UNIVERSITE D'ALGER
ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT ECONOMIE

THESE DE FIN D'ETUDES

METHODE DES DEUX BASES MAJORANTES

POUR LA RESOLUTION DES PROGRAMMES LINEAIRES EN NOMBRES ENTIERS

SUJET

Proposé par :

M. AIT-OUYAHIA M.

Traité par :

R. BERERHI & S. DIB

ANNÉE SCOLAIRE 1969 - 1970

TABLE DES MATIERES

PAGE

INTRODUCTION

Certains problèmes nécessitent des solutions Entières. Comme la formulation générale des problèmes Economiques aboutit à un programme géant, la résolution de celui-ci en nombres Entiers pose de grandes difficultés.
Il est facile de voir que la méthode employée pour la Résolution des programmes à variables Continues est inopérante ici. Ainsi un Industriel qui se pose le problème d'effectifs pour une nouvelle usine, essaie d'arriver à ses fins en le resolvant par la Programmation linéaire. Or la Solution obtenue lui donne le résultat suivant : 472,4.
Il serait fort désagréable à cet Industriel d'embaucher 472,4 ouvriers. D'où la nécessité de résoudre des programmes en nombre entiers uniquement.
C'est là qu'intervient le Travail de Ralph Gomory entrepris à partir de 1958 sur la resolution des

Programmes linéaires à variables entières. Alors que G. Dantzig avait résolu ceux à variables discrètes depuis 1947.

Si la méthode de Gomory permet de répondre avantageusement au problème de l'Industriel posé plus haut, elle a un inconvénient, c'est la longueur des calculs pour atteindre l'Optimum.

Lorsque l'on connait le prix de l'heure d'utilisation d'un Ordinateur nous comprenons la difficulté.

C'est ainsi que dans cette Etude, nous donnons une variante qui nous permettra d'arriver plus rapidement à l'Optimum et par là même raccourcir le temps de calcul sur l'Ordinateur, d'où une réduction importante de Coût ; tel étant en définitive le But final.

CHAPITRE . I .

THEORIE GENERALE DES PROGRAMMES LINEAIRES ENTIERS

I. Théorie générale des programmes linéaires entiers

I.1.— Rappels

I.1.1.— Définition

Le problème général de la programmation linéaire consiste en la recherche de l'optimum d'une fonction linéaire de n variables liées par des relations linéaires appelées contraintes.

– traduction algébrique et tableau condensé

	1	$-x^J$
x_{i_0}	(i_0,j_0)	(i_0,J)
x_I	(I,j_0)	(I,J)

$$x_{i_0} = (i_0,j_0) + (i_0,J)(-x^J)$$
$$x_I = (I,j_0) + (I,J)(-x^J)$$
$$x_I \,;\, x^J \geq 0$$

I.1.2.— Base

– Définition : une base B sera l'ensemble des n plans linéairement indépendants choisis parmi (1)

$$(1)\quad \begin{cases} x_I = 0 \\ x_J = 0 \end{cases}$$

– Ensembles de bases associées aux programmes linéaires

a) $B_1 \subset B$: Elles sont définies par le point d'intersection de $\underline{n}$ plans de la base –

b) $B_2 \subset B$: le point d'intersection est optimal dans (C), (C) étant le cône positif associé à la base –

c) $B_3 \notin B_1$ et B_2

B_1 est appelée base I-admissible

B_2 est appelée base J-admissible

B_3 est appelée base non admissible ou quelconque

– optimum

L'optimum est atteint lorsque la base est à la fois I et J admissible

– Tableaux récapitulatifs –

	1	$-x^J$
x_{io}		+ + 0 + + +
x_I	+ − − 0 + + 0 +	

Base J-admissible

	1	$-x^J$
x_{io}		+ + − − 0 + + − +
x_I	+ + + 0 + + 0 +	

base I-admissible

	1	$-x^J$
x_{io}		+ + 0 + + + 0 +
x_I	+ + + + 0 + +	

base optimale

	1	$-x^J$
x_{io}		+ − − + + + − + +
x_I	+ − − + 0 + + −	

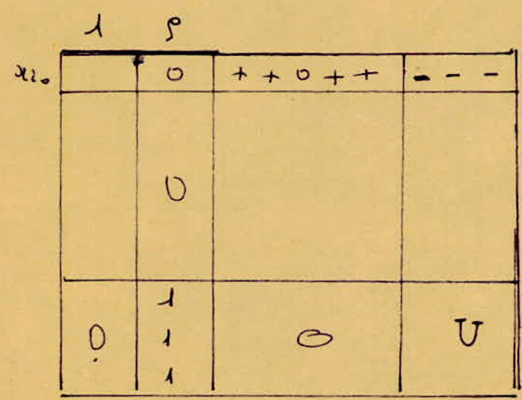
base quelconque

– Redondances

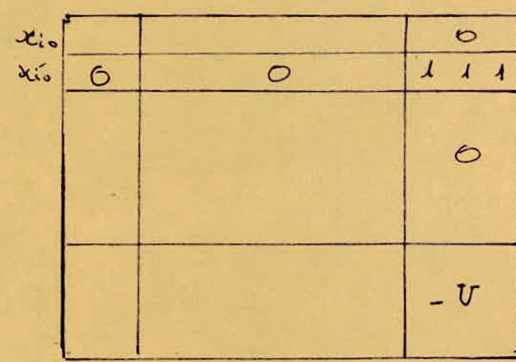
Certaines lignes ou colonnes peuvent présenter les particularités suivantes :

- Phase I

Si la base est quelconque nous disposons de deux sortes de tableaux pour la rendre soit J-admissible (III), soit I-admissible (IV) -



III



IV

U : matrice unité
ρ : 2e second membre
x'_{i0} : 2e fonction économique

- Dégénérescence - Méthode lexicographique

Pour la I-admissibilité nous rajoutons n-1 seconds membres

Pour la J-admissibilité nous rajoutons m-1 fonctions économiques

I - 2. — Méthode de Gomory pour la résolution des programmes linéaires en nombres entiers

I - 2 - 1. — Considérations générales

Le problème se formulait ainsi pour les nombres continus :

$$x_{i_0} = C^J x^J$$

$$x_I = (I,J)(-x^J) + \bar{b}_I$$

$$x_I ; x^J \geq 0$$

x_{i_0} étant la fonction économique qu'il faudra optimiser.

x_I : variables d'écart.

x^J : variables réelles.

$x_I \geq 0$ et $x^J \geq 0$ constituent un domaine $\mathcal{D}$

Si nous rajoutons la contrainte : x^J et x_I entiers au domaine $\mathcal{D}$ nous définissons un nouveau domaine $\mathcal{D}_e$.

Admettons qu'il faille maximiser la fonction économique x_{i_0} ; pour cela nous devons l'écrire ainsi : $x_{i_0} = C^J(-x^J)$.

Le problème de maximisation est le même pour les deux domaines.

Etudions le domaine $\mathcal{D}_e$ - Il est contenu dans $\mathcal{D}$

$$(\mathcal{D}_e) \subset (\mathcal{D})$$

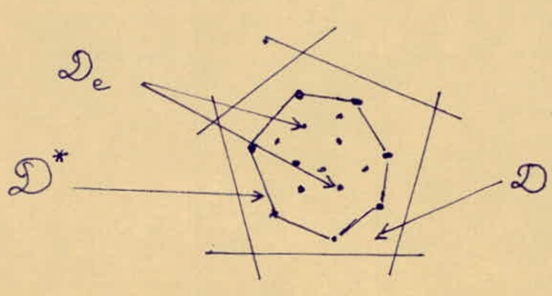


fig - 1 -

De plus

$$\sup_{(\mathcal{D})} x_{i_0} \geq \sup_{(\mathcal{D}_e)} x_{i_0}$$

Considérons l'ensemble des domaines $\mathcal{D}_e$ - Quels sont les domaines $\mathcal{D}$ convexes qui peuvent contenir $\mathcal{D}_e$?

Il existe un domaine $\mathcal{D}^*$ particulier qui est contenu dans tous les domaines $\mathcal{D}$.

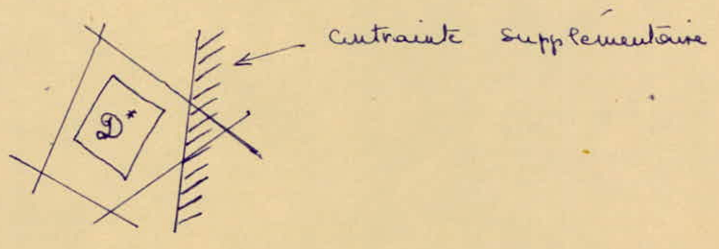
$$(\mathcal{D}^*) \subset \forall (\mathcal{D})$$

$\mathcal{D}^*$ sera la fermeture convexe de $\mathcal{D}_e$. (fig. 1.)

$\mathcal{D}^*$ constitue un polyèdre convexe dont les sommets sont entiers -

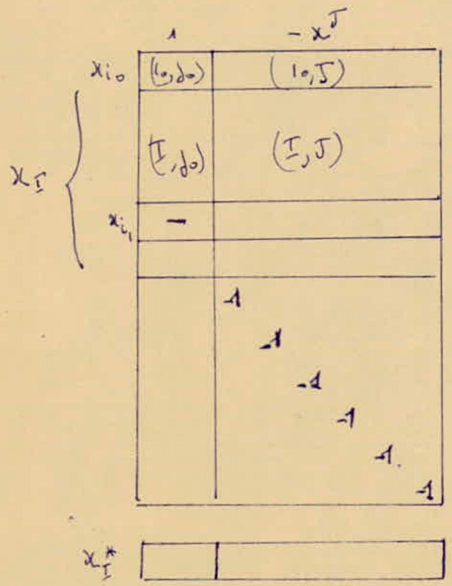
Possédant $\mathcal{D}^*$ il sera facile de maximiser le programme linéaire en oubliant la propriété : x^J entiers ainsi que x_I ; sachant que l'on tombera certainement sur des nombres entiers - Mais en vérité il est difficile de fabriquer $\mathcal{D}^*$. D'où l'introduction de Troncatures -

<u>Troncatures</u> : possédant un domaine convexe, en enlevant un demi espace c'est à dire en ajoutant une contrainte supplémentaire, on dit que l'on a introduit une troncature - Nous n'avons pas le droit de faire une troncature dans le domaine $\mathcal{D}^*$ car nous enlèverions certainement un nombre entier.



I-2-2.– Méthode de Gomory.

Gomory s'est servi de la méthode de résolution des programmes linéaires continus, notamment le cheminement par base J-admissible, avec par conséquent au départ une base J-admissible : $(i_0, J) \geq 0$. Il s'est également servi de contraintes surabondantes (Redondances) – La représentation du tableau condensé s'est faite comme ceci :



$$x_{i_0} = (i_0, j_0) + (i_0, J)(-x^J)$$

$$x_I = (I, j_0) + (I, J)(-x^J)$$

x_I^* étant la troncature

x_{i_1} : contrainte non satisfaite.

fig-2.

Nous constatons, qu'à la différence des programmes linéaires continus, le tableau est plus allongé puisque Gomory rajoute à la suite des x_I les x^J.

De même Gomory utilise un opérateur $\left[\frac{M}{\lambda}\right]$; M étant un nombre algébrique positif, négatif ou nul et λ un nombre supérieur ou égal à 1.
L'opérateur $\left[\frac{M}{\lambda}\right]$ définit un nombre unique qui généralise la division de M par λ.

Diviser M par λ c'est trouver un couple (q, r) tel que $M = \lambda q + r$ avec la condition $r < \lambda$.

$$\boxed{M = \lambda q + r \quad \text{avec } r < \lambda}$$

(q, r) étant unique et q entier

règles élémentaires

. $M < 0 \Rightarrow \left[\frac{M}{\lambda}\right] \to < 0$ si $-3 < \frac{M}{\lambda} < -2 \Rightarrow \frac{M}{\lambda} = -3$

. $M > 0 \Rightarrow \left[\frac{M}{\lambda}\right] \to \geq 0$ si $2 < \frac{M}{\lambda} < 3 \Rightarrow \frac{M}{\lambda} = 2$

— générations de troncatures

quatre conditions sont exigées :

(2)
a) surabondantes (pour ne pas changer le domaine $\mathcal{D}^*$).
b) candidate - pivot (pour accéder à une contrainte non satisfaite).
c) coefficients entiers.
d) pivot $= -1$.

Cette génération est possible sauf

1) si on est à l'optimum
2) $\mathcal{D}^*$ est vide

I-2.3.– Notions de contraintes génératrices

Définition : toute contrainte non satisfaite est une contrainte génératrice.

La fig. 2. présente des contraintes génératrices. Introduisons un indice I_1 qui est l'ensemble des x_I et des x^J

$$I_1 = I \cup J \qquad x_{I_1} = \begin{vmatrix} x^J \\ x_I \end{vmatrix}$$

Il existe donc un négatif dans la colonne x_{I_1} - Soit par exemple x_{i_1}.

$$x_{i_1} = (i_1, j_0) + (i_0, J)(-t^J)$$

Introduisons l'opérateur $\left[\frac{M}{\lambda}\right]$ nous aurons la troncature suivante

$$x^*_\lambda = \left[\frac{(i_1, j_0)}{\lambda}\right] + \left[\frac{(i_0, J)}{\lambda}\right](-t^J)$$

Vérifions les conditions précédentes (2)

b) : vérifiée. car (i_1, j_0) est négatif $\Rightarrow \left[\frac{(i_1, j_0)}{\lambda}\right] < 0$

c) : vérifiée. tous les coefficients sont entiers.

d) : vérifiée. Pour λ suffisamment grand nous sommes assurés que $\left[\frac{M}{\lambda}\right] = -1$ parce que $\left[\frac{M}{\lambda}\right] < 0$ si $M < 0$.

Cela se joue par conséquent sur λ (il faut le prendre très grand) -

a) : nous savons que

pour $\forall$ entier $\mathcal{D} \Rightarrow x^J \geq 0$, $x_I \geq 0$, x_I et x^J entiers (3)

il faut que cela vérifie $x^*_\lambda \geq 0$ entier

démonstration -

nous avons $M = \lambda\left[\frac{M}{\lambda}\right] + r \qquad r < \lambda$

de plus $(i_1, j_0) = \lambda\left[\frac{(i_1, j_0)}{\lambda}\right] + r_0 \quad 0 \leq r_0 < \lambda$

$(i_0, J) = \lambda\left[\frac{(i_0, J)}{\lambda}\right] + r^J \quad 0 \leq r^J < \lambda$

divisons

$$\frac{x_{i_1}}{\lambda} = \frac{i_1, j_0}{\lambda} + \frac{i_0, J}{\lambda}(-t^J)$$

$$= \left[\frac{(i_1, j_0)}{\lambda}\right] + \frac{r_0}{\lambda} + \left\{\frac{(i_0, J)}{\lambda} + \frac{r^J}{\lambda}\right\}(-t^J)$$

soit en fonction de x^*

$$\boxed{\frac{x_{i_1}}{\lambda} = x^*_\lambda + \frac{r_0}{\lambda} + \frac{r^J}{\lambda}\left(-t^J\right)} \quad \text{avec} \begin{cases} 0 \leq \frac{r_0}{\lambda} < 1 \\ 0 \leq \frac{r^J}{\lambda} < 1 \\ \lambda \leq 1 \end{cases}$$

Mais il ne faut pas oublier la vérification (3) - soit

$$x^*_\lambda = \frac{x_{i_1}}{\lambda} - \frac{r_0}{\lambda} + \underbrace{\frac{r^J}{\lambda}}_{\geq 0} \cdot \underbrace{t^J}_{\geq 0}$$

Il faut que $\quad x^*_\lambda \geq \frac{x_{i_1}}{\lambda} - \frac{r_0}{\lambda} \quad$ dans $\mathcal{D}^*$

or $\quad 0 \leq \frac{r_0}{\lambda} < 1$

il faudra donc minorer $\frac{x_{i_1}}{\lambda}$ car $x_{i_1} \geq 0$ donc

$$x^*_\lambda \geq -\frac{r_0}{\lambda} \quad \text{avec } -1 < -\frac{r_0}{\lambda} \leq 0$$

x^*_λ sera positif ou nul parce que x^*_λ est entier

$$\boxed{x^*_\lambda \geq 0} \qquad \text{C.Q.F.D.}$$

I-2-4.- Algorithme de la méthode -

- Base de départ lexicographiquement J-admissible ; sinon nous opérons une Phase I.
- Sélection de i_1 (règle de Dantzig).
- Détermination de la colonne du pivot (j_1) telle que $j_1 = \inf(i_0, J_1)$
- Calcul de λ_m par l'intermédiaire de

a) $\mu_j \; (j \in J_1) = \left[\frac{i_0, j}{i_0, j_1}\right]$

b) $\lambda_j = \left[\frac{i_1, j}{-\mu_j}\right]$

c) $\lambda_m = \operatorname{Sup} \lambda_j$

— calcul de la troncature x^*_λ

$$x^*_\lambda = \left[\frac{x_{i1}}{\lambda_m}\right]$$

— pivotage pour atteindre l'optimum —

I-2-5.— Convergence théorique de la méthode de Gomory

En regardant le second membre, nous constatons qu'à la suite des itérations, ce vecteur décroît lexicographiquement ; ceci grâce à la propriété des vecteurs linéairement indépendants —
Il décroît d'une manière finie car nous avons affaire à des nombres entiers — La colonne j_0 a une valeur optimale bien déterminée —
Elle représentera une borne inférieure —

I-2-6.— Philosophie de la méthode —

Nous constatons de plus en plus de dégénérescences lorsque nous avançons dans les itérations — La fonction économique se vide ainsi que la matrice — Si le zéro est un accident dans les programmes continus, il est exact dans les programmes entiers —
La partie supérieure tend à être corrodée par les zéros c'est à dire mathématiquement, nous tendons vers une diagonalisation de la matrice —
autres constatations :

- le volume de calcul est lié au problème de la taille.
- le choix de la ligne génératrice intervient sensiblement.
- nous notons une complexe interaction de ses différentes sensibilités -

CHAPITRE . II .

