellement les données mathématiques nécessaires à la réalisation du problème de fabrication de nourriture en AMPL, en apportant le code AMPL associé à chacune des définitions.*

*(On retrouvera l'ensemble du code AMPL commenté dans la partie Annexe)*

#### Les Ensembles:

*Soit  $VEG = \{ VEG1, VEG2 \}$  l'ensemble décrivant les huiles végétales brutes.*

*Soit  $ANI = \{ ANI1, ANI2, ANI3 \}$  l'ensemble décrivant les huiles animales brutes.*

*Soit  $HUILE = VEG \cup ANI$ , l'ensemble des huiles brutes.*

*Soit  $MOIS = \{ n \in \llbracket 1, N \rrbracket \mid (\forall n_1, n_2 \in MOIS, n_1 \neq n_2) \wedge (n=1 \Leftrightarrow n \sim \text{Janvier}, \dots, n=12 \Leftrightarrow n \sim \text{Décembre}) \}$  l'ensemble décrivant les mois d'une période de production de nourriture.*

#### Code AMPL:

```
set VEG;  
set ANI;  
set HUILE := VEG union ANI;  
set MOIS := 1 .. N;
```

#### Les paramètres:

- *Soit  $N \in \mathbb{N}$ ,  $N = \text{Card}(MOIS)$  un paramètre indiquant le nombre de mois de production de nourriture.*
- *Soit  $\text{raf\_max\_veg} \in \mathbb{N}$ , un paramètre indiquant le nombre de tonnes d'huiles végétales maximales que l'on peut raffiner dans un mois.*
- *Soit  $\text{raf\_max\_ani} \in \mathbb{N}$ , un paramètre indiquant le nombre de tonnes d'huiles animales maximales que l'on peut raffiner dans un mois.*
- *Soit  $\text{durete\_min} \in \mathbb{N}$ , un paramètre indiquant la « dureté » minimale du produit final*
- *Soit  $\text{durete\_max} \in \mathbb{N}$ , un paramètre indiquant la « dureté » maximale du produit final*
- *On a:  $\text{durete\_min} < \text{durete\_max}$ .*
- *Soit  $\text{st\_max} \in \mathbb{N}$ , un paramètre indiquant le stock maximal en tonnes de chaque huile pour usage future.*
- *Soit  $\text{st\_initial} \in \mathbb{N}$ , un paramètre indiquant le stock initial en tonnes de chaque huile au début du premier mois de production de nourriture.*
- *Soit  $\text{st\_final} \in \mathbb{N}$ , un paramètre indiquant le stock final en tonnes de chaque huile à la fin du dernier mois de production de nourriture.*

- Soit  $\text{prix\_stockage\_mensuel} \in \mathbb{N}$  , un paramètre indiquant le coût de stockage en euros par tonne et par mois.
- Soit  $\text{taux\_inflation} \in \mathbb{N}$  , un paramètre indiquant le taux d'inflation des prix d'achats des huiles prévisionnels pour les mois prochain.
- Soit  $\text{prix\_mensuel} \in M_{N, \text{Card}(\text{HUILE})}(\mathbb{N})$ , la matrice décrivant les prix des huiles brutes en fonction des mois. (voir Figure 1.1)
- Soit  $\text{durete} \in V_{\text{Card}(\text{HUILE}), 1}(\mathbb{R}^+)$ , le vecteur décrivant les dureté des huiles brutes en fonction des types d'huiles. (voir Figure 1.2)
- Soit  $\text{cst\_inflation} \in M_{N, \text{Card}(\text{HUILE})}(\mathbb{N})$ , la matrice décrivant l'évolution des prix des huiles brutes en fonction des mois. (voir Figure 3.1)

### Code AMPL:

```

param N integer >=0;

param raf_max_veg >=0;

param raf_max_ani >=0;

param durete_min;

param durete_max;

param st_max;

param st_initial;

param st_final;

param prix_stockage_mensuel;

param taux_inflation;

param prix_mensuel {MOIS, HUILE} >=0;

param durete {HUILE};

param cst_inflation {MOIS, HUILE};

```

### Les Variables:

Les variables  $x_i$ ,  $y_i$  et  $z_i$  sont définies dans la partie précédente **Modélisation**.

- 

Soit  $\text{qte\_achete} : \text{MOIS} \times \text{HUILE} \rightarrow \text{TONNES}$   
 $(m, h) \rightarrow x_i$

- 

que l'on définit par la quantité en tonnes d'huile 'h' achetée au mois 'm'.

Soit  $\text{qte\_utilise} : \text{MOIS} \times \text{HUILE} \rightarrow \text{TONNES}$   
 $(m, h) \rightarrow y_i$

- 

que l'on définit par la quantité en tonnes d'huile 'h' utilisée pour produire de la nourriture au mois 'm'.

Soit  $\text{qte\_stock} : \{1, \dots, N+1\} \times \text{HUILE} \rightarrow \text{TONNES}$   
 $(m, h) \rightarrow z_i$

- 

que l'on définit par la quantité en tonnes d'huile 'h' non utilisée et stockée à la fin du mois 'm'.

On définit aussi ces 2 applications qui ne serviront que dans la description formelle du modèle, elles sont inexistantes dans la réalisation en AMPL:

- 

Soit  $\text{ord} : \text{HUILE} \rightarrow \llbracket 1, \text{Card}(\text{HUILE}) \rrbracket$   
 $h \rightarrow 1 \text{ si } h = \text{VEG1}, \dots, 5 \text{ si } h = \text{ANI3}$   
que l'on définit par la position de l'huile 'h' dans l'ensemble ordonné HUILE.