METHODE DES DEUX BASES MAJORANTES

II. Méthode des "deux Bases"

II - 1. Introduction :

Les Conclusions que l'on peut tirer de la méthode de Gomory nous aident à mettre au point une nouvelle méthode ; mieux adaptée aux programmes linéaires Geants :

1°) La methode de Gomory est faiblement Convergente :
La Theorie montre que la Convergence existe toujours. En effet si le domaine n'est pas vide les variables auront une valeur optimale qui est une borne pour la décroissance lexicographique, le cheminement se faisant par variables Entières.
La pratique des calculs, elle, montre que le cheminement est très lent, le nombre d'iterations très elevé.

2°) On remarque qu'au fur et à mesure que les iterations ont lieu, la fonction economique d'abord, la matrice ensuite tend à se vider dans la region superieure à la diagonale. On tend vers un shema semblable à celui de la figure 1 où : — x_{ik} est le premier terme qui decroit.

— la partie du Tableau qui va de x_i à x_j ne varie pas

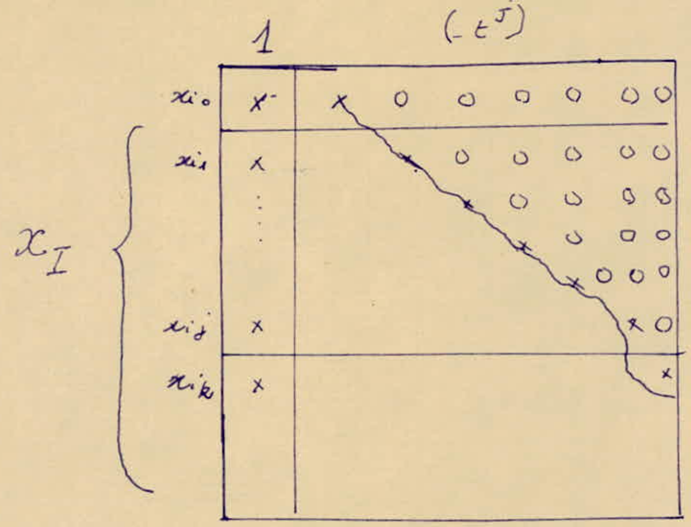


fig 1.

II. 2. — Definition de la troncature à elements negatifs:

. Nous avons déjà defini la troncature d'une façon generale et dit les Conditions qu'elle doit satisfaire.

Il serait interessant de ne prendre en consideration que les elements negatifs de la ligne generatrice et leur faire correspondre des elements dans la troncature. En effet cela nous assurerait de la decroissance lexicographique et par consequent nous eviterait de Cycler.

. Il est facile de voir que cette troncature est surabondante:

. Soit x_{i_1} une ligne generatrice (fig. 2)

$(i_1\ j')$ = Ensemble des elements < 0 de la ligne

$(i_1\ j'')$ = " " " $\geqslant 0$ " " "

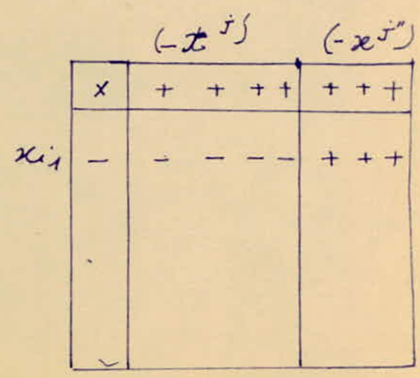


fig 2

. On peut ecrire : $x_{i_1} = (i_1 j_0) + (i_1 j')(-x^{j'}) + (i_1 j'')(-x^{j''})$

or on peut ecrire : $i_1 j_0 = \lambda \left[\frac{i_1 j_0}{\lambda}\right] + r \qquad 0 \leq r < \lambda$

$$i_1 j' = \lambda \left[\frac{i_1 j'}{\lambda}\right] + r' \qquad 0 \leq r' < \lambda$$

$$i_1 j'' = \lambda \left[\frac{i_1 j''}{\lambda}\right] + r'' \qquad 0 \leq r'' < \lambda$$

$$\frac{x_{i_1}}{\lambda} = \left[\frac{i_1 j_0}{\lambda}\right] + \frac{r}{\lambda} + \left[\left[\frac{i_1 j'}{\lambda}\right] + \frac{r'}{\lambda}\right](-x^{j'}) + \left[\left[\frac{i_1 j''}{\lambda}\right] + \frac{r''}{\lambda}\right](-x^{j''})$$

Si l'on designe par x_λ^{**} la troncature à elements negatifs :

$$x_\lambda^{**} = \left[\frac{i_1 j_0}{\lambda}\right] + \left[\frac{i_1 j'}{\lambda}\right](-x^{j'})$$

$$\Rightarrow \quad \frac{x_{i_1}}{\lambda} = x_\lambda^{**} + \frac{r}{\lambda} - \frac{r'}{\lambda}(x^{j'}) - A(x^{j''}) \quad \text{où } A = \left[\left[\frac{i_1 j''}{\lambda}\right] + \frac{r''}{\lambda}\right] \geq 0$$

et $0 \leq \frac{r'}{\lambda} < 1 \Rightarrow \frac{x_{i_1}}{\lambda} \leq x_\lambda^{**} + \frac{r}{\lambda} \Rightarrow x_\lambda^{**} \geq \frac{x_{i_1}}{\lambda} - \frac{r}{\lambda}$

or $x_{i_1} \geq 0 \Rightarrow x_\lambda^{**} \geq -\frac{r}{\lambda}$ avec $-1 < -\frac{r}{\lambda} \leq 0$

x_λ^{**} etant entier on ne peut avoir que :

$$\boxed{x_\lambda^{**} \geq 0} \qquad \text{c.q.f.d.}$$

. La troncature est donc bien surabondante.

II.3.— Probleme à lignes prèredondantes

. Un tel problème se pose ainsi :
$$\begin{cases} x_{i_0} = i_0 j_0 + (i_0 J)(-x^J) \\ x_I = I j_0 + (I, J)(-x^J) \\ x^J,\ x_I \geq 0 \text{ et Entiers.} \end{cases}$$

avec la Condition supplementaire : $\left\{\begin{matrix} (i_o, \bar{J}) \\ \\ (I, \bar{J}) \end{matrix}\right\} \underset{\geqslant}{\ell} 0$

. En outre on n'a qu'un seul negatif par ligne.

Supposons que l'on ait egalement : les positifs de la ligne inferieurs en valeur absolue au negatif de la même ligne. Alors la Troncature se reduit à : (-1) sous la colonne candidate et à 1 nombre $\alpha < 0$ tel que : $\alpha = \left[\frac{i_k j_o}{i_k j_n}\right]$ sous le second membre

Les problemes à lignes precedoutants permettent d'expliquer la Notion de <u>Base Majorante</u> <u>fig. 3</u>

	1	$(-t^{j})$			$-t^{jk}$	
	x	+	+	+	+	+
	$-$	+	$-$	+	+	+
x_{ik}	$-$	+	+	+	$-$	+
	$-$	$-$	+	+	+	+
	$-$	+	+	$-$	+	+
	$-$	+	+	+	$-$	+
	⋮					
x^{**}_{λ}	α				(-1)	

1 . <u>Definition</u> :

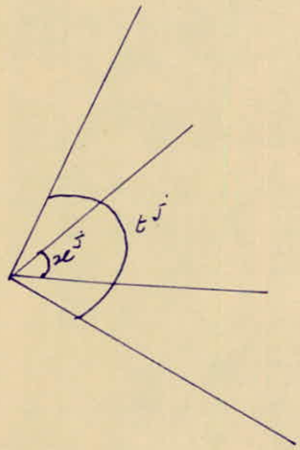
Un Base Majorante est une base qui par rapport à une autre base quelconque, a la propriété de l'envelopper et telle qu'une direction de plan ne coupe qu'une arrête. fig. 5.

Ceci se traduit dans le Tableau par l'existence d'un seul negatif par ligne. Si le probleme est carré, et s'il n'y a pas de redondances-colonne il n'y aura egalement qu'un seul negatif par colonne. fig. 4.

L'intérêt d'une telle méthode est évident.

Le nombre d'itérations pour atteindre l'Optimum sera plus faible puisque l'on a vu que la troncature se limite à (-1) sous la colonne candidate ; α sous le second membre et qu'à chaque itération nous satisfaisons 1 ligne.

fig. 5



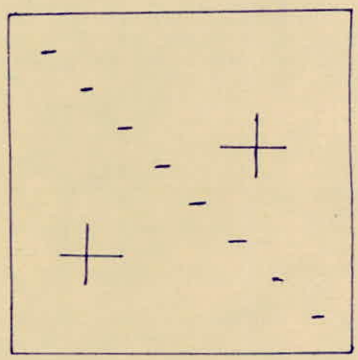


fig. 4

Structure de Base Majorante

Considérons maintenant la Représentation géométrique suivante : (fig. 6)

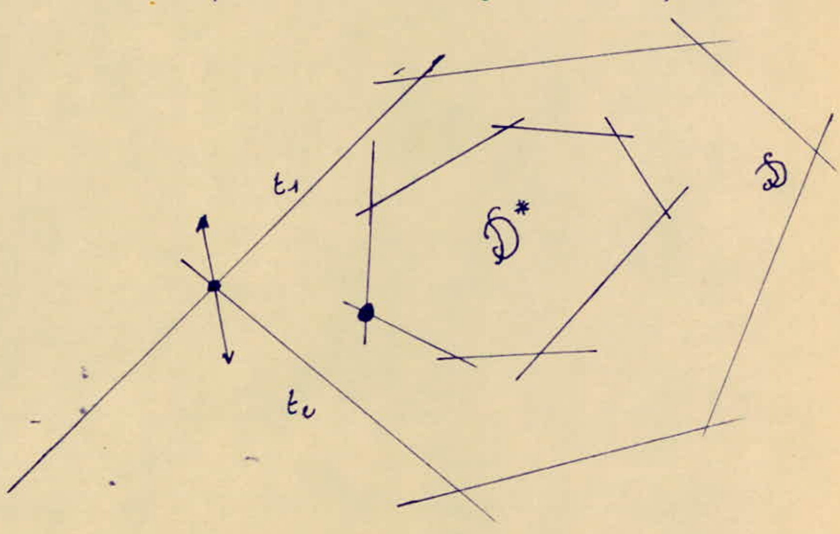


fig. 6.

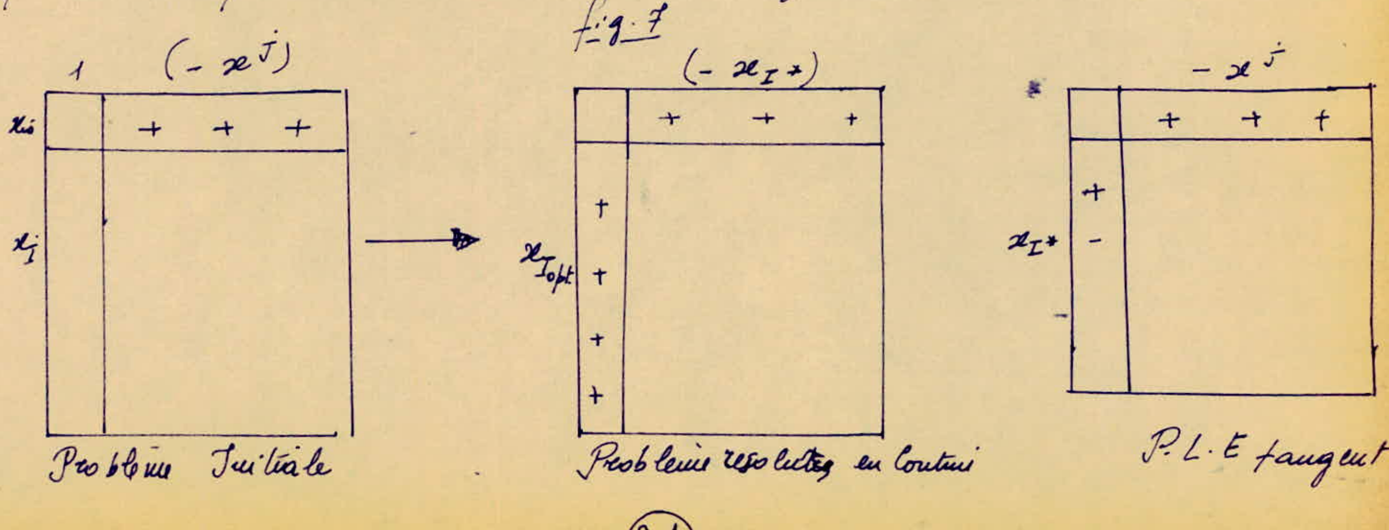
. On voit tout de suite que seules les contraintes t_1 et t_2 jouent à l'Optimum des autres contraintes peuvent etre supprimées.

Si donc x_{I^*} est l'ensemble des contraintes t^J qui interviennent à l'optimum autrement dit si x_{I^*} est la <u>Base Optimale</u> du probleme resolu en Continu; on peut alors n'utiliser que cette derniere pour la recherche de l'optimum du probleme en Nombres entiers.

la premiere etape du probleme est donc la recherche de l'Optimum du probleme en nombre Continu. Nous aurons alors la Base x_I^*, qui exprimee en fonction des variables x^J nous donnera un probleme Canonique Carré que nous appellerons "Probleme Entier tangent". (fig. 7)

Ce probleme une fois reduit (suppression des redondances s'il y a lieu) devra alors être transformé en un probleme ayant un seul negatif et par colonne : Autrement dit trouver pour ce probleme une Base Majorante.

fig. 7



Probleme Initiale

Probleme resolu en Continu

P.L.E tangent

II. 4. — Recherche de la 1ère Base Majorante :

Le problème se pose ainsi :

Ayant une Base J-admissible (Il est toujours possible de s'y ramener par une phase I) Comment faire pour obtenir un seul negatif par ligne et par colonne. C'est-à-dire Comment faire pour Majorer la Base x^J par une base courante t^J de façon à conserver intacts tous les points appartenant au domaine $\mathcal{D}^*$ defini par les Conditions initiales ; L'un des points de ce domaine étant susceptible d'être l'optimum cherchée.

Cette base, qui existe toujours, sera appelée « première Base Majorante »

Compte tenu des Remarques faites à propos de la Methode de Gomory, deux façons de proceder s'offre à nous :

★ Soit faire des Troncatures à elements negatifs sur les lignes qui en contiennent plus d'un.

★ Soit Commencer par trianguulariser le Tableau dans sa partie Superieure pour essayer dans une première phase d'approcher au maximum, la Base Majorante puis faire des Troncatures pour terminer.

La façon de proceder dependra de la nature du problème.

Car il sera inutile et fastidieux de trianguulariser si le Problème au depart à l'avantage de presenter une structure proche de celle recherchée. La triangularisation nous ferait perdre cet avantage.

II.4.1 : Utilisation de la Troncature à element negatifs:

Un processus operatoire Consiste à :

- choisir la ligne ayant le plus de negatif.
- choisir la colonne par la règle de l'inf lexicographique parmi les colonnes correspondant aux differents negatifs de la ligne generatrice choisie.
- Choisir les elements negatifs de la troncature (β_i) de façon à rester lexicographiquement positif.

pour cela on prend $\beta_j = \left[-\frac{a_{ij}}{a_{ik}} \right]$ (cf. fig. 8)

si $\alpha_j - \beta_j \alpha_k > 0$ on choisit β_j

si $\alpha_j - \beta_j \alpha_k < 0$ on fait $\beta_j = \beta_j - 1$ et on recommence le test.

si $\alpha_j - \beta_j \alpha_k = 0$ on passe a la ligne suivante et on fait le test sur $a_{1j} - \beta_j a_{1k}$ et ainsi de suite.

- Une fois les β_j determinés on effectue le pivotage.

La figure 8 montre :

$\cdot\ \lambda_i \leq 0$; $\alpha_k = \inf (\alpha_j)$ $\{j = 1, \dots n\}$

$\alpha_j \geq 0$; $\beta_j < 0$ sauf pour $a_{ij} \geq 0$ $\{j = 1, \dots n\}$

alor $\beta_j = 0$

$$
\begin{array}{c|c|cccccccc}
 & & & & (-t^J) & & \downarrow & & & \\
\hline
 & & \alpha_1 & \alpha_2 & \cdots & \alpha_j & \cdots & \alpha_k & \cdots & \alpha_n \\
\hline
x_{i_1} & \lambda_1 & a_{11} & \cdots & \cdots & a_{1j} & & a_{1k} & \cdots & a_{1n} \\
 & \vdots & & & & & & & & \\
x_I & \vdots & & & & & & & & \\
\rightarrow & \lambda_i & a_{i1} & \cdots & & a_{ij} & & a_{ik} & \cdots & \\
 & \vdots & & & & & & & & \\
 & \lambda_n & a_{n1} & & & & & & & a_{nn} \\
\hline
x_\lambda^{**} & & \beta_1 & \cdots & \cdots & \beta_j & \cdots & (-1) & \cdots & \beta_n
\end{array}
$$

fig 8

Tant que le nombre de negatifs dans une ligne generatrice quelconque depasse 1, le processus se continue et l'on arrivera finalement a la base majorante cherchée.

II.4.2.— Methode d'approche par la triangularisation.

Remarques sur la fig. 9.

1. Le signe moins (−) de la dernière colonne est necessaire car un signe plus (+) ou un zero ~~nombre nul~~ donnerai une redondance.

$$
\begin{array}{c|ccc}
 & + & + & + \\
\hline
- & \times & \times & \times \\
- & \times & \times & \times \\
- & \times & \times & \times
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{c|ccc}
 & + & 0 & 0 \\
\hline
- & - & + & 0 \\
- & \times & \times & \times \\
- & \times & \times & -
\end{array}
$$

fig. 9.

2. le nombre a_{11} sera necessairement negatif sinon le domaine serait non borné.

3. Le nombre positif en tête de chaque colonne, necessaire à la lexico-positivité, n'est autre que le P.G.C.D des nombres > 0 de chaque ligne.

La triangularisation se fait alors sans problemes, en vidant systematiquement chaque ligne en commençant par le haut et en utilisant des troncatures de Gomory.

Une fois la triangularisation terminée on commence à eliminer les moins (–) supplementaires dans chaque ligne. Mais cette fois on ne souci même plus de la lexico-positivité celle-ci etant respectée automatiquement. On utilisera pour cela des troncatures à elements negatifs. On commencera à satisfaire le tableau à partir du bas.

Le cyclage reste cependant possible car une troncature sur la ligne x_{i_k} peut creer un ou plusieurs negatifs sur la ligne $x_{i_{k+1}}$. Il faut alors y revenir jusqu'à obtention de la base majorante.

II. 5 . — Exemples d'application des la methode :

II. 5. 1 : Exemple où la triangularisation est inutile :

. Soit à chercher une base majorante dans le programme lineaire suivant :

	1	$-t_1$	$-t_2$	$-t_3$	t_4
x_0	1	2	1	5	6
x_1	-2	-5	4	9	10
x_2	-35	7	-7	-8	2
x_3	-3	1	-1	-2	3
x_4	-2	1	-1	-2	-2

appliquons la 1ère methode :

* La 1ère troncature se fait sur la ligne x_4 qui possede 3 negatifs.
* La colonne est t_2 car $\inf(1,5;6)=1$
* la lexico-positivité nous permet $\begin{cases} \beta_3 = -2 \\ \beta_4 = -2 \end{cases}$

(t_1)

	1	$-t_1$	$-t_2$	$-t_3$	$-t_4$
x_0	1	2	1	5	6
x_1	-2	-5	4	9	10
x_2	-35	7	-7	-8	2
x_3	-3	1	-1	-2	3
x_4	-2	1	-1	-2	-2
x_λ^{**}	.	.	-1	-2	-2

(t_2)

	1	$-t_1$	$-t_2'$	$-t_3$	$-t_4$
x_0	1	2	1	3	4
x_1	-2	-5	4	1	2
x_2	-35	7	-7	6	16
x_3	-3	1	-1	.	5
x_4	-2	1	-1	.	.
		.			

En 1 seule troncature nous atteignons la structure cherchée.

et 2 colonnes redondantes : $\begin{cases} t_3 = 0 \\ t_4 = 0 \end{cases}$

(t_3)

	1	$-t_1$	$-t_2'$
x_0	1	2	1
x_1	-2	-5	4
x_2	-35	7	-7
x_3	-3	1	-1
x_4	-2	1	-1

(t_4)

	1	$-t_1$	$-x_3''$
x_0	-2	3	1
x_1	-14	-1	4
x_2	-14	.	-7
t_2'	3	-1	-1
x_4	1	.	-1

(t_5)

	1	$-t_1$	$-x_3$
x_0	-2	3	1
x_1	-14	-1	4
x_2	-14	.	-7

$x_4 = 1 - t_1 + x_3$

. On voit que t_3 ne possède plus que 1 seul negatif par ligne. Un pivotage autour de (1) de la ligne preredondante $-x_3$ nous donne le tableau t_4 où les lignes t_2 et x_4 sont devenues redondantes.

Des Calculs relativement simples et rapides, nous permettent d'obtenir le Tableau cherché à un ~~tro~~ negatif par ligne et par colonne ; soit t_5.

. La Triangularisation nous aurait fort eloigné de la solution.

II. 5. 2 : Exemple de Triangularisation :

Soit t_0 le Tableau de depart à Triangulariser avant la Recherche de la Base Majorante :

t_0

	1	$-x_1$	$-x_2$	$-x_3$	$-x_4$
x_0	.	1	2	3	4
x_5	-200	-3	-1	.	-2
x_6	-100	6	-2	-3	.
x_7	-50	.	1	-3	-5
x_8	.	.	.	-1	.
t_1^*	.	-1	-2	-3	-4

	1	$-t_1$	$-x_2$	$-x_3$	$-x_4$
x_0	.	1	.	.	.
x_5	-200	-3	5	9	10
x_6	-100	6	-14	-21	-24
x_7	-50	.	1	-3	-5
x_8	.	.	.	-1	.
t_2^*	.	.	-1	-1	-2

	1	$-t_1$	$-t_2$	$-x_3$	$-x_4$
x_0	.	1	.	.	.
x_5	-200	-3	5	4	.
x_6	-100	6	-14	-7	4
x_7	-50	.	1	-4	-7
x_8	.	.	.	-1	.
	.	.	-1	-1	.

t_0 t_1 t_2

	1	$-t_1$	$-t_2$	$-t_3$	$-x_4$
x_0	.	1	.	.	.
x_5	-200	-3	1	4	.
x_6	-100	6	-7	-7	4
x_7	-50	.	5	-4	-7
x_8	.	.	1	-1	.
t^*_λ	.	.	-1	-4	

(t_3)

	1	$-t_1$	$-t'_2$	$-t_3$	$-x_4$
x_0		1	.	.	.
x_5	-200	-3	1	.	.
x_6	-100	6	-7	21	4
x_7	-50	.	5	-24	-7
x_8	.	.	1	-5	.
t^*_λ				-5	-1

(t_4)

	1	$-t_1$	$-t'_2$	$-t'_3$	$-t_4$
x_0		1	.	.	.
x_5	-200	-3	1	.	.
x_6	-100	6	-7	1	4
x_7	-50	.	5	11	-7
x_8	.	.	1	-5	.
t^*					

(t_5)

Intervertissons x_7 et x_8 on obtient t_6

	1	$-t_1$	t'_2	$-t_3$	$-t_4$
x_0	.	1	.	.	.
x_5	-200	-3	1	.	.
x_6	-100	6	-7	1	4
x_8	.	.	1	-5	.
x_7	-50	.	5	11	-7

(t_6)

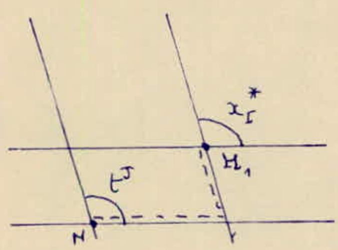
On peut dores et déjà arrêter la Triangularisation car le Tableau (t_6) possède déjà la Base Majorante cherchée.

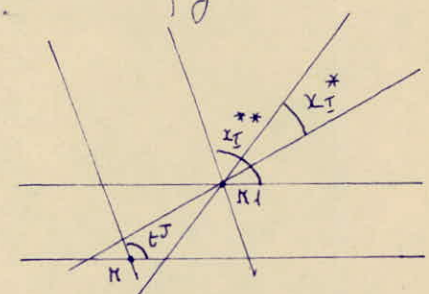
D'où l'Intérêt d'un tel processus.

II - 6 - TRIANGULARISATION -

Comme toujours, le problème est d'atteindre l'optimum le plus rapidement possible – Nous possédons la base majorante parfaite – Si le nombre d'itérations est très grand ceci provient du fait que les bases t^J et x_I^* ne sont pas identiques – Géométriquement en rendant parallèles les plans représentant ces bases, nous atteindrons l'optimum en deux itérations (fig - 10 -) -

(fig-10-)

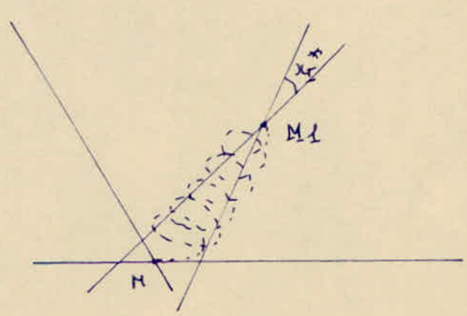




(fig - 11)

Mais présentement x_I^* se trouve sous la forme illustrée dans la fig 11 -, nous devons introduire une base auxiliaire x_I^{**} parallèle à t^J et ayant le même sommet que x_I^* pour atteindre rapidement l'optimum -

Pour obtenir ce parallélisme nous faisons appel à la triangularisation qui permettra d'éviter dans le cas de <u>n</u> contraintes un cheminement en forme d'hélice (fig 12)

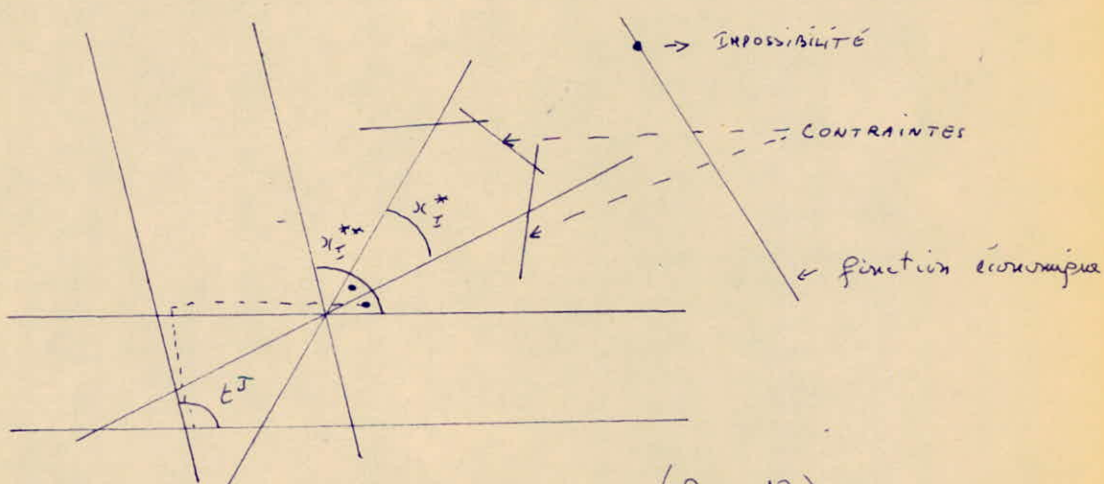


(fig - 12)

II - 7 - FINITIONS

après les n itérations opérées suivant les pivots négatifs de la base Majorante x_I^{**}, nous constatons alors la positivité du second membre pour cette partie - Géométriquement le sommet se situe dans x_I^{**} - Il reste à vérifier s'il appartient aussi à x_I^* c'est à dire si le reste du second membre (contraintes n'appartenant pas à la base optimale) est positif - S'il ne l'est pas nous effectuons des itérations supplémentaires (en nombre restreint d'ailleurs car nous ne sommes pas loin de l'optimum) appelées finitions.

Remarque: à la suite des itérations et des finitions, l'optimum peut ne pas être atteint - Géométriquement le point est dans x_I^{**} ainsi que dans x_I^* mais il ne vérifie pas les contraintes (fig 13). C'est l'impossibilité.



(fig. 13)

II. 8 : Synthese Generale

La Methode des deux Bases Majorantes, s'applique donc à n'importe quel probleme pourvu qu'il ait une solution.

A. — Si le Tableau de depart est J-admissible le processus Operatoire est le suivant :

① Resoudre le programme lineaire en nbres continus. Qui nous donne une Base Optimale de Rang N.

② Cette base exprimée en fonction de la base initiale B_0 donne un probleme Carré.

③ On cherche une Base Majorante pour la Base initiale

④ On cherche une deuxieme Base Majorante pour la Base Optimale par triangularisation.

⑤ Après N iterations et eventuellement quelques Operations de finitions nous atteignons l'Optimum.

B. — Si le Tableau de depart n'est pas J-Admissible

① On installe les données de la PHASE I.

② On resout le probleme en Continu pour la PHASE I. A l'Optimum nous avons une Base B1

③ On resout le Probleme en Continu pour la Phase II.

On trouve une Base Optimale B_2.

(4) On exprime B_1 en fonction de B_0 → problème Carré.

(5) Pivotage autour de la Matrice Unité → I-Admissibilité.

(6) Recherche des 2 Bases Majorants.

(7) N. Itérations - finition.

(8) Suppression des variables artificielles et du Nouveau second membre.

(9) Recherche des 2 Bases Majorants pour B_2.

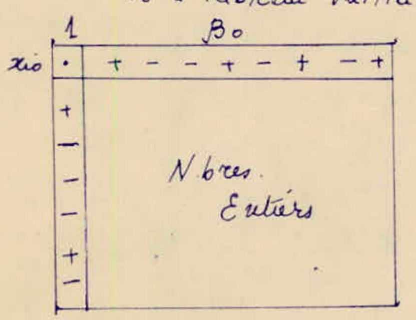
(10) N Itérations - finition

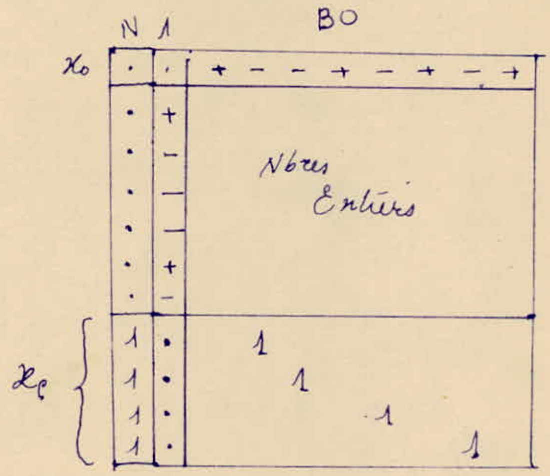
→ Optimum du problème.

Schema Resumé

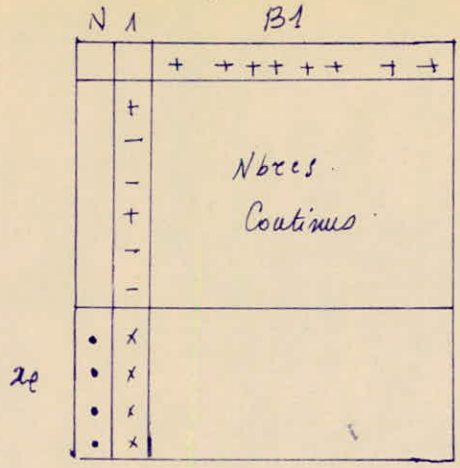
Installation Phase I.

T_0 = tableau Initiale.

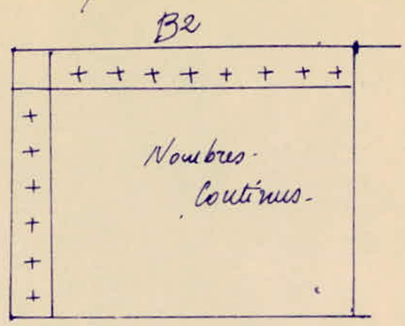


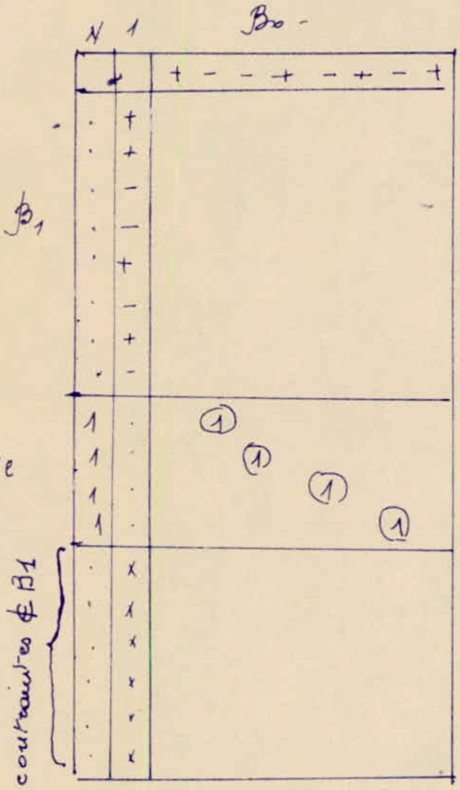


Optimum Base I.

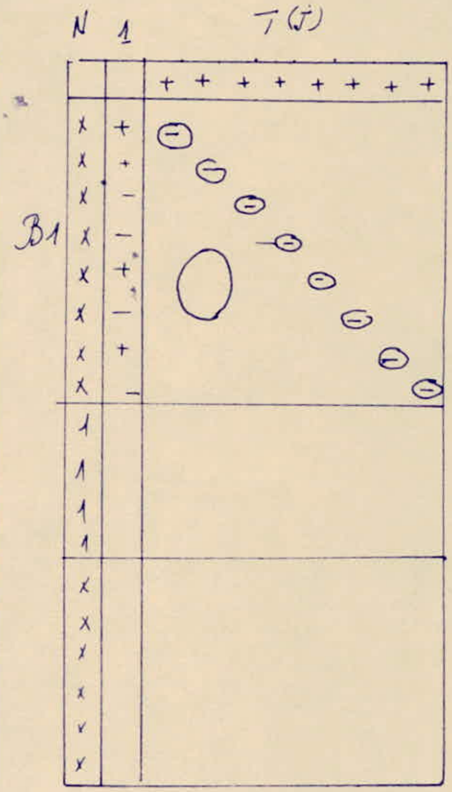


Supression V.A. et N. aux second membre.
Optimum Φ II.

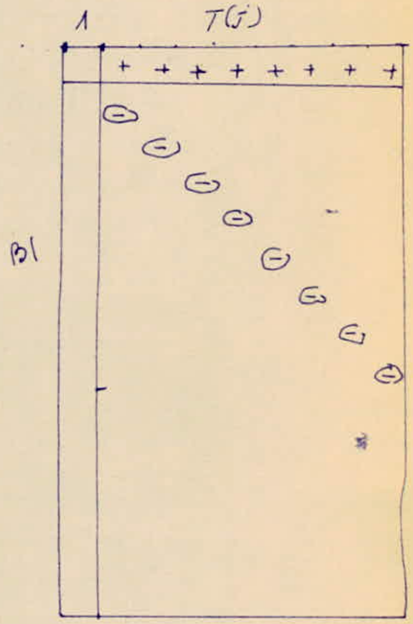




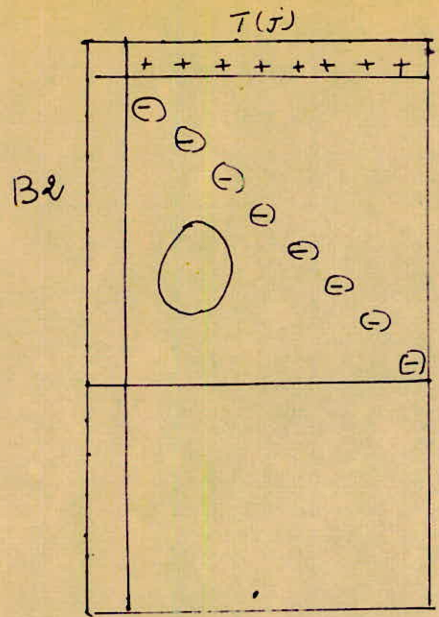
Installation Φ I.
pivotage autour
de la matrice.
Unité.