- 

Soit  $\text{durete} : \text{HUILE} \rightarrow \mathbb{R}^+$   
 $h \rightarrow (\text{durete}_{\text{ord}(h), 1})$   
que l'on définit par la valeur de la dureté de l'huile 'h' dans le vecteur durete.

### Code AMPL:

```
var qte_achete {MOIS, HUILE} >= 0;  
var qte_utilise {MOIS, HUILE} >= 0 ;  
var qte_stock {1 .. N+1, HUILE} >= 0;
```

## Objectif

Nous avons vu précédemment dans la partie modélisation que notre objectif final était sous la forme :

$$\sum_{m \in \text{MOIS}} [-c_{m,1}(x_1)_m - c_{m,2}(x_2)_m - c_{m,3}(x_3)_m - c_{m,4}(x_4)_m - c_{m,5}(x_5)_m + 150(Y)_m - 5(Z)_m] = \Theta(\max)$$

Si l'on généralise sur une période de plusieurs mois, on obtient en notation formelle :

$$P(m, h) = -\text{prix\_mensuel}(m, h) * ((100 + \text{cst\_inflation}(m, h) * \text{taux\_inflation}) / 100) * \text{qte\_achete}(m, h)$$

$$U(m, h) = 150 * \text{qte\_utilise}(m, h)$$

$$S(m, h) = -\text{prix\_stockage\_mensuel} * \text{qte\_stock}(m, h)$$

Notre objectif s'écrit alors :

$$\sum_{m \in \text{MOIS}, h \in \text{HUILE}} (P(m, h) + U(m, h) + S(m, h)) = \Theta(\max)$$

## Code AMPL:

**maximize** profit:

*sum {m in MOIS, h in HUILE}*

*(150\*qte\_utilise [m,h] - prix\_mensuel [m,h] \* (100 + cst\_inflation [m,h] \* taux\_inflation)/100) \* qte\_achte [m,h] - prix\_stockage\_mensuel\*qte\_stock [m,h])*

;

## Les Contraintes

### Limitation du raffinage des huiles brutes:

Nous avons vu précédemment dans la partie modélisation que les capacités pour raffiner les Huiles végétales et animales sont limitées.

$$\begin{aligned}x_1 + x_2 &\leq 200 \\x_3 + x_4 + x_5 &\leq 250\end{aligned}$$

Si l'on généralise sur une période de plusieurs mois, on obtient en notation formelle:

$$\begin{aligned}\forall m \in \text{MOIS}, \sum_{v \in \text{VEG}} \text{qte\_utilise}(m, v) &\leq \text{raf\_max\_veg} \\ \forall m \in \text{MOIS}, \sum_{a \in \text{ANI}} \text{qte\_utilise}(m, a) &\leq \text{raf\_max\_ani}\end{aligned}$$

### Code AMPL:

**subject to** utilisation\_limite\_veg\_max {m in MOIS}:  
sum {v in VEG} qte\_utilise [m,v] <= raf\_max\_veg;

**subject to** utilisation\_limite\_ani\_max {m in MOIS}:  
sum {a in ANI} qte\_utilise [m,a] <= raf\_max\_ani;

### Limitation technologique de la dureté du produit final:

Nous avons vu précédemment que:

$$\begin{aligned}\text{Limitation technologique supérieure :} \\ (8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5) &\leq 6Y \\ \text{Limitation technologique inférieure :} \\ (8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5) &\geq 3Y\end{aligned}$$

On peut réécrire ces deux inéquations sous cette forme:

$$\begin{aligned}\text{Limitation technologique supérieure et inférieure :} \\ 3Y \leq (8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5) &\leq 6Y \\ \text{Or d'après l'équation de continuité on a :} \\ 3 \leq \frac{(8.8x_1 + 6.1x_2 + 2x_3 + 4.2x_4 + 5x_5)}{(x_1 + x_2 + x_3 + x_4 + x_5)} &\leq 6, \quad \text{car } Y \geq 0\end{aligned}$$

Finalement on généralise sur une période de plusieurs mois et on obtient en notation formelle :

$$\forall m \in \text{MOIS}, \text{durete\_min} \leq \frac{\sum_{h \in \text{HUILE}} (\text{qte\_utilise}(m, h) * \text{durete}(h))}{\sum_{h \in \text{HUILE}} \text{qte\_utilise}(m, h)} \leq \text{durete\_max}$$

### Code AMPL:

**subject to** durete\_final\_min {m in MOIS}:

$$\sum \{h \text{ in } HUILE\} (qte\_utilise [m,h] * durete [h]) \geq \sum \{h \text{ in } HUILE\} (durete\_min * qte\_utilise [m,h]);$$

**subject to** durete\_final\_max {m in MOIS}:

$$\sum \{h \text{ in } HUILE\} (qte\_utilise [m,h] * durete [h]) \leq \sum \{h \text{ in } HUILE\} (durete\_max * qte\_utilise [m,h]);$$

### Stockage des huiles brutes:

*Nous avons besoin de définir 3 nouvelles contraintes pour la gestion du stockage des huiles brutes.*

*Une contrainte sur le stock initial:*

$$\forall h \in HUILE, qte\_stock(1,h) = st\_initial$$

*Une contrainte sur le stock final:*

$$\forall h \in HUILE, qte\_stock(N+1,h) = st\_final$$

*Une contrainte sur le stock maximal:*

$$\forall (h, m) \in HUILE \times MOIS, qte\_stock(m, h) \leq st\_max$$

*Nous devons aussi modifier notre équation de continuité en lui rajoutant la gestion du stockage:*