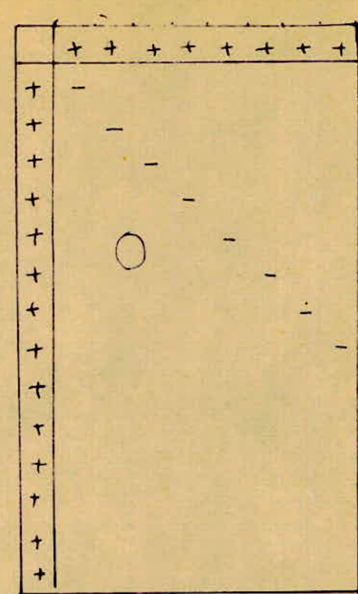
Base
Majorante
parfaite



Optimum Φ I.
suppression V.A.
et Naux S.M.



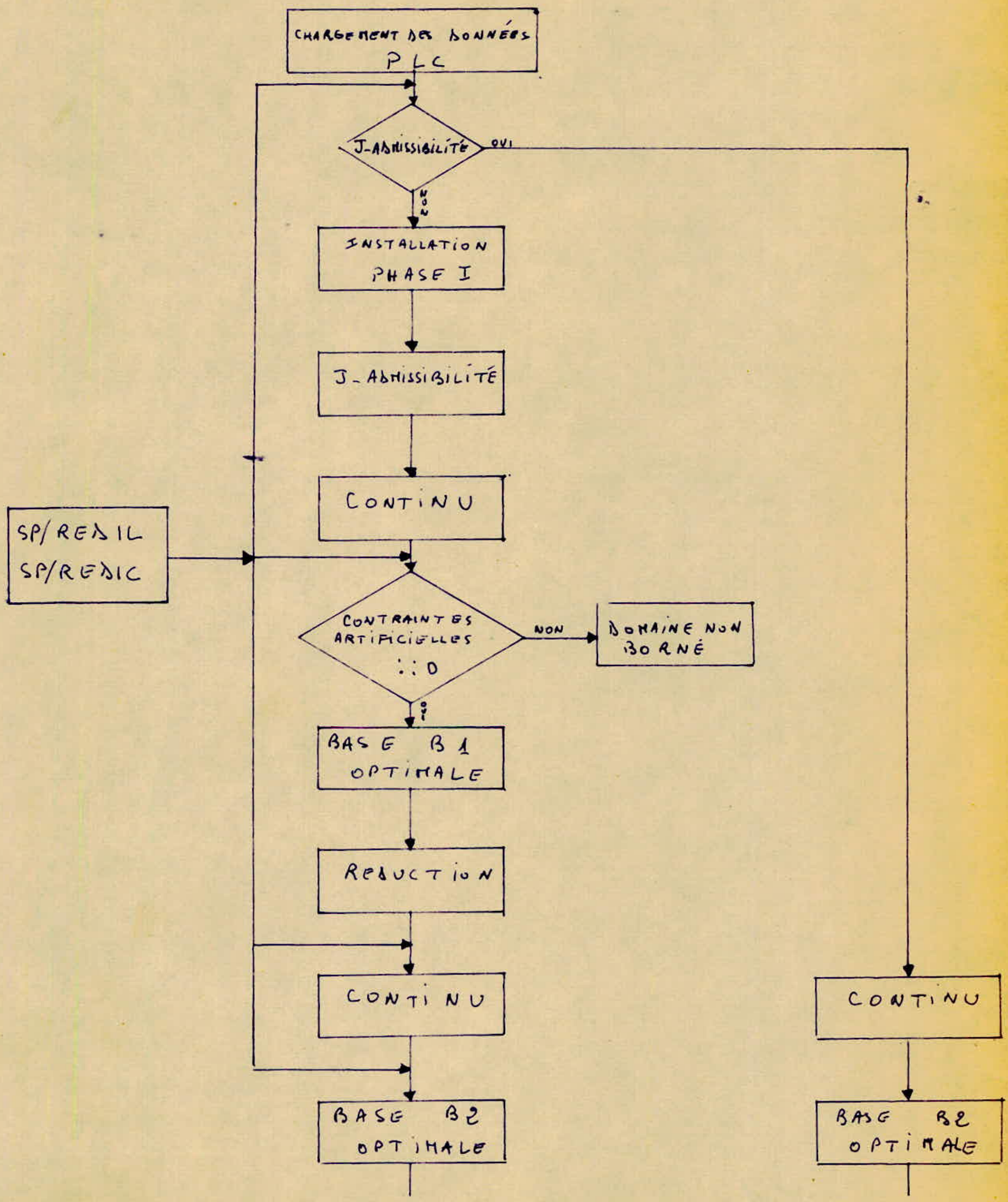
Recherche d'une base
Majorante Parfaite pour B2.



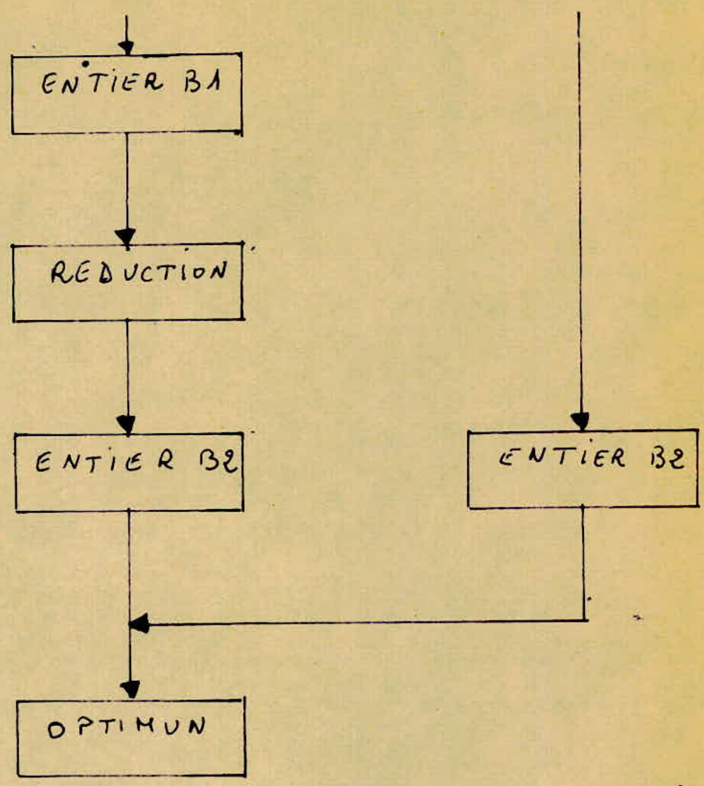
Optimum du problème

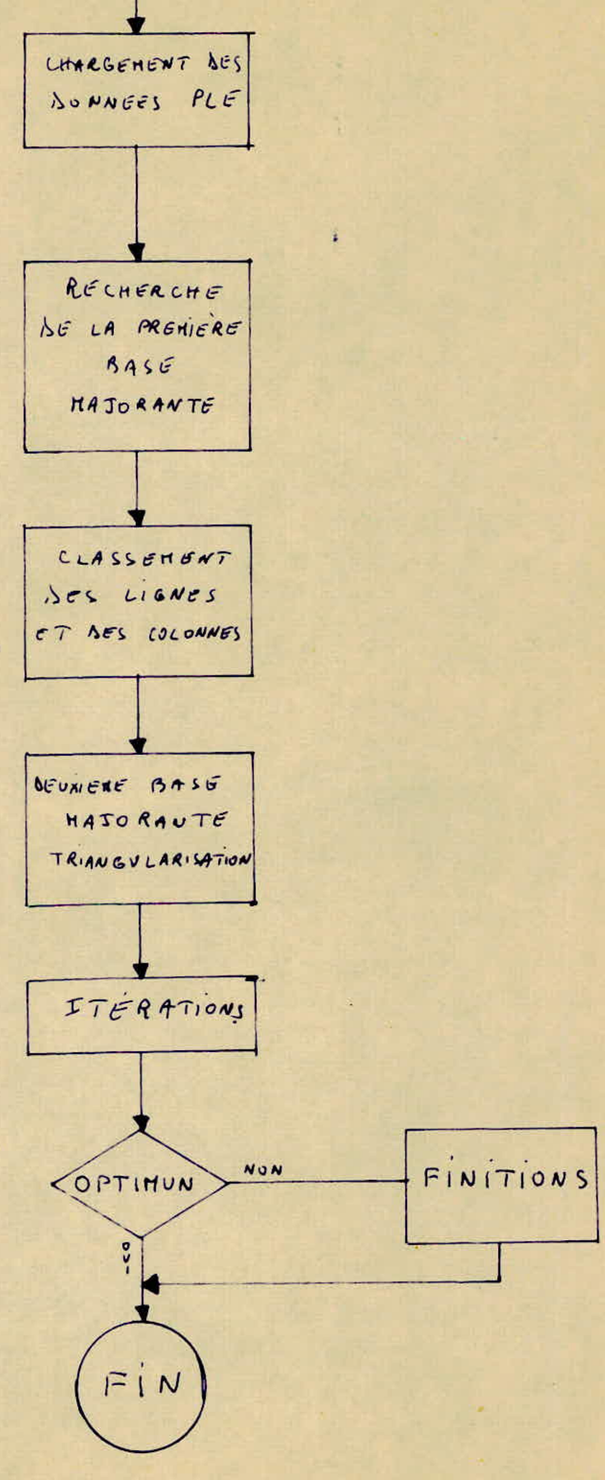
– ORGANIGRAMMES –

ORGANIGRAMME GÉNÉRAL - I -

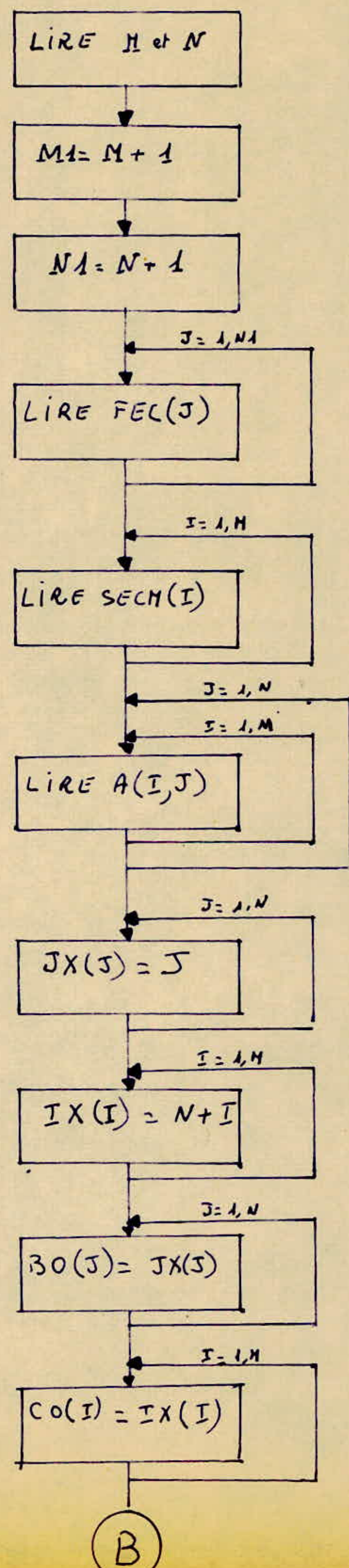


ORGANIGRAMME GÉNÉRAL - II -

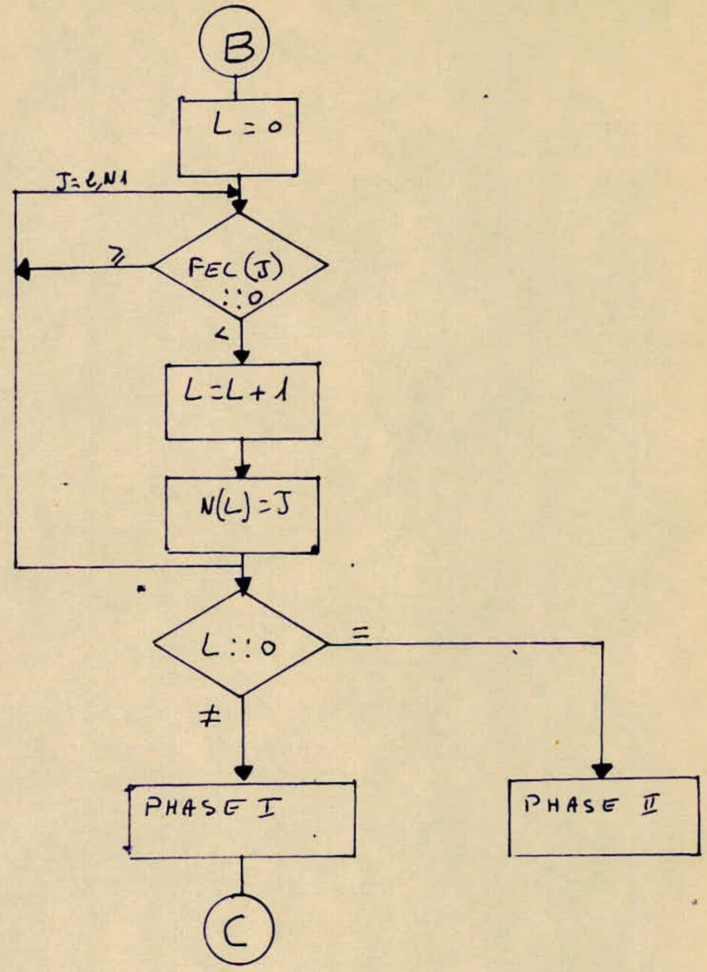



CHARGEMENT DES
DONNEES PLE
RECHERCHE
DE LA PREMIERE
BASE
MAJORANTE
CLASSEMENT
DES LIGNES
ET DES COLONNES
DEUXIEME BASE
MAJORANTE
TRIANGULARISATION
ITÉRATIONS
OPTIMUN
NON
FINITIONS
OUI
FIN

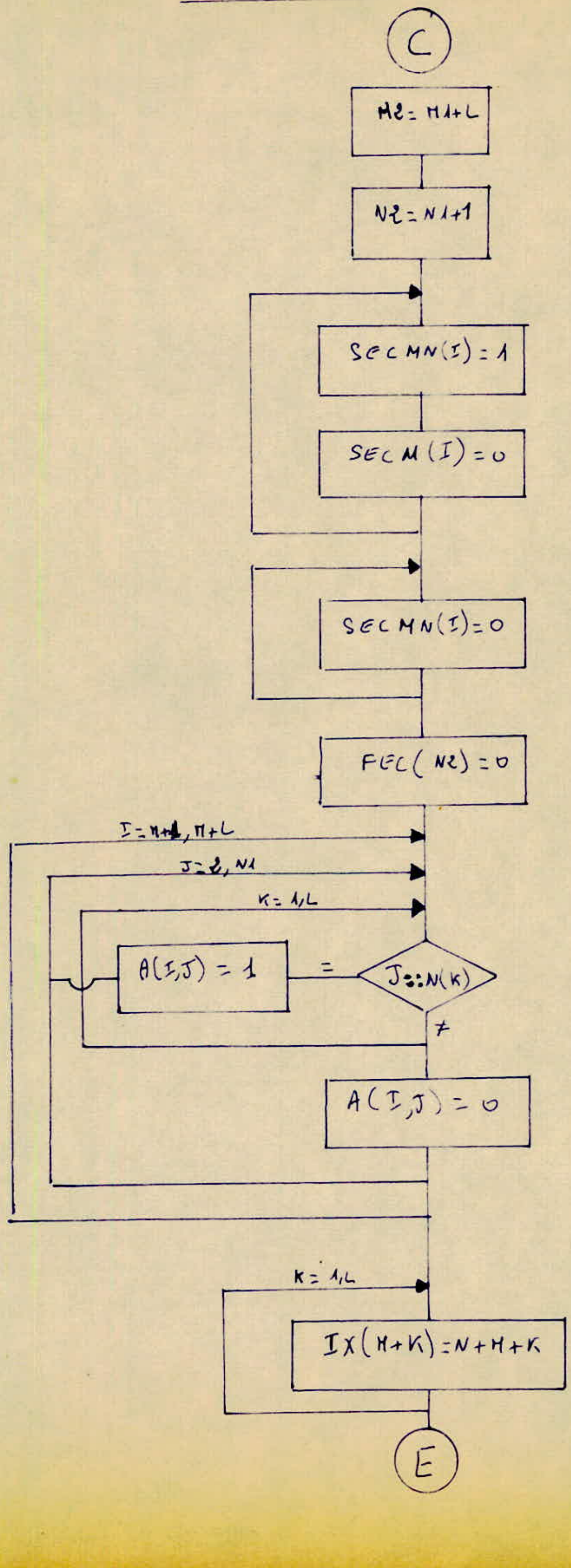
- CHARGEMENT DES DONNÉES -



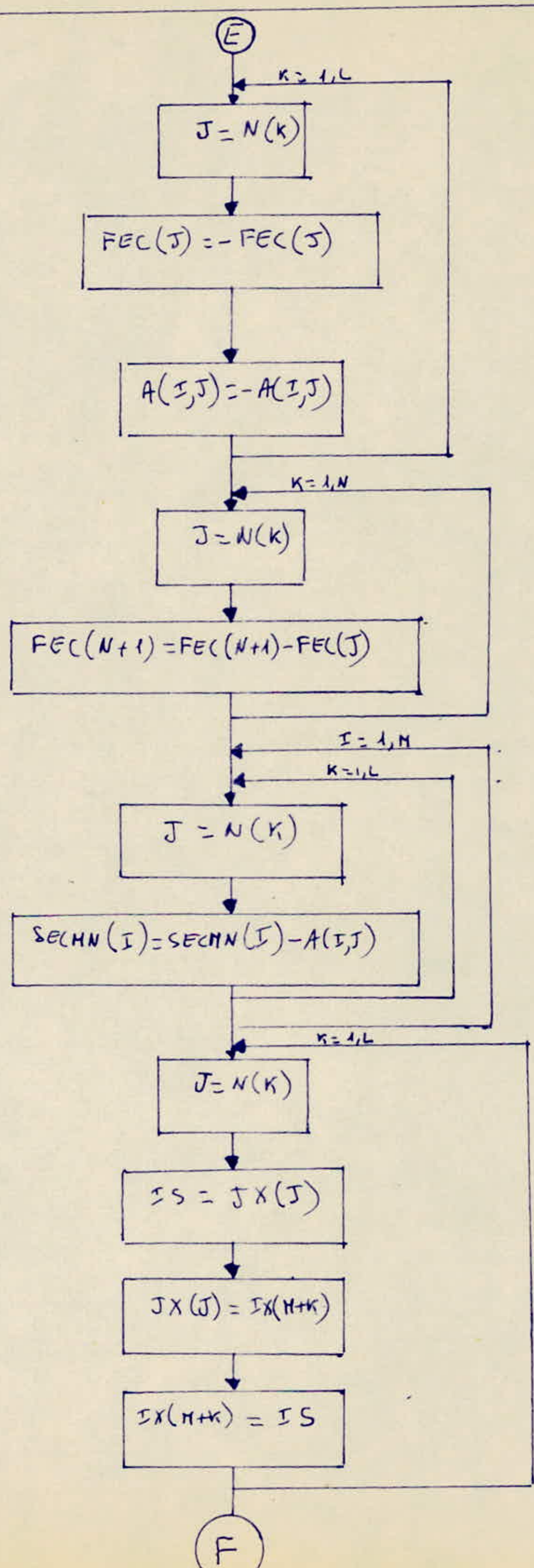
- TEST DE LA J-ADMISSIBILITÉ -



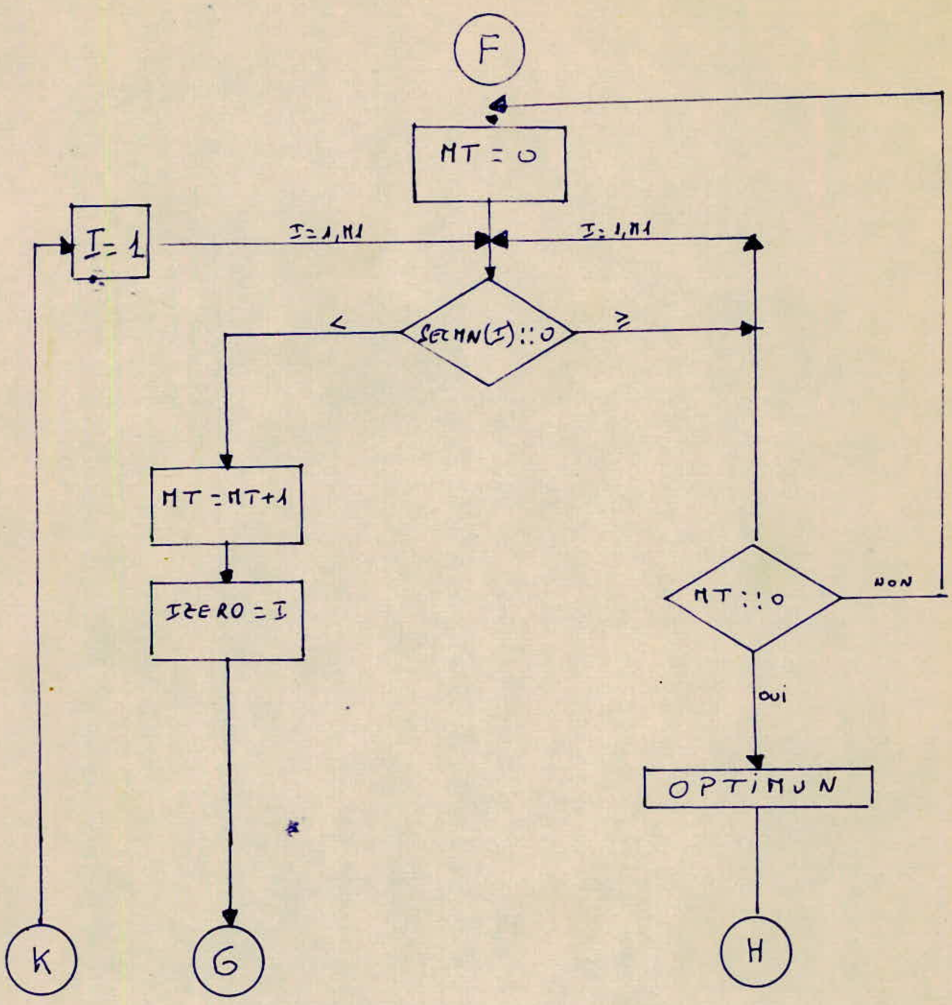
INSTALLATION DE LA PHASE I



PHASE I. - PIVOTAGE AUTOUR DE LA MATRICE UNITÉ -

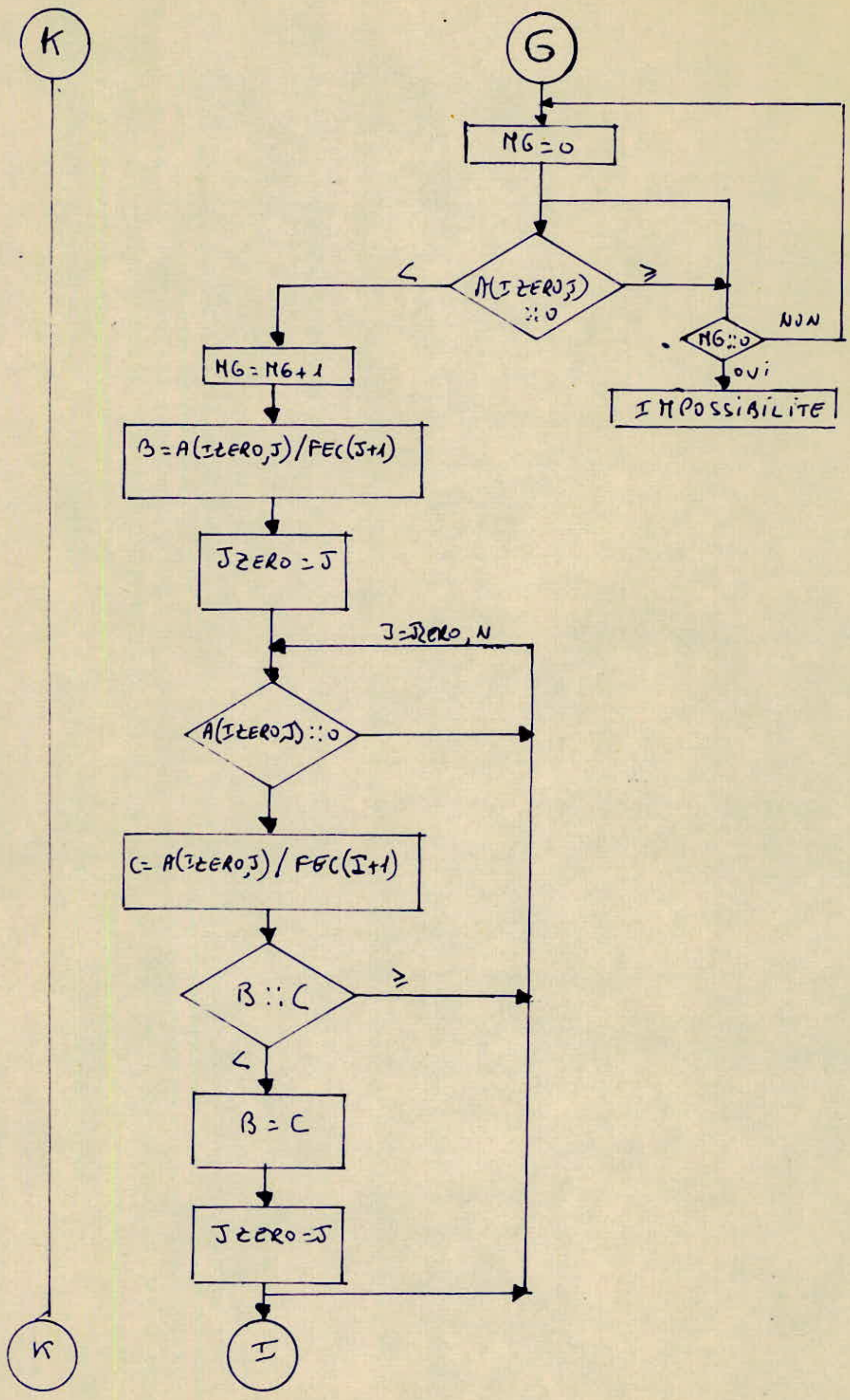


- CONTINU - I -

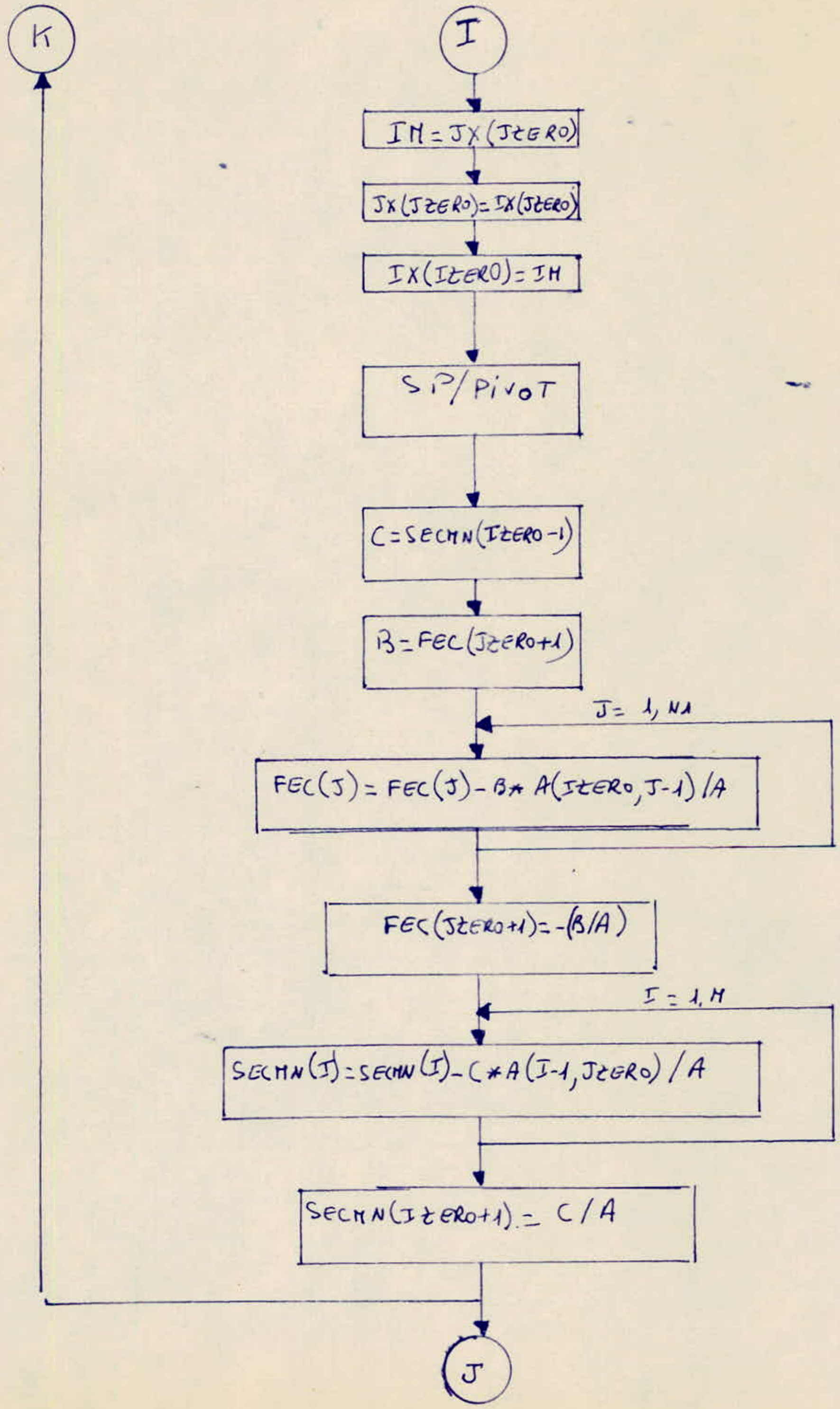


F
MT = 0
I = 1
I = 1, M1
I = 1, M1
SEMN(I) :: 0
<
≥
MT = MT+1
IZERO = I
MT :: 0
NON
OUI
OPTIMUN
K
G
H

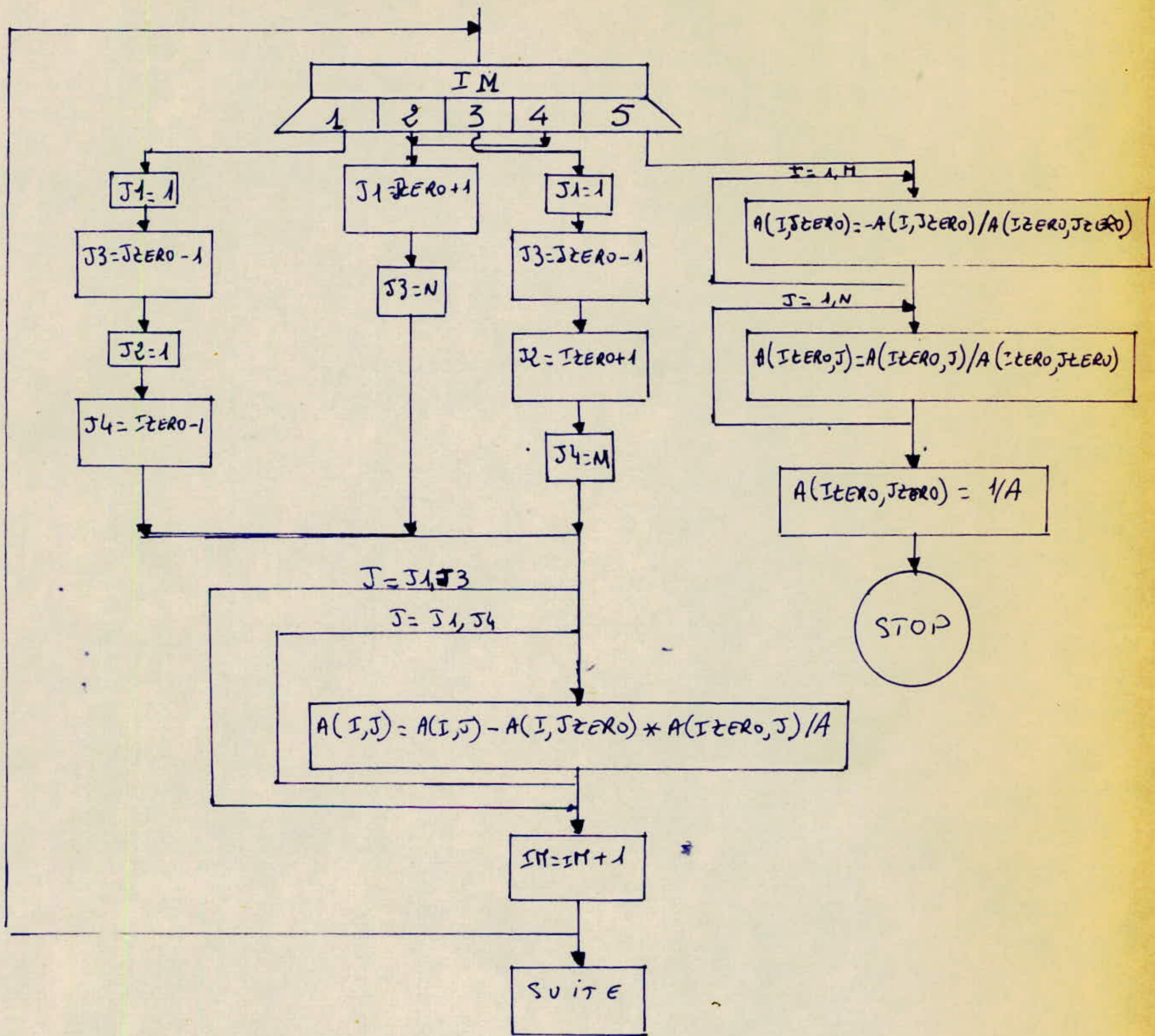
CONTINU - SUITE - II -



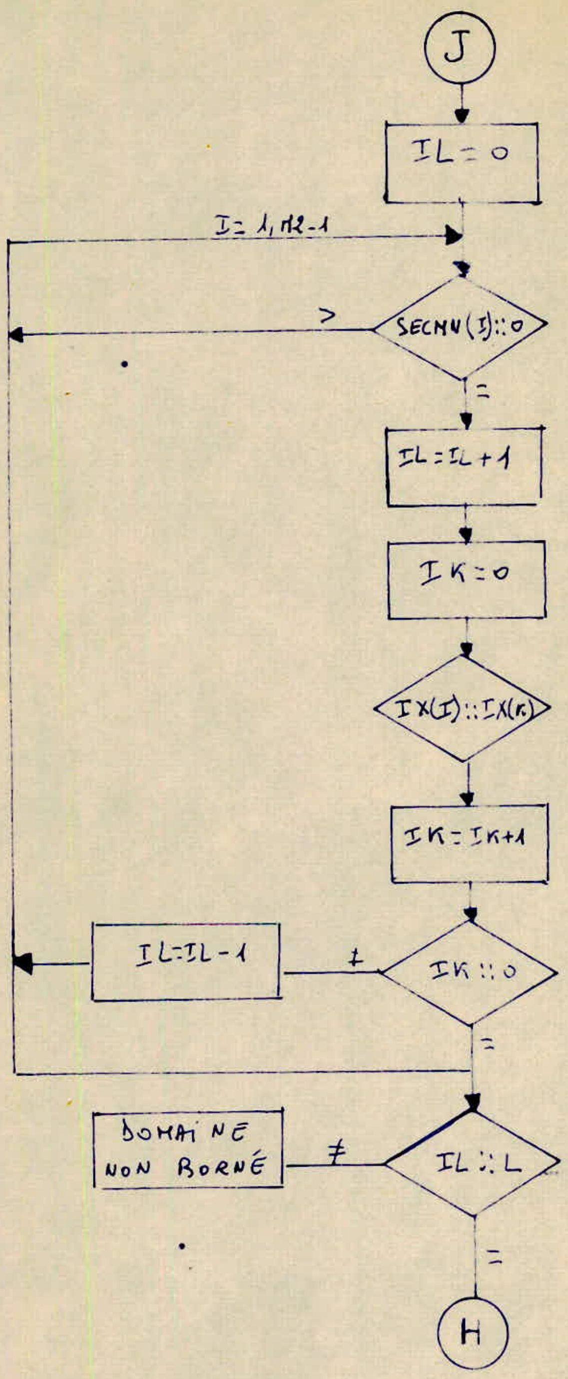
CONTINU - SUITE - III -