*Pour gérer le stockage d'huile non utilisée d'un mois m au mois suivant m+1 on doit respecter ceci:*

*(Equation de continuité + gestion de stockage):*

$$(x_1 + x_2 + x_3 + x_4 + x_5)_m + (z_1 + z_2 + z_3 + z_4 + z_5)_m = (y_1 + y_2 + y_3 + y_4 + y_5)_m + (z_1 + z_2 + z_3 + z_4 + z_5)_{m+1}$$

$$\Leftrightarrow \left(\sum_{i=1}^5 x_i\right)_m + \left(\sum_{i=1}^5 z_i\right)_m = \left(\sum_{i=1}^5 y_i\right)_m + \left(\sum_{i=1}^5 z_i\right)_{m+1}$$

*Finalement, on peut généraliser cette égalité sur tous les mois et nous obtenons cette contrainte:*

*(Equation d'équilibre):*

$$\forall (m, h) \in MOIS \times HUILE, qte\_stock(m, h) + qte\_achete(m, h) = qte\_utilise(m, h) + qte\_stock(m+1, h)$$

### Code AMPL:

**subject to** stock\_initial {h in HUILE} : qte\_stock [1,h] = st\_initial;

**subject to** stock\_final {h in HUILE} : qte\_stock [N+1,h] = st\_final;

**subject to** stock\_max {m in MOIS, h in HUILE} : qte\_stock [m,h] <= st\_max;

**subject to** equilibre {m in MOIS, h in HUILE} : qte\_stock [m,h] + qte\_achte [m,h] = qte\_stock [m+1,h] + qte\_utilise [m,h];

## Les Données du Problème

```
data;
/* Ensemble des huiles végétales et animales */
set VEG := VEG1 VEG2;
set ANI := ANI1 ANI2 ANI3;
/* période de production de nourriture de janvier à juin */
param N := 6;
/* Matrice A des prix d'achat des huiles brutes en euros */
param prix_mensuel :
      VEG1  VEG2  ANI1  ANI2  ANI3 :=
1      110    120    130    110    115
2      130    130    110     90    115
3      110    140    130    100     95
4      120    110    120    120    125
5      100    120    150    110    105
6       90    100    140     80    135;
/* Limitation de raffinage des huiles végétales et animales dans un atelier en tonnes */
param raf_max_veg := 200;
param raf_max_animal := 250;
/* La dureté minimale et maximale du produit final */
param durete_min := 3;
param durete_max := 6;
param durete :=
VEG1  8.8
VEG2  6.1
ANI1  2.0
ANI2  4.2
ANI3  5.0;

param st_max := 1000 ; /* stock maximal des huiles dans un mois en tonnes */
param st_initial := 500 ; /* stock initial de chaque huiles au début du premier mois de production en tonnes */
param st_final := 500 ; /* stock final de chaque huiles après la fin du dernier mois de production en tonnes */

/* Matrice B */
param est_inflation :
      VEG1  VEG2  ANI1  ANI2  ANI3 :=
1       0     0     0     0     0
2       1     1     2     2     2
3       2     2     4     4     4
4       3     3     6     6     6
5       4     4     8     8     8
6       5     5    10    10    10;

param prix_stockage_mensuel := 5 ; /* le cout de stockage des huiles non utilisées à la fin du mois en euros */

/* le taux d'inflation noté x dans le sujet et dans la partie modélisation */
param taux_inflation := 5 ; /* 0 ou 5 selon la trace d'exécution dans l'annexe */
```

## Résolution

```
/* pour résoudre en nombres entiers */
option solver cplex;
solve;
```

Voir l'annexe pour les traces d'exécution du programme AMPL avec un taux d'inflation égal à zéro ou à cinq.



## 4. Variantes

### Variante 1:

*la nourriture produite doit être obtenue en mélangeant au plus trois types d'huiles brutes différentes .*

On introduit une variable booléenne ainsi que des contraintes supplémentaires:

Soit  $\text{est\_utilise} : \text{MOIS} \times \text{HUILE} \rightarrow \text{BOOLEEN}$   
 $(m, h) \rightarrow \begin{cases} 1 & \text{si } h \text{ est utilisé dans le mois } m \\ 0 & \text{sinon} \end{cases}$

- un prédicat que l'on définit pour déterminé si une huile brute 'h' est utilisé pour produire de la nourriture au mois 'm'.

Ici nous sommes confronté à un nouveau problème, nous devons trouver des inéquations satisfaisant le résultat du prédicat  $\text{est\_utilise}$ .

Sachant que dans le sujet, il existe deux types d'huiles brutes différentes ( animales et végétales), le nombre d'inéquations recherchées s'élèvent à 2. Elles dépendent des paramètres  $\text{raf\_max\_veg}$  et  $\text{raf\_max\_ani}$ .

L'astuce ici est d'utiliser le fait que si l'huile n'est pas utilisé au cours du mois alors la quantité utilisé doit être nécessairement nulle sinon elle doit être majorée par  $\text{raf\_max\_veg}$  ou  $\text{raf\_max\_ani}$  selon son type.