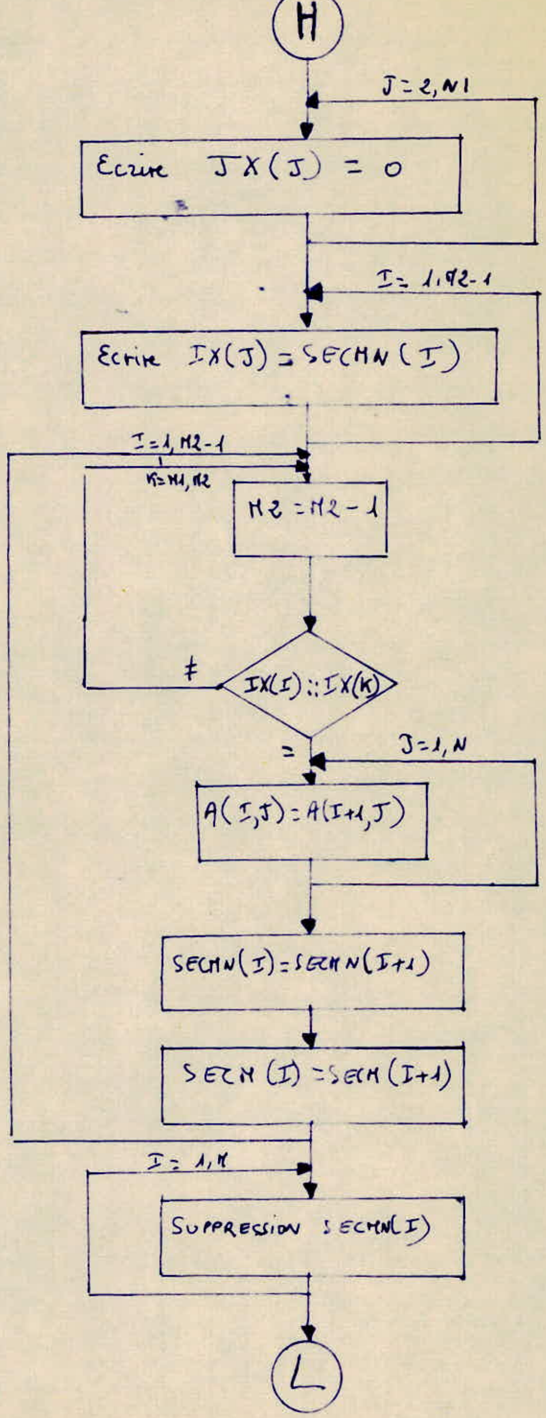
SOUS PROGRAMME PIVOT SP/PIVOT -



- TEST D'OPTIMALITÉ -



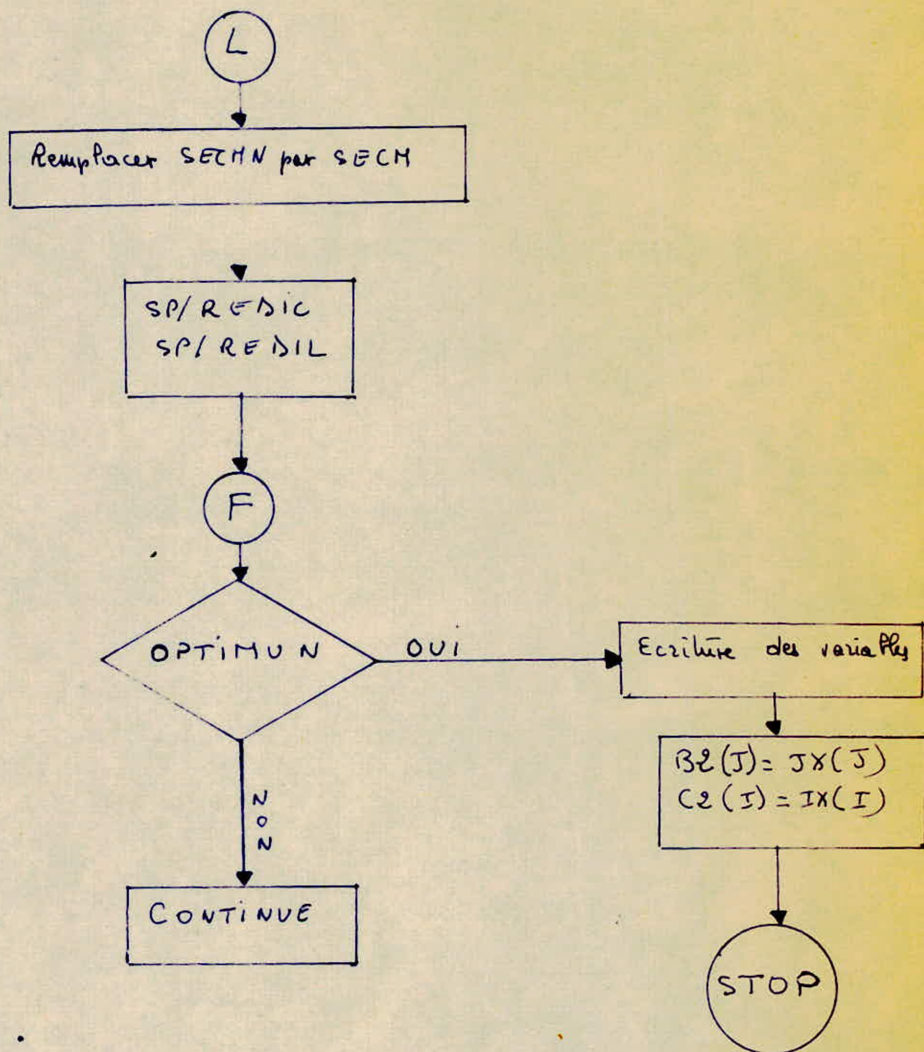
OPTIMUN ET REDUCTION



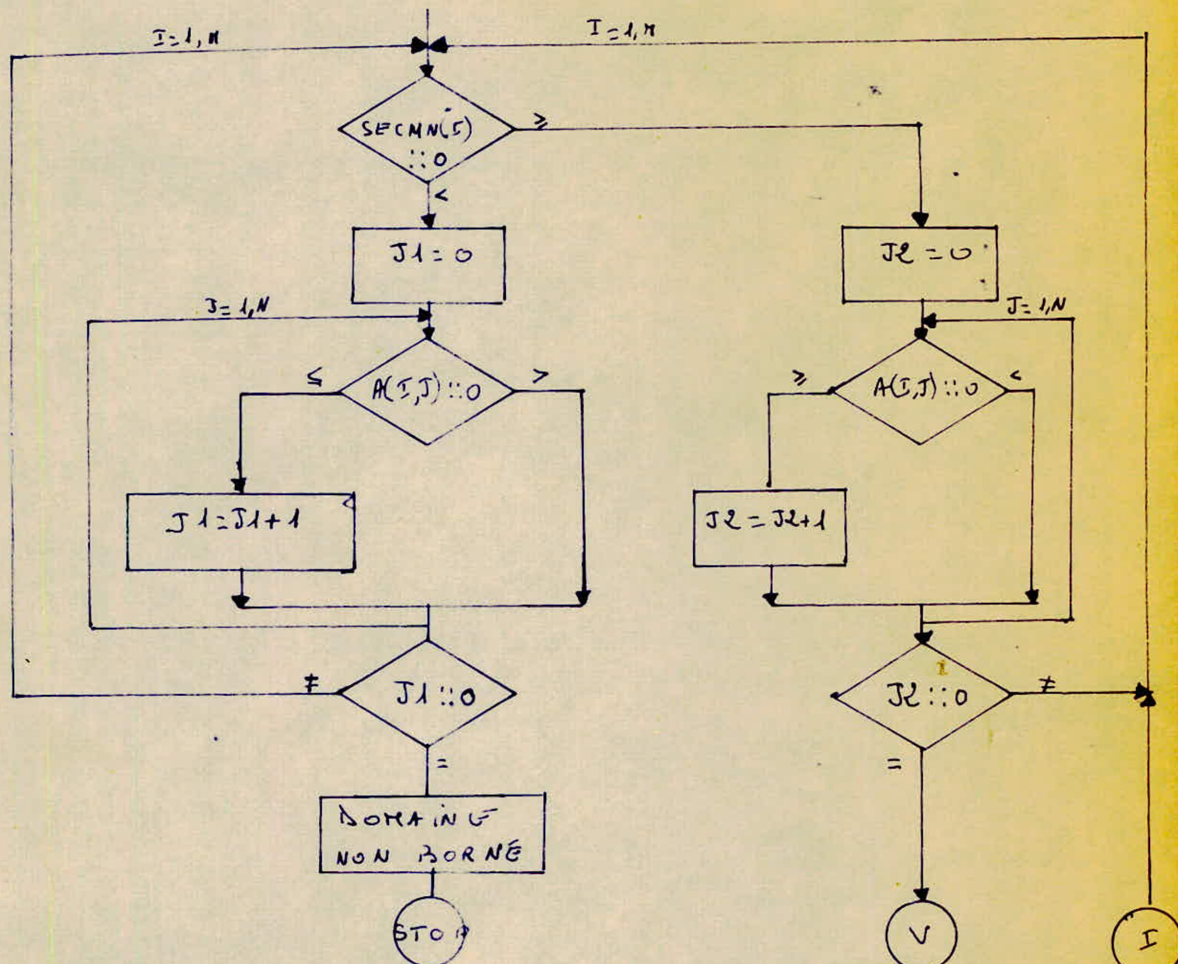
Ecriture des valeurs des variables à l'OPTIMUN.

REDUCTION

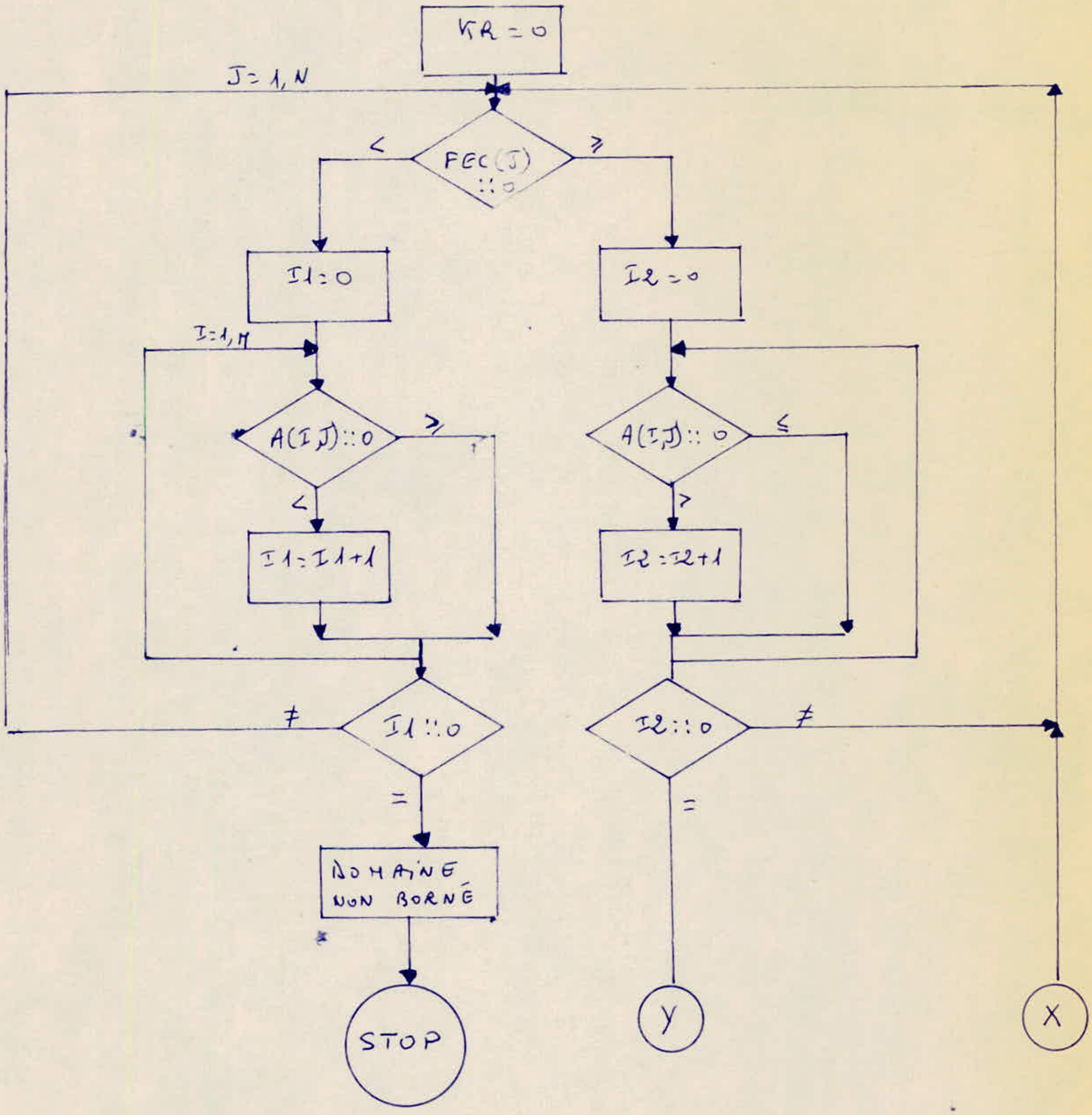
- PHASE II -


L
Remplacer SECHN par SECH
SP/REDIC
SP/REDIL
F
OPTIMUN
OUI
NON
CONTINUE
Ecriture des variables
B2(J) = JX(J)
C2(I) = IX(I)
STOP

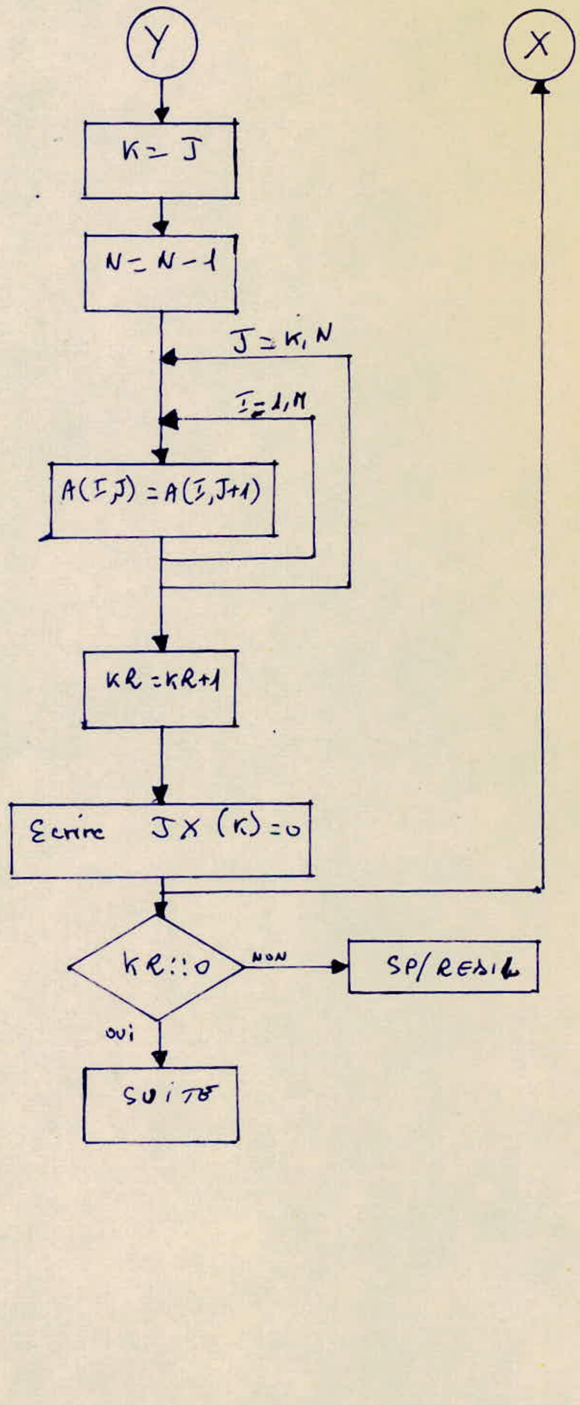
SOUS PROGRAMME · REDONDANCE _ IMPOSSIBILITÉ par LIGNE .



SOUS PROGRAMME REDONDANCE IMPOSSIBILITÉ COLONNE .