On obtient alors les deux inéquations suivantes:

$$\forall (m, h) \in \text{MOIS} \times \text{VEG}, \text{qte\_utilise}(m, h) \leq \text{raf\_max\_veg} * \text{est\_utilise}(m, h)$$

et

$$\forall (m, h) \in \text{MOIS} \times \text{ANI}, \text{qte\_utilise}(m, h) \leq \text{raf\_max\_ani} * \text{est\_utilise}(m, h)$$

En effet, on peut vérifier que tous les cas possibles fonctionnent:

Soit  $h \in \text{VEG}$ ,

$(h \text{ est utilisée au mois } m) \Rightarrow (\text{est\_utilise}(m, h) = 1) \Rightarrow (\text{qte\_utilise}(m, h) \leq \text{raf\_max\_veg})$

$(h \text{ n'est pas utilisée au mois } m) \Rightarrow (\text{est\_utilise}(m, h) = 0) \Rightarrow (\text{qte\_utilise}(m, h) \leq 0)$

or  $\text{qte\_utilise}(m, h) \in \mathbb{R}^+ \Rightarrow \text{qte\_utilise}(m, h) = 0$ .

Même raisonnement si  $h \in \text{ANI}$ , on doit juste changer  $\text{raf\_max\_veg}$  par  $\text{raf\_max\_ani}$ .

Finalement, on doit limiter le mélange des huiles brutes nécessaires à la production de nourriture :

On ajoute un paramètre:

- Soit  $\text{melange\_max} \in \mathbb{N}$ , un paramètre indiquant le nombre maximal de mélange possible d'huile différentes pour produire la nourriture.

On ajoute la contrainte supplémentaire de limitation des mélanges d'huiles brutes:

- $\forall m \in \text{MOIS}, \sum_{h \in \text{HUILE}} \text{est\_utilise}(m, h) \leq \text{melange\_max}$

**Variante 2:**

***Si un type d'huile est utilisé un certain mois alors au moins 20 tonnes doivent l'être***

*.Ici nous sommes confronté à un nouveau problème, nous devons trouver des inéquations satisfaisant la condition de la variante numéro 2.*

*Sachant que dans le sujet, il existe deux types d'huiles brutes différentes ( animales et végétales), le nombre d'inéquations recherchées s'élèvent à 2.*

*On ajoute un paramètre:*

- *Soit  $\text{prod\_min} \in \mathbb{N}$  , un paramètre indiquant le nombre minimal de tonnes de nourriture à produire si un type d ' huile est utilisé.*

*L'astuce ici est d'utiliser le fait que si l'huile n'est pas utilisé au cours du mois alors la quantité utilisé doit être nécessairement nulle sinon elle doit être minorée par notre paramètre  $\text{prod\_min}$ .*

*On obtient alors les deux inéquations suivantes:*

$$\forall (m, h) \in \text{MOIS} \times \text{VEG}, \text{qte\_utilise}(m, h) \geq \text{prod\_min} * \text{est\_utilise}(m, h)$$

*et*

$$\forall (m, h) \in \text{MOIS} \times \text{ANI}, \text{qte\_utilise}(m, h) \geq \text{prod\_min} * \text{est\_utilise}(m, h)$$

*En effet, on peut vérifier que tous les cas possibles fonctionnent:*

*Soit  $h \in \text{VEG}$  ,*

*$(h \text{ est utilisée au mois } m) \Rightarrow (\text{est\_utilise}(m, h) = 1) \Rightarrow (\text{qte\_utilise}(m, h) \geq \text{prod\_min})$*

*$(h \text{ n'est pas utilisée au mois } m) \Rightarrow (\text{est\_utilise}(m, h) = 0) \Rightarrow (\text{qte\_utilise}(m, h) \geq 0)$*

*mais d ' après la variante 1 on a :*

*$\text{qte\_utilise}(m, h) \leq \text{raf\_max\_veg} * \text{est\_utilise}(m, h)$*

*nous somme dans le cas ou  $h$  n' est pas utilisée donc on a :*

*$\text{qte\_utilise}(m, h) \leq 0$*

*Finalement , on a :*

*$(\text{qte\_utilise}(m, h) \leq 0) \wedge (\text{qte\_utilise}(m, h) \geq 0) \Rightarrow \text{qte\_utilise}(m, h) = 0.$*

*Même raisonnement si  $h \in \text{ANI}$  , on dois juste changer  $\text{raf\_max\_veg}$  par  $\text{raf\_max\_ani}$  .*

**Variante 3:**

**Si VEG1 et VEG2 sont utilisées alors ANI3 doit l'être aussi.**

L'astuce ici est d'analyser ce problème par une approche d'algèbre de Boole en utilisant uniquement le prédicat  $\text{est\_utilise}(m, h)$ .

**Notation:**

Uniquement dans ce tableau, pour simplifier la syntaxe on dira que:  $\text{est\_utilise}(m, h) \Leftrightarrow h$

On a :

<i>VEG1</i>	<i>VEG2</i>	<i>VEG1</i> $\vee$ <i>VEG2</i>	<i>ANI3</i>
0	0	0	{0,1}
1	0	1	{0,1}
0	1	1	{0,1}
1	1	1	1

Modélisation du **OU** logique

<i>VEG1</i>	<i>VEG2</i>	<i>VEG1</i> $\wedge$ <i>VEG2</i>	<i>ANI3</i>
0	0	0	{0,1}
1	0	0	{0,1}
0	1	0	{0,1}
1	1	1	1

Modélisation du **ET** logique

On a donc deux choix possibles:

$$\bullet \quad \forall m \in \text{MOIS}, \sum_{h \in \text{VEG}} \text{est\_utilise}(m, h) \leq 2 * \text{est\_utilise}(m, \text{ANI3}) \quad (i)$$

ou

$$\bullet \quad \forall m \in \text{MOIS}, \prod_{h \in \text{VEG}} \text{est\_utilise}(m, h) \leq 2 * \text{est\_utilise}(m, \text{ANI3}) \quad (ii)$$

Dans les 2 cas, l'huile animale ANI3 sera utilisé si VEG1 et VEG2 sont utilisées.