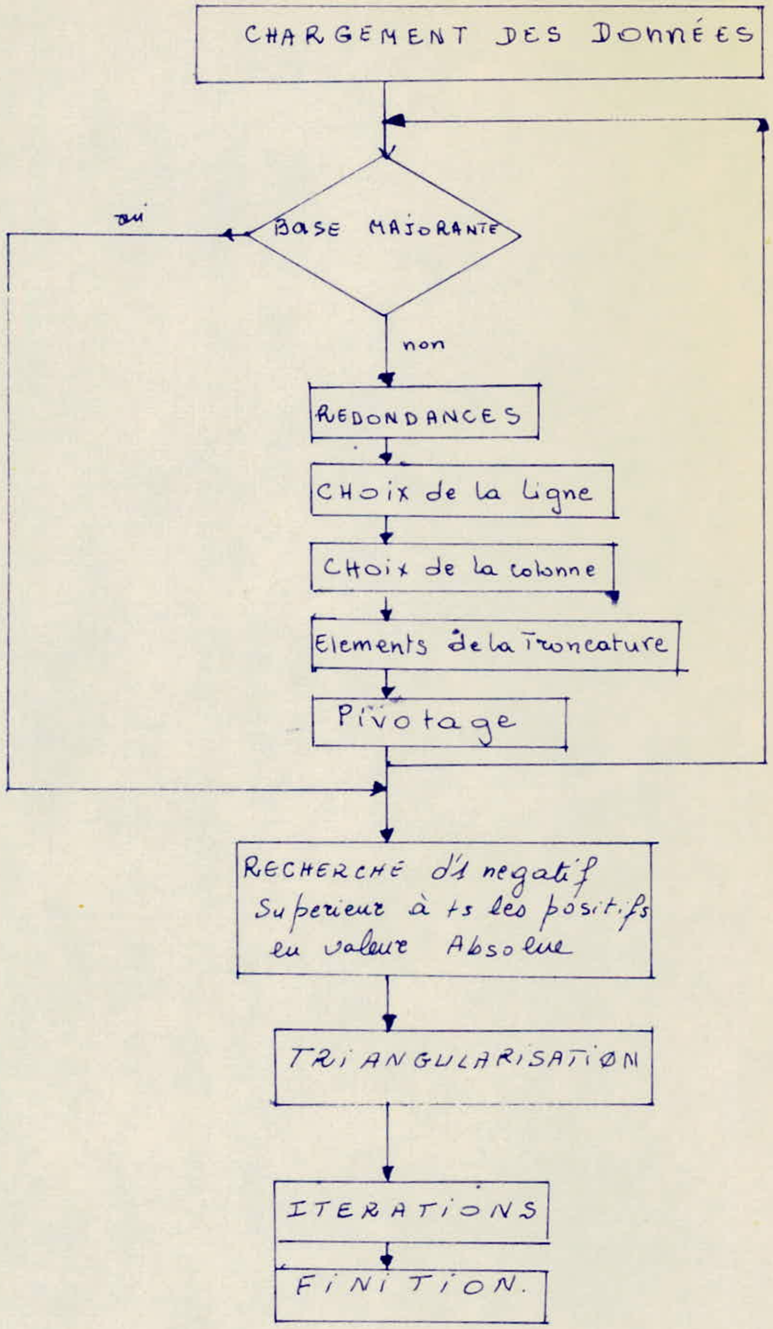


SOUS PROGRAMME REDONDANCE IMPOSSIBILITÉ COLONNE - SUITE -

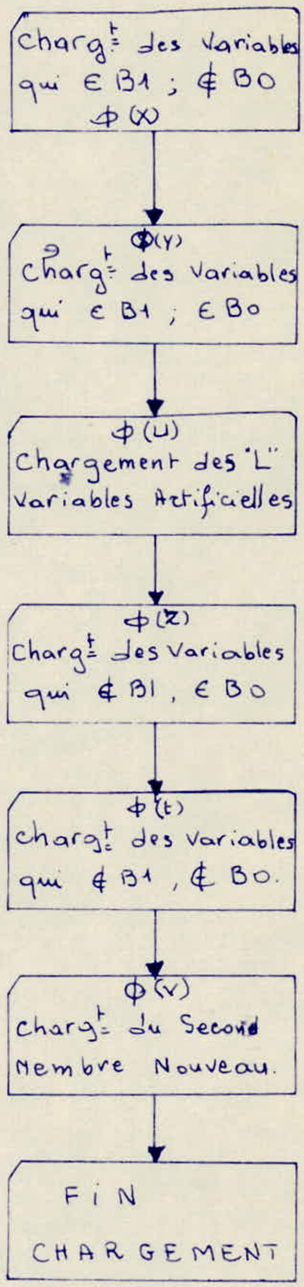


Recherche de la Solution en Nbres Entiers.

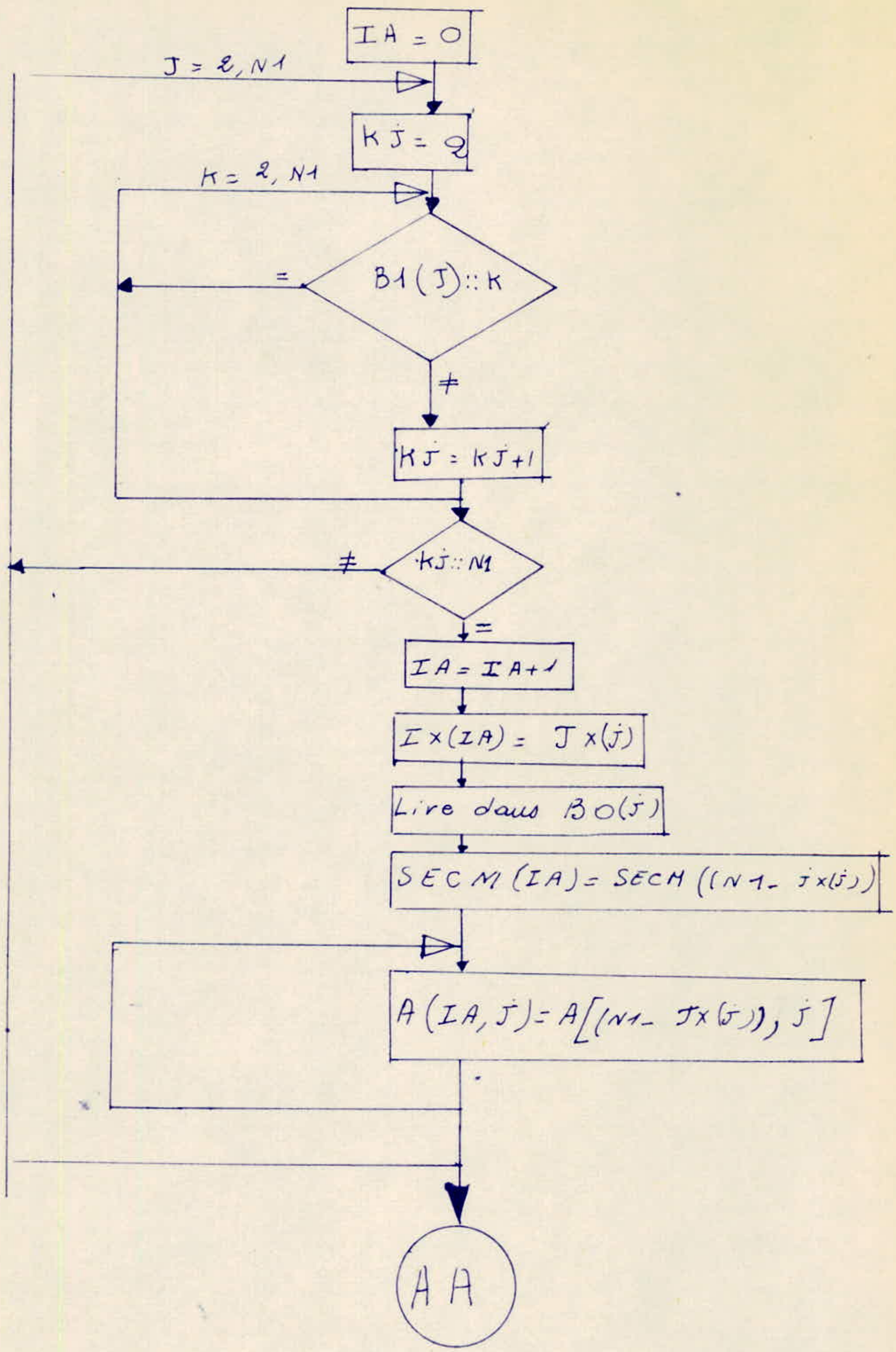
Organigramme general:



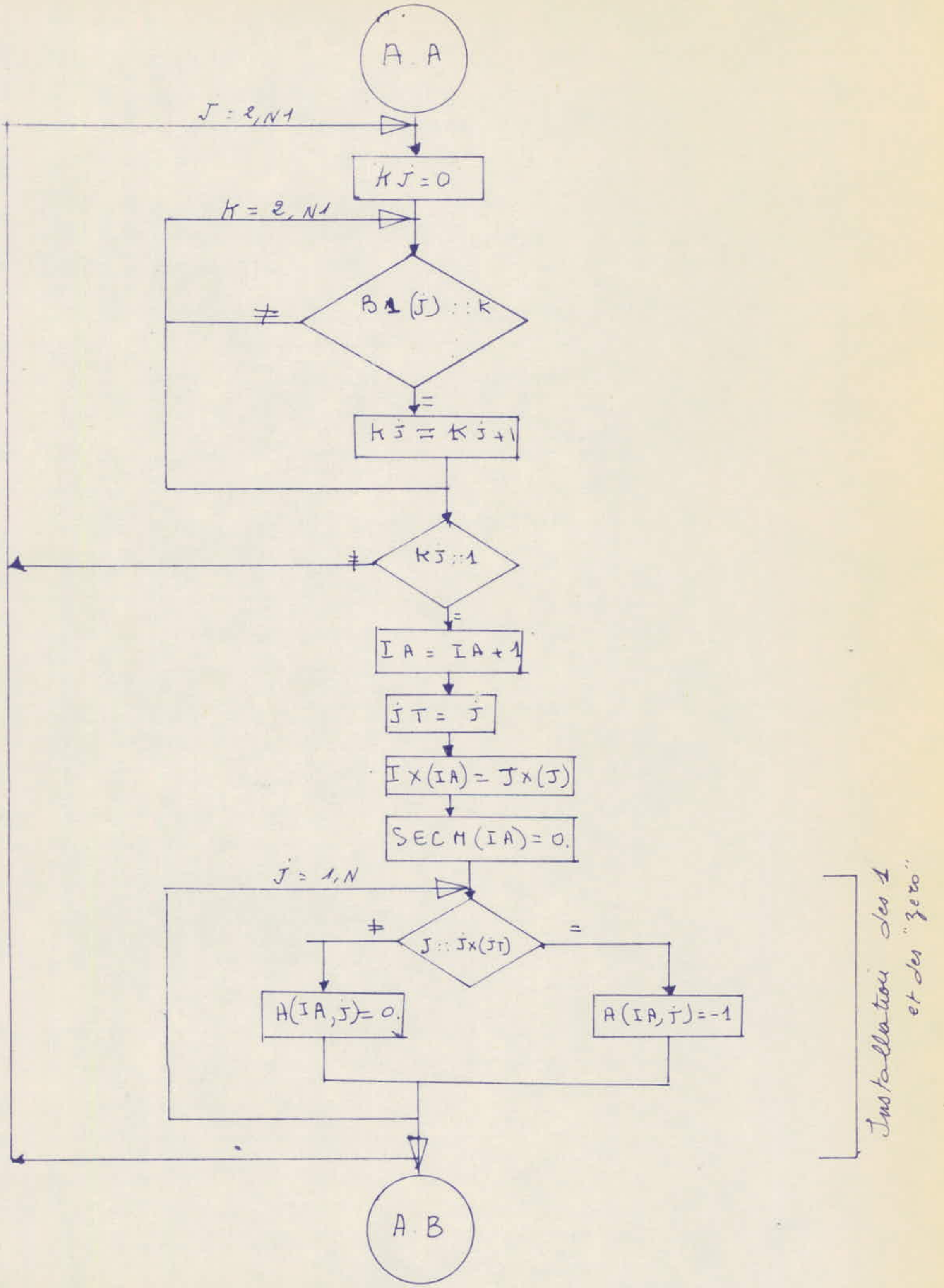
CHARGEMENT DES DONNÉES



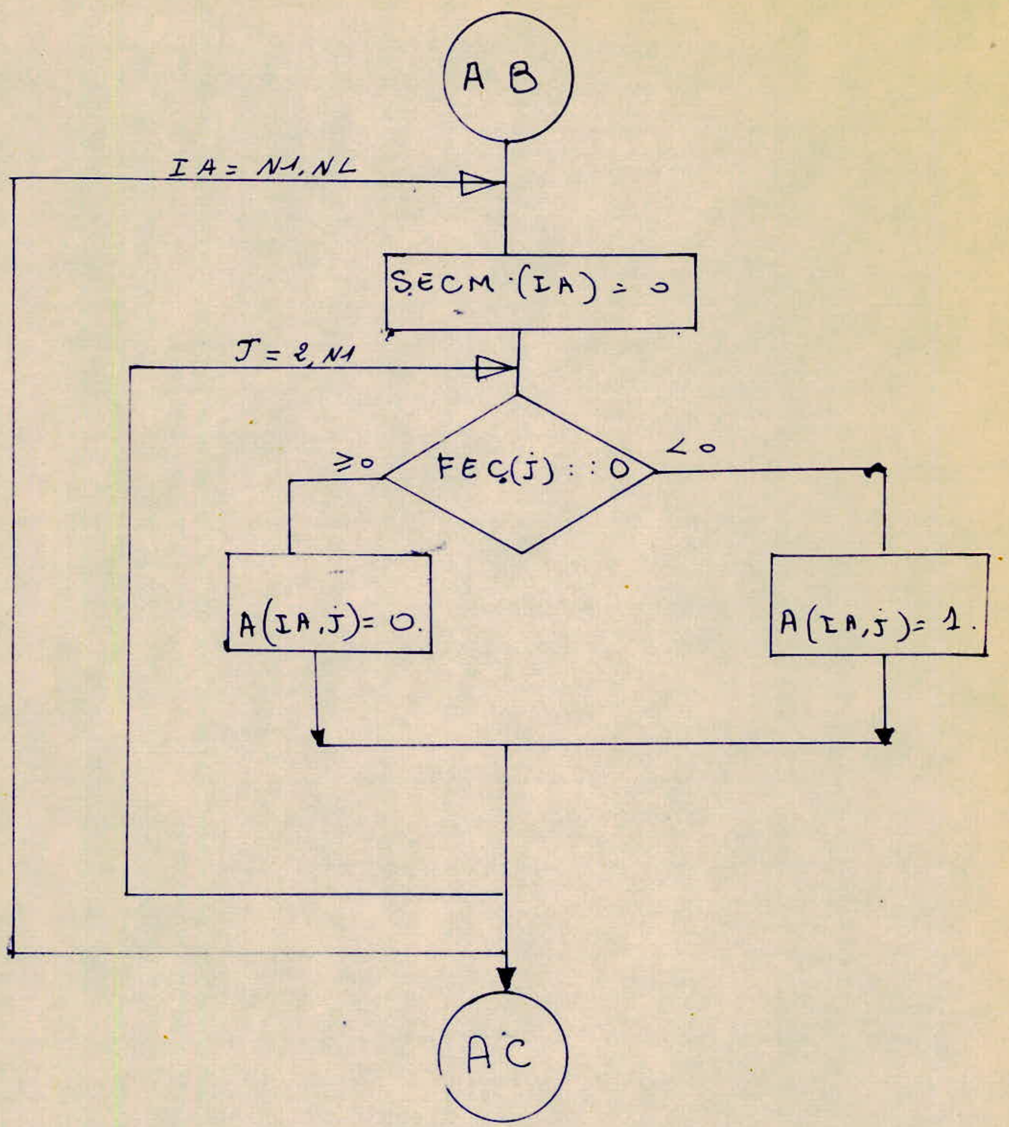
CHARGEMENT DES DONNÉES $\Phi(x)$



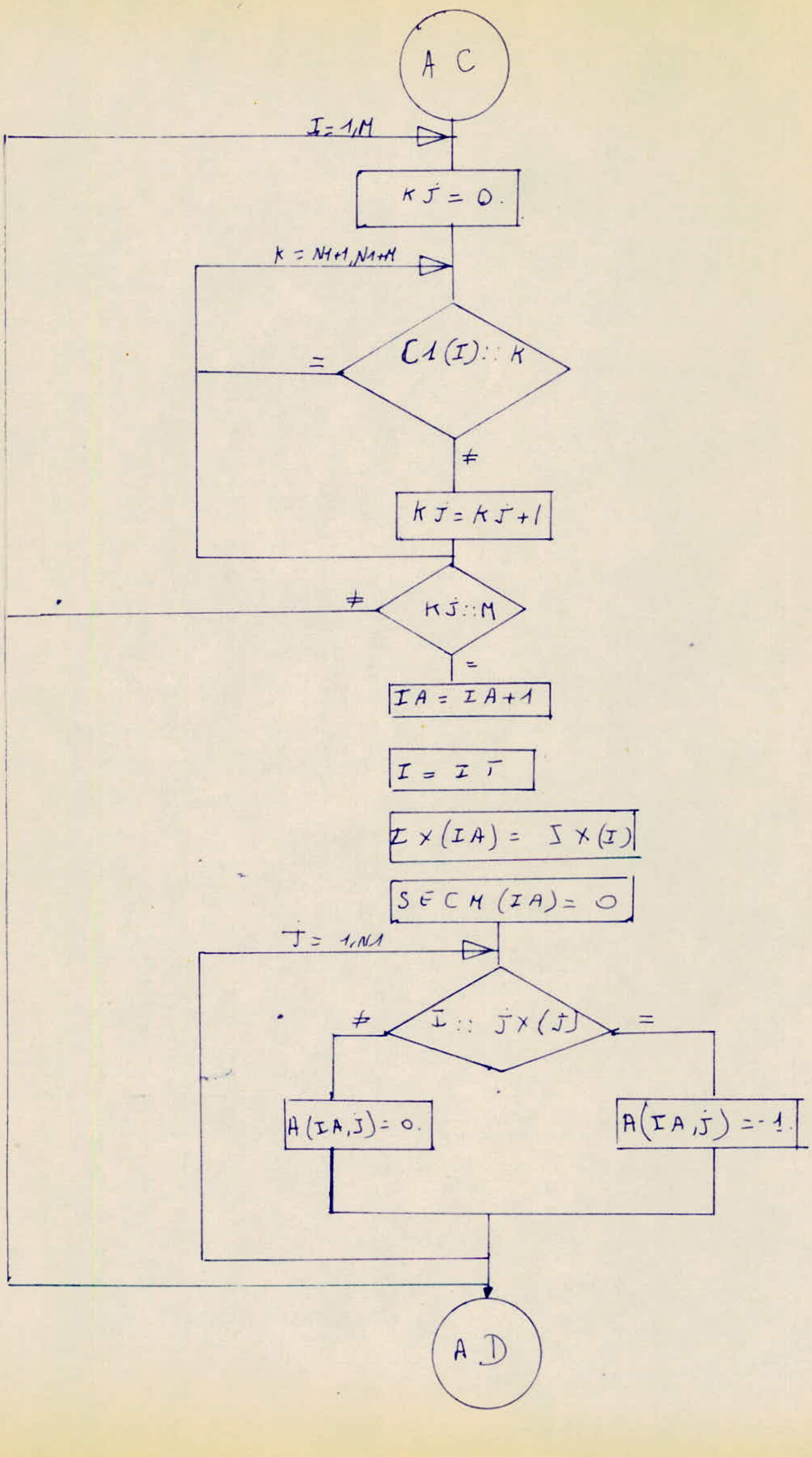
ΦY Chargement des données.