Mais le **OU** oblige ANI3 à être aussi utilisé si VEG1 ou VEG2 le sont contrairement au **ET**.

Le tableau suivant montre les objectifs avec le **OU** et le **ET** :

<i>Avec (i) on obtient(inflation = 0):</i>	<i>Avec (ii) on obtient(inflation = 0):</i>
ampl: model projet2.ampl ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal integer solution within mipgap or absmipgap; objective <b>104250</b> 16726 MIP simplex iterations 4150 branch-and-bound nodes absmipgap = 6.40722, relmipgap = 6.14602e-05 No basis. 5 flow-cover cuts 8 Gomory cuts	ampl: model projet2.ampl; ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal integer solution; objective <b>100278.7037</b> 1938 MIP simplex iterations 415 branch-and-bound nodes 8 flow-cover cuts 8 Gomory cuts 12 implied-bound cuts 5 zero-half cuts

<i>Avec (i) on obtient(inflation = 5):</i>	<i>Avec (ii) on obtient(inflation = 5):</i>
ampl: reset; ampl: model projet2.ampl; ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal integer solution; objective <b>35530</b> 283 MIP simplex iterations 60 branch-and-bound nodes 1 cover cut 14 flow-cover cuts 4 Gomory cuts 13 implied-bound cuts 1 zero-half cut	ampl: reset; ampl: model projet2.ampl; ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal integer solution; objective <b>39300</b> 1854 MIP simplex iterations 391 branch-and-bound nodes No basis. 4 flow-cover cuts 8 Gomory cuts

*On constate que avec le **OU** sans taux d'inflation, on obtient un bénéfice légèrement supérieure que avec le **ET**.*

*Mais avec un taux d'inflation  $x=5$ , on voit une amélioration beaucoup plus significative est avantageuse pour l'atelier de production de nourriture  $39300 - 35530 = 3770$  euros en plus de bénéfices.*

*C'est pourquoi on optera pour **la solution (ii)** qui est plus avantageuse sur **long terme**.*

## 5. Annexes

Code AMPL du problème de nourriture avec les variantes :

*/\*PARTIE DECLARATION \*/*

*/\* ensemble pour les huiles VEG1 VEG2 \*/*

**set** VEG ;

*/\* ensemble pour les huiles ANI1 ANI2 ANI3 \*/*

**set** ANI ;

*/\* ensemble des huiles brutes \*/*

**set** HUILE := VEG **union** ANI;

*/\* l'union des ensembles a pour avantage de réduire la quantité des données (paramètres et contraintes) \*/*

*/\* le nombres de mois \*/*

**param** N integer >=0;

*/\* ensemble pour les MOIS :*

*1 -> Janvier*

*2-> Février*

*...*

*n-> ..*

*12-> Décembre*

*\*/*

**set** MOIS := 1 .. N;

*/\* quantités maximums qu'on peut utiliser pour raffiner les huiles brutes pour produire la nourriture \*/*

**param** raf\_max\_veg >=0; */\* pour les huile appartenant a l'ensemble VEG \*/*

**param** raf\_max\_ani >=0; */\* pour les huile appartenant a l'ensemble ANI \*/*

*/\* les duretés de chaque huiles \*/*

**param** durete {HUILE};

*/\* dureté minimum de la nourriture : pour le projet c'est 3\*/*

**param** durete\_min;

*/\* dureté maximum de la nourriture : pour le projet c'est 6 \*/*

**param** durete\_max;

*/\*prix de la nourriture; pour le projet c'est 150 \*/*

**param** prix\_nourriture;

*/\* stockage maximum des huiles pour un mois: pour le projet c'est 1000 \*/*

**param** st\_max;

*/\*stock initial en tonnes des huiles brutes pour chaque huiles: pour le projet c'est 500 \*/*

**param** st\_initial;

*/\*stock final en tonnes des huiles brutes pour chaque huiles: pour le projet c'est 500 \*/*

**param** st\_final;

*/\*production minimale de nourriture: pour le projet c'est 20 \*/*

**param** prod\_min;

*/\*nombre de mélange maximal autorisé pour les huiles: pour le projet c'est 3 \*/*

**param** melange\_max;

```

/* la matrice des prix mensuels des huiles par tonne */
param prix_mensuel {MOIS,HUILE} >=0;

/* le prix mensuel de stockage par tonne: pour le projet c'est 5 euros */
param prix_stockage_mensuel;

/* paramètre pour la constante d'inflation */
param cst_inflation {MOIS,HUILE};

/* paramètre pour le taux d'inflation */
param taux_inflation;

/* variable pour les quantités en tonnes qu'on utilise pour produire la nourriture */
var qte_utilise {MOIS,HUILE} >=0;
/* variable pour les quantités d'huiles en tonnes qu'on achète pour produire la nourriture */
var qte_achte {MOIS,HUILE} >=0;
/* variable pour les quantités des huiles non utilisée en tonnes qu'on stocke à la fin du mois pour le mois suivant */
var qte_stock {1 .. N+1, HUILE} >=0;

/* Matrice booléenne : 1 si on utilise un type d'huile dans le mois m et 0 sinon */
var est_utilise {MOIS,HUILE} binary;

/* OBJECTIVE */
maximize profit:
    sum {m in MOIS,h in HUILE}
    (prix_nourriture * qte_utilise [m,h] - prix_mensuel [m,h] * ( (100 + cst_inflation [m,h] * taux_inflation)/100 ) *
    qte_achte [m,h] - prix_stockage_mensuel*qte_stock [m,h]);

/*
limitation pour les huiles végétales
 $x1 + x2 \leq 200$ 
*/
subject to utilisation_limite_veg_max {m in MOIS}:
    sum {v in VEG} qte_utilise [m,v] <= raf_max_veg;

/*
limitation pour les huiles animales
 $x3 + x4 + x5 \leq 250$ 
*/
subject to utilisation_limite_ani_max {m in MOIS}:
    sum {a in ANI} qte_utilise [m,a] <= raf_max_ani ;

/* contraintes pour la dureté: */
subject to durete_final_min {m in MOIS}:
    sum {h in HUILE} (qte_utilise [m,h] * durete [h]) >= sum {h in HUILE} (durete_min*qte_utilise [m,h]);

subject to durete_final_max {m in MOIS}:
    sum {h in HUILE} (qte_utilise [m,h] * durete [h]) <= sum {h in HUILE} (durete_max*qte_utilise [m,h]);

/* le stock au début du premier mois de production */
subject to stock_initial {h in HUILE} :
    qte_stock [1,h] = st_initial;

/* le stock final après le dernier mois de production */
subject to stock_final {h in HUILE} :
    qte_stock [N+1,h] = st_final;

```

*/\* limite pour stocker les huiles brutes tous les mois \*/*

**subject to** stock\_max {m in MOIS, h in HUILE} :  
qte\_stock [m,h] <= st\_max;

*/\* équation d'équilibre \*/*

**subject to** equilibre {m in MOIS, h in HUILE} :  
qte\_stock [m,h] + qte\_achte [m,h] = qte\_stock [m+1,h] + qte\_utilise [m,h];

*/\* PARTIE POUR LES VARIANTES \*/*

*/\* VARIANTE 1 \*/*

*/\* si le type d'huile végétales est utilisé dans un mois m on a : est\_utilise[m,h] =1 sinon =0 \*/*

**subject to** limitations\_veg\_m {m in MOIS, h in VEG} :  
qte\_utilise [m,h] - raf\_max\_veg \* est\_utilise [m,h] <= 0;

*/\* si le type d'huile animales est utilisé dans un mois m on a : est\_utilise[m,h] =1 sinon =0 \*/*

**subject to** limitations\_ani\_m {m in MOIS, a in ANI} :  
qte\_utilise [m,a] - raf\_max\_ani \* est\_utilise [m,a] <= 0;

*/\* nombre de mélange des huiles pour produire la nourriture au cours d'un mois doit être inférieur à une limite \*/*

**subject to** limite\_a\_utilise {m in MOIS} :  
sum {h in HUILE} est\_utilise [m,h] <= melange\_max;

*/\* VARIANTE 2 \*/*

*/\* si on utilise un type d'huile végétales au cours d'un mois alors au moins 20 tonnes doivent être utilisées \*/*

**subject to** limitations\_veg\_n {m in MOIS, h in VEG} :  
qte\_utilise [m,h] - prod\_min \* est\_utilise [m,h] >= 0;

*/\* si on utilise un type d'huile animales au cours d'un mois alors au moins 20 tonnes doivent être utilisées \*/*

**subject to** limitations\_ani\_n {m in MOIS, a in ANI} :  
qte\_utilise [m,a] - prod\_min \* est\_utilise [m,a] >= 0;

*/\* VARIANTE 3 \*/*

*/\* SI on utilise VEG1 et VEG2 ALORS on doit utiliser obligatoirement ANI3 \*/*

*/\* Dans cette contrainte on choisit l'huile grâce au critère de sa dureté (5 pour ANI3) \*/*

**subject to** si\_veg1\_et\_veg2\_alors\_ani3 {m in MOIS, a in ANI} :  
2\*est\_utilise [m,a] >= if (durete [a]=5) then prod {v in VEG} est\_utilise[m,v];

*/\* PARTIE DONNEES \*/*

**data;**

**set** VEG := VEG1 VEG2;

**set** ANI := ANI1 ANI2 ANI3;

**param** N := 6;

**param** prix\_nouriture := 150;

*/\* Figure 1.1 \*/*

**param** prix\_mensuel :

	VEG1	VEG2	ANI1	ANI2	ANI3 :=
1	110	120	130	110	115
2	130	130	110	90	115
3	110	140	130	100	95
4	120	110	120	120	125
5	100	120	150	110	105
6	90	100	140	80	135;

**param** raf\_max\_veg := 200;

**param** raf\_max\_ani := 250;

**param** durete\_min := 3;

**param** durete\_max := 6 ;

*/\* Figure 1.2 \*/*

**param** durete :=

VEG1 8.8

VEG2 6.1

ANI1 2.0

ANI2 4.2

ANI3 5.0;

**param** st\_max := 1000;

**param** st\_initial := 500;

**param** st\_final := 500;

*/\* Figure 3.1 \*/*

**param** cst\_inflation :

	VEG1	VEG2	ANI1	ANI2	ANI3 :=
1	0	0	0	0	0
2	1	1	2	2	2
3	2	2	4	4	4
4	3	3	6	6	6
5	4	4	8	8	8
6	5	5	10	10	10;

**param** prix\_stockage\_mensuel := 5;

**param** taux\_inflation := 0; */\* ou 5 selon la trace d'exécution ci-dessous \*/*

**param** prod\_min := 20;

**param** melange\_max := 3;



Trace d'exécution et affichage du résultat sans variantes avec taux\_inflation = 0::

<pre> ampl: model PROJET.ampl; ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal solution; objective 107842.5926 59 dual simplex iterations (12 in phase I) ampl: display qte_utilise; qte_utilise [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 0 250 0 22.2222 177.778 2 0 250 0 0 200 3 0 250 0 159.259 40.7407 4 0 250 0 159.259 40.7407 5 0 250 0 159.259 40.7407 6 0 250 0 159.259 40.7407 ; </pre>	<pre> ampl: display qte_stock ampl? ; qte_stock [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 500 500 500 500 500 2 500 250 500 477.778 322.222 3 500 750 500 477.778 122.222 4 500 500 500 318.519 81.4815 5 500 250 500 159.259 40.7407 6 500 0 500 0 0 7 500 500 500 500 500 ; </pre>
<pre> ampl: display qte_achte; qte_achte [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 0 0 0 0 0 2 0 750 0 0 0 3 0 0 0 0 0 4 0 0 0 0 0 5 0 0 0 0 0 6 0 750 0 659.259 540.741 ; </pre>	<pre> ampl: display qte_utilise, qte_achte, qte_stock; : qte_utilise qte_achte qte_stock := 1 ANI1 0 0 500 1 ANI2 250 0 500 1 ANI3 0 0 500 1 VEG1 22.2222 0 500 1 VEG2 177.778 0 500 2 ANI1 0 0 500 2 ANI2 250 750 250 2 ANI3 0 0 500 2 VEG1 0 0 477.778 2 VEG2 200 0 322.222 3 ANI1 0 0 500 3 ANI2 250 0 750 3 ANI3 0 0 500 3 VEG1 159.259 0 477.778 3 VEG2 40.7407 0 122.222 4 ANI1 0 0 500 4 ANI2 250 0 500 4 ANI3 0 0 500 4 VEG1 159.259 0 318.519 4 VEG2 40.7407 0 81.4815 5 ANI1 0 0 500 5 ANI2 250 0 250 5 ANI3 0 0 500 5 VEG1 159.259 0 159.259 5 VEG2 40.7407 0 40.7407 6 ANI1 0 0 500 6 ANI2 250 750 0 6 ANI3 0 0 500 6 VEG1 159.259 659.259 0 6 VEG2 40.7407 540.741 0 7 ANI1 . . 500 7 ANI2 . . 500 7 ANI3 . . 500 7 VEG1 . . 500 7 VEG2 . . 500 ; </pre>

Trace d'exécution et affichage du résultat sans variantes avec taux\_inflation = 5.:

<pre> ampl: model PROJET.ampl ampl: solve; CPLEX 11.2.0: optimal solution; objective 42990.74074 59 dual simplex iterations (12 in phase I) ampl: display qte_utilise; qte_utilise [*,*] : ANI1 ANI2 ANI3  VEG1    VEG2    := 1  0  250  0    0    200 2  0  250  0  159.259  40.7407 3  0  250  0  159.259  40.7407 4  0  250  0  159.259  40.7407 5  0  250  0  159.259  40.7407 6  0  250  0  159.259  40.7407 ; </pre>	<pre> ampl: display qte_stock; qte_stock [*,*] : ANI1 ANI2 ANI3  VEG1    VEG2    := 1  500  500  500  500    500 2  500  250  500  500    300 3  500  1000  500  340.741  259.259 4  500  750  500  181.481  218.519 5  500  500  500  22.2222  177.778 6  500  250  500  0    137.037 7  500  500  500  500    500 ; </pre>
<pre> ampl: display qte_achte; qte_achte [*,*] : ANI1 ANI2 ANI3  VEG1    VEG2    := 1  0    0  0  0    0 2  0  1000  0  0    0 3  0    0  0  0    0 4  0    0  0  0    0 5  0    0  0  137.037  0 6  0  500  0  659.259  403.704 ; </pre>	<pre> ampl: display qte_utilise, qte_achte, qte_stock; : qte_utilise qte_achte qte_stock := 1 ANI1  0    0    500 1 ANI2  250    0    500 1 ANI3  0    0    500 1 VEG1  0    0    500 1 VEG2  200    0    500 2 ANI1  0    0    500 2 ANI2  250  1000    250 2 ANI3  0    0    500 2 VEG1  159.259  0    500 2 VEG2  40.7407  0    300 3 ANI1  0    0    500 3 ANI2  250    0  1000 3 ANI3  0    0    500 3 VEG1  159.259  0  340.741 3 VEG2  40.7407  0  259.259 4 ANI1  0    0    500 4 ANI2  250    0  750 4 ANI3  0    0    500 4 VEG1  159.259  0  181.481 4 VEG2  40.7407  0  218.519 5 ANI1  0    0    500 5 ANI2  250    0    500 5 ANI3  0    0    500 5 VEG1  159.259  137.037  22.2222 5 VEG2  40.7407  0  177.778 6 ANI1  0    0    500 6 ANI2  250  500    250 6 ANI3  0    0    500 6 VEG1  159.259  659.259  0 6 VEG2  40.7407  403.704  137.037 7 ANI1  .    .    500 7 ANI2  .    .    500 7 ANI3  .    .    500 7 VEG1  .    .    500 7 VEG2  .    .    500 ; </pre>

Trace d'exécution et affichage du résultat avec variantes avec taux\_inflation = 0::

<pre> ampl: model projet2.ampl ampl: option solver cplex; ampl: solve; CPLEX 11.2.0: optimal integer solution within mipgap or absmipgap; objective 104250 16726 MIP simplex iterations 4150 branch-and-bound nodes absmipgap = 6.40722, relmipgap = 6.14602e-05 No basis. 5 flow-cover cuts 8 Gomory cuts ampl: display qte_utilise; qte_utilise [*,*] : ANI1    ANI2    ANI3    VEG1   VEG2    := 1  0  0      250  20      180 2  50 200      0 200      1.42109e-14 3  0 230      20 1.42109e-14 200 4  0 2.84217e-14 250 80      120 5  50 200      0 200      0 6  0 250      0 0      200 ; </pre>	<pre> ampl: display qte_stock; qte_stock [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1  500  500  500  500  500 2  500  500  250  480  320 3  550  430  250  280  320 4  550  200  750  280  120 5  550  200  500  200   0 6  500   0  500   0   0 7  500  500  500  500  500 ;  ampl: display est_utilise; est_utilise [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1  0  0  1  1  1 2  1  1  0  1  0 3  0  1  1  0  1 4  0  0  1  1  1 5  1  1  0  1  0 6  0  1  0  0  1 ; </pre>
<pre> ampl: display qte_achte; qte_achte [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1  0  0  0  0  0 2  100 130 0 0 0 3  0  0 520 0 0 4  0  0 0 0 0 5  0  0 0 0 0 6  0 750 0 500 700 ; </pre>	<pre> ampl: display qte_utilise, qte_achte, qte_stock, est_utilise; : qte_utilise qte_achte qte_stock est_utilise := 1 ANI1 0 0 500 0 1 ANI2 0 0 500 0 1 ANI3 250 0 500 1 1 VEG1 20 0 500 1 1 VEG2 180 0 500 1 2 ANI1 50 100 500 1 2 ANI2 200 130 500 1 2 ANI3 0 0 250 0 2 VEG1 200 0 480 1 2 VEG2 1.42109e-14 0 320 0 3 ANI1 0 0 550 0 3 ANI2 230 0 430 1 3 ANI3 20 520 250 1 3 VEG1 1.42109e-14 0 280 0 3 VEG2 200 0 320 1 4 ANI1 0 0 550 0 4 ANI2 2.84217e-14 0 200 0 4 ANI3 250 0 750 1 4 VEG1 80 0 280 1 4 VEG2 120 0 120 1 5 ANI1 50 0 550 1 5 ANI2 200 0 200 1 5 ANI3 0 0 500 0 5 VEG1 200 0 200 1 5 VEG2 0 0 0 0 6 ANI1 0 0 500 0 6 ANI2 250 750 0 1 6 ANI3 0 0 500 0 6 VEG1 0 500 0 0 6 VEG2 200 700 0 1 7 ANI1 . . 500 . 7 ANI2 . . 500 . 7 ANI3 . . 500 . 7 VEG1 . . 500 . 7 VEG2 . . 500 . ; </pre>

Trace d'exécution et affichage du résultat avec variantes avec taux\_inflation = 5.:

<pre> ampl: model projet2.ampl; ampl: solve; CPLEX 11.2.0: optimal integer solution; objective 39300 1854 MIP simplex iterations 391 branch-and-bound nodes No basis. 4 flow-cover cuts 8 Gomory cuts ampl: display qte_utilise; qte_utilise [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 0 250 0 0 200 2 0 250 0 0 200 3 50 200 0 200 0 4 50 200 0 200 0 5 50 200 0 200 0 6 50 200 0 200 0 ; </pre>	<pre> ampl: display qte_stock; qte_stock [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 500 500 500 500 500 2 500 250 500 500 300 3 700 1000 500 500 100 4 650 800 500 300 100 5 600 600 500 100 100 6 550 400 500 0 100 7 500 500 500 500 500 ;  ampl: display est_utilise; est_utilise [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 0 1 0 0 1 2 0 1 0 0 1 3 1 1 0 1 0 4 1 1 0 1 0 5 1 1 0 1 0 6 1 1 0 1 0 ; </pre>
<pre> ampl: display qte_achte; qte_achte [*,*] : ANI1 ANI2 ANI3 VEG1 VEG2 := 1 0 0 0 0 0 2 200 1000 0 0 0 3 0 0 0 0 0 4 0 0 0 0 0 5 0 0 0 100 0 6 0 300 0 700 400 ; </pre>	<pre> ampl: display qte_utilise, qte_achte, qte_stock, est_utilise; : qte_utilise qte_achte qte_stock est_utilise := 1 ANI1 0 0 500 0 1 ANI2 250 0 500 1 1 ANI3 0 0 500 0 1 VEG1 0 0 500 0 1 VEG2 200 0 500 1 2 ANI1 0 200 500 0 2 ANI2 250 1000 250 1 2 ANI3 0 0 500 0 2 VEG1 0 0 500 0 2 VEG2 200 0 300 1 3 ANI1 50 0 700 1 3 ANI2 200 0 1000 1 3 ANI3 0 0 500 0 3 VEG1 200 0 500 1 3 VEG2 0 0 100 0 4 ANI1 50 0 650 1 4 ANI2 200 0 800 1 4 ANI3 0 0 500 0 4 VEG1 200 0 300 1 4 VEG2 0 0 100 0 5 ANI1 50 0 600 1 5 ANI2 200 0 600 1 5 ANI3 0 0 500 0 5 VEG1 200 100 100 1 5 VEG2 0 0 100 0 6 ANI1 50 0 550 1 6 ANI2 200 300 400 1 6 ANI3 0 0 500 0 6 VEG1 200 700 0 1 6 VEG2 0 400 100 0 7 ANI1 . . 500 . 7 ANI2 . . 500 . 7 ANI3 . . 500 . 7 VEG1 . . 500 . 7 VEG2 . . 500 . ; </pre>