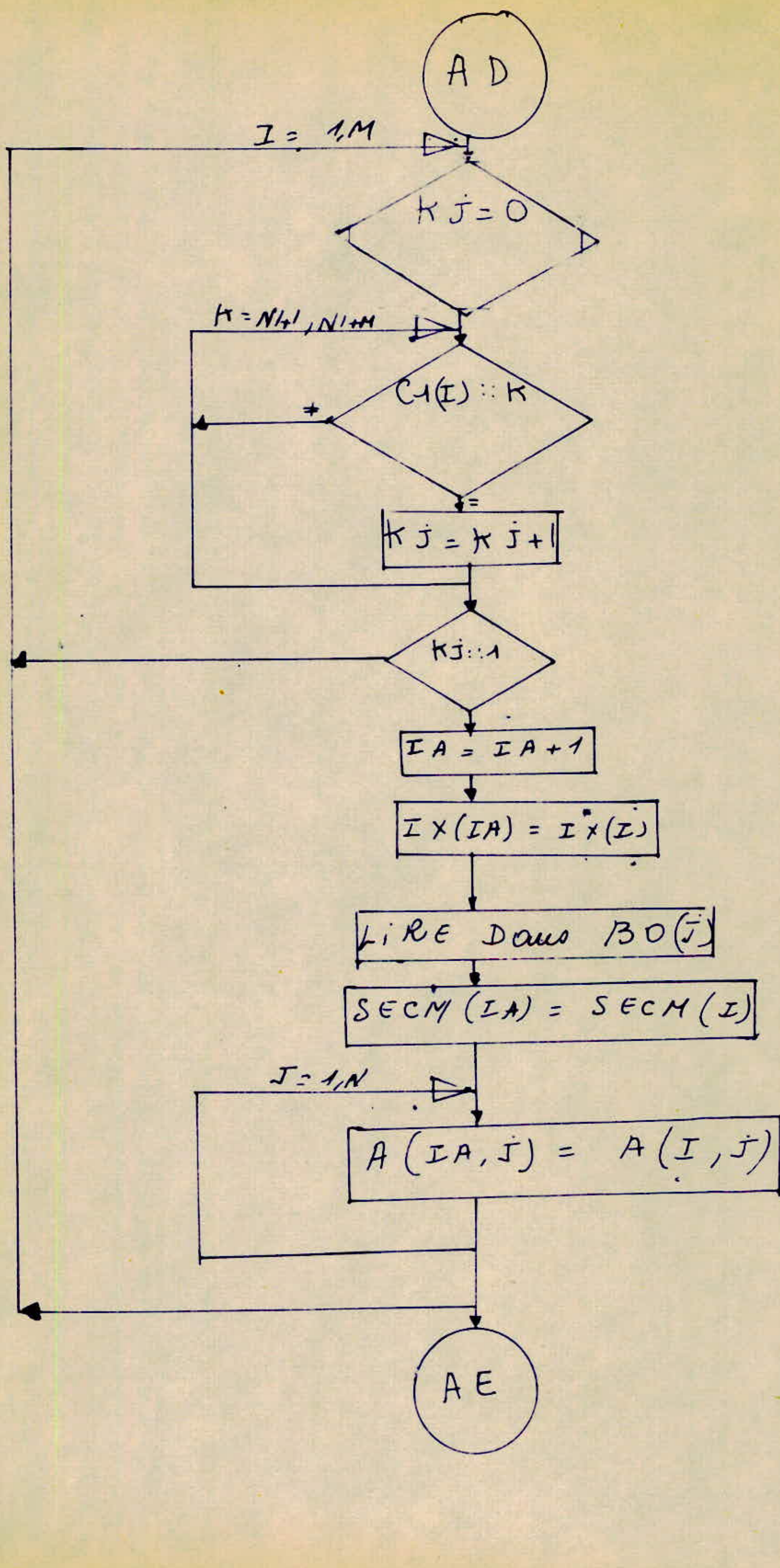
ϕ(4) Chargement des données



Φ(2) Chargement des Données :

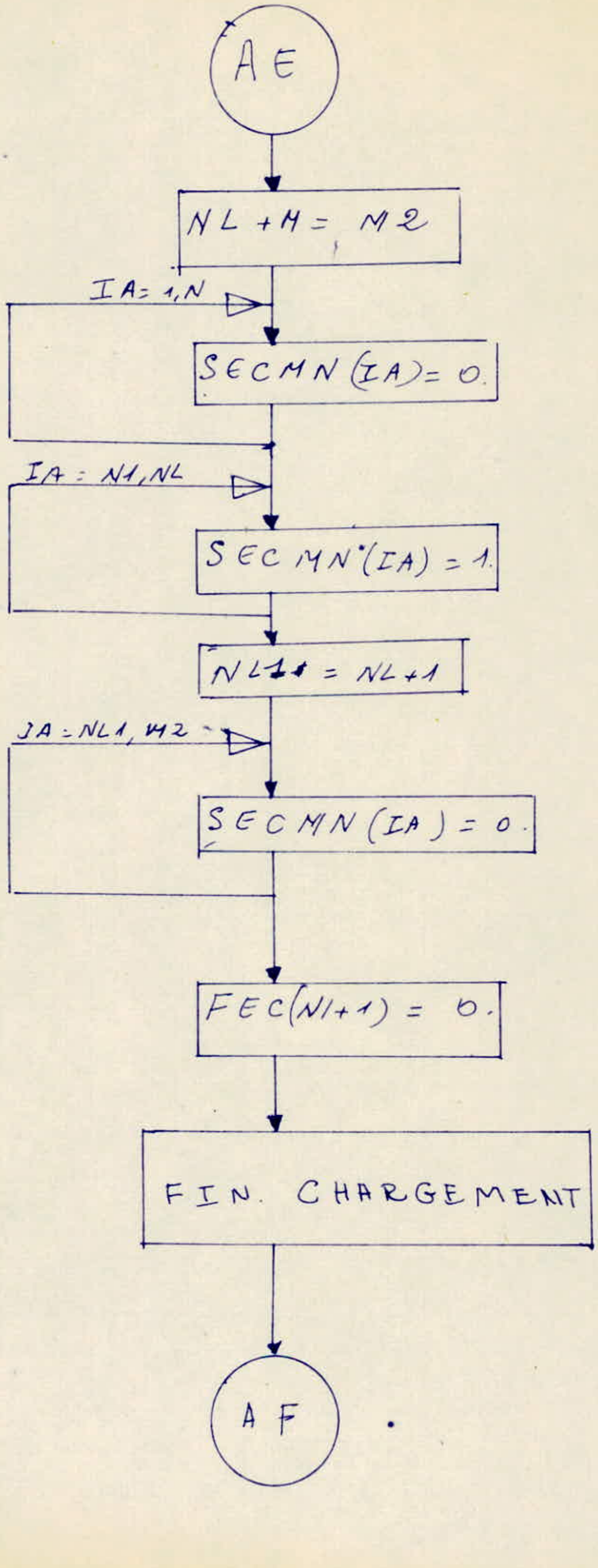


Chargement des données

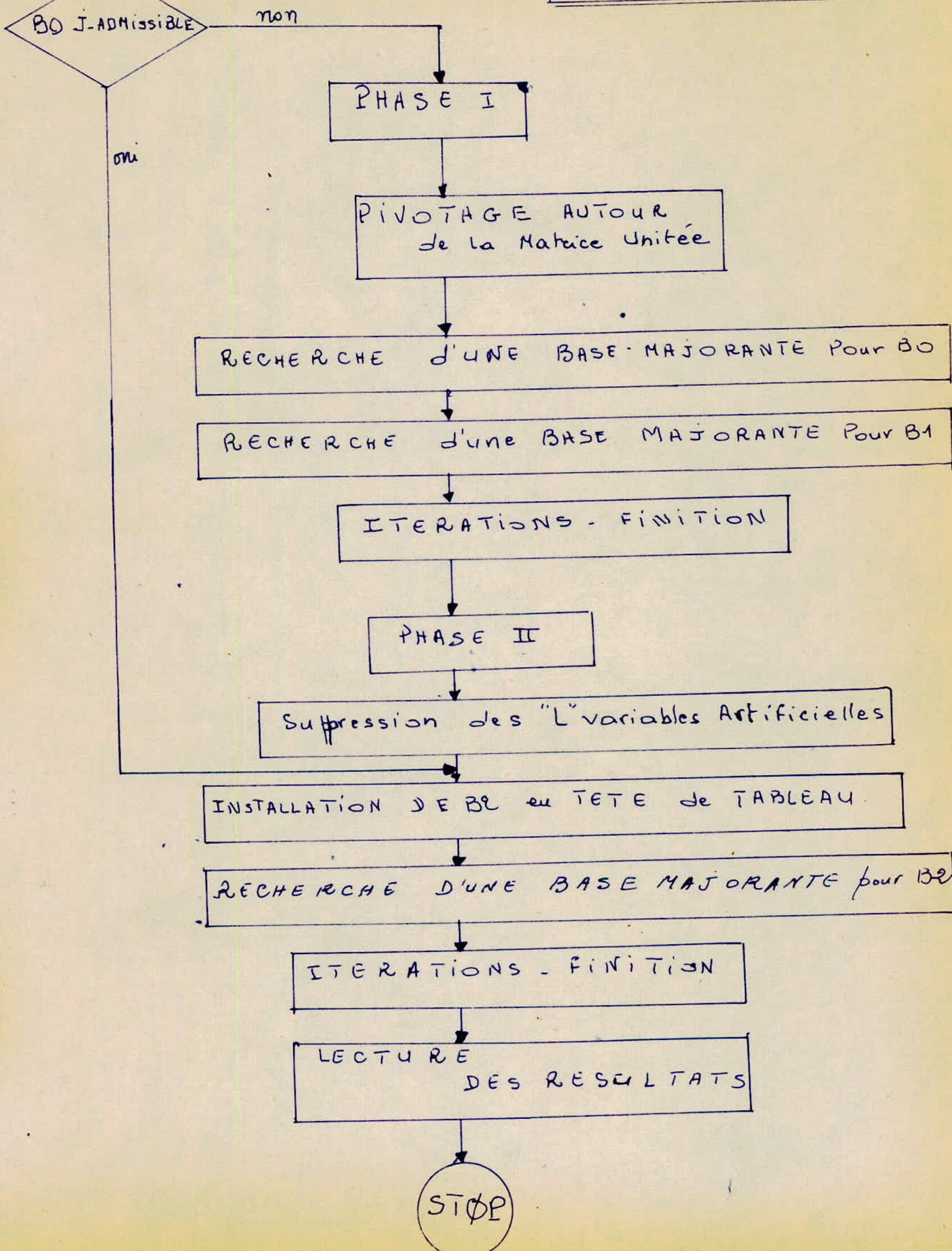


Chargement des données

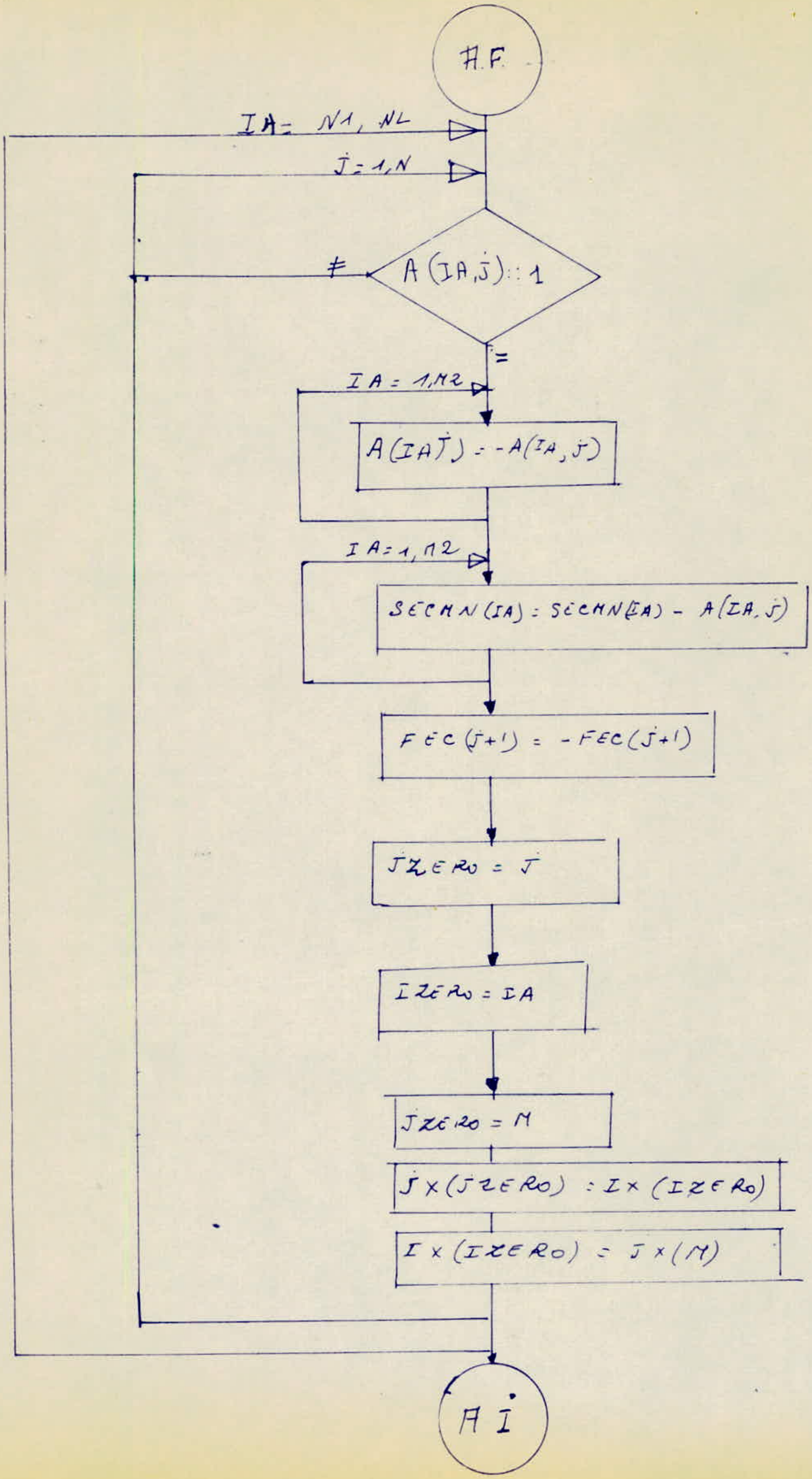
φ (V)



Adaptation de la methode à un probleme Non J- Admissible

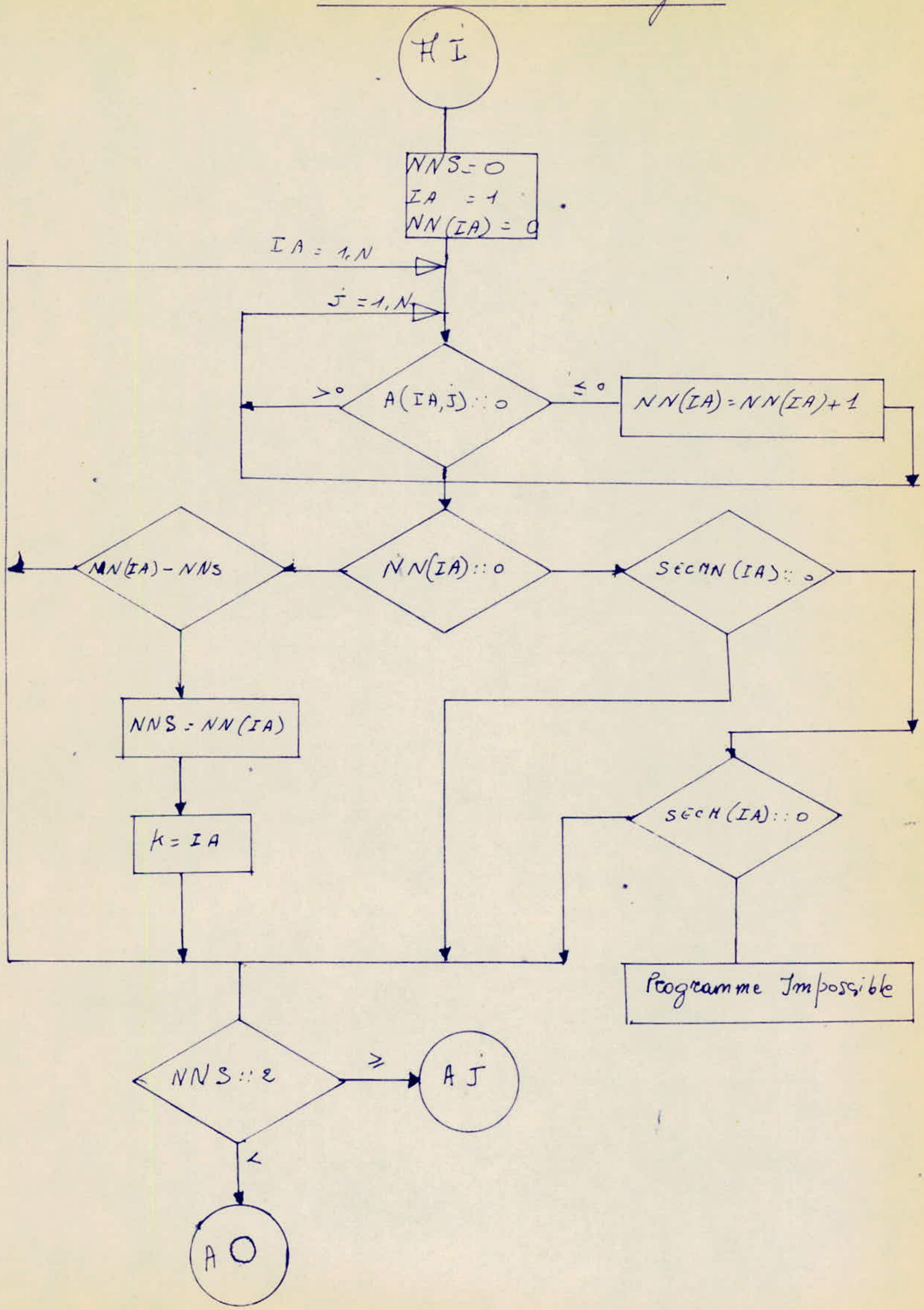


Pivotage autour de la matrice Unitée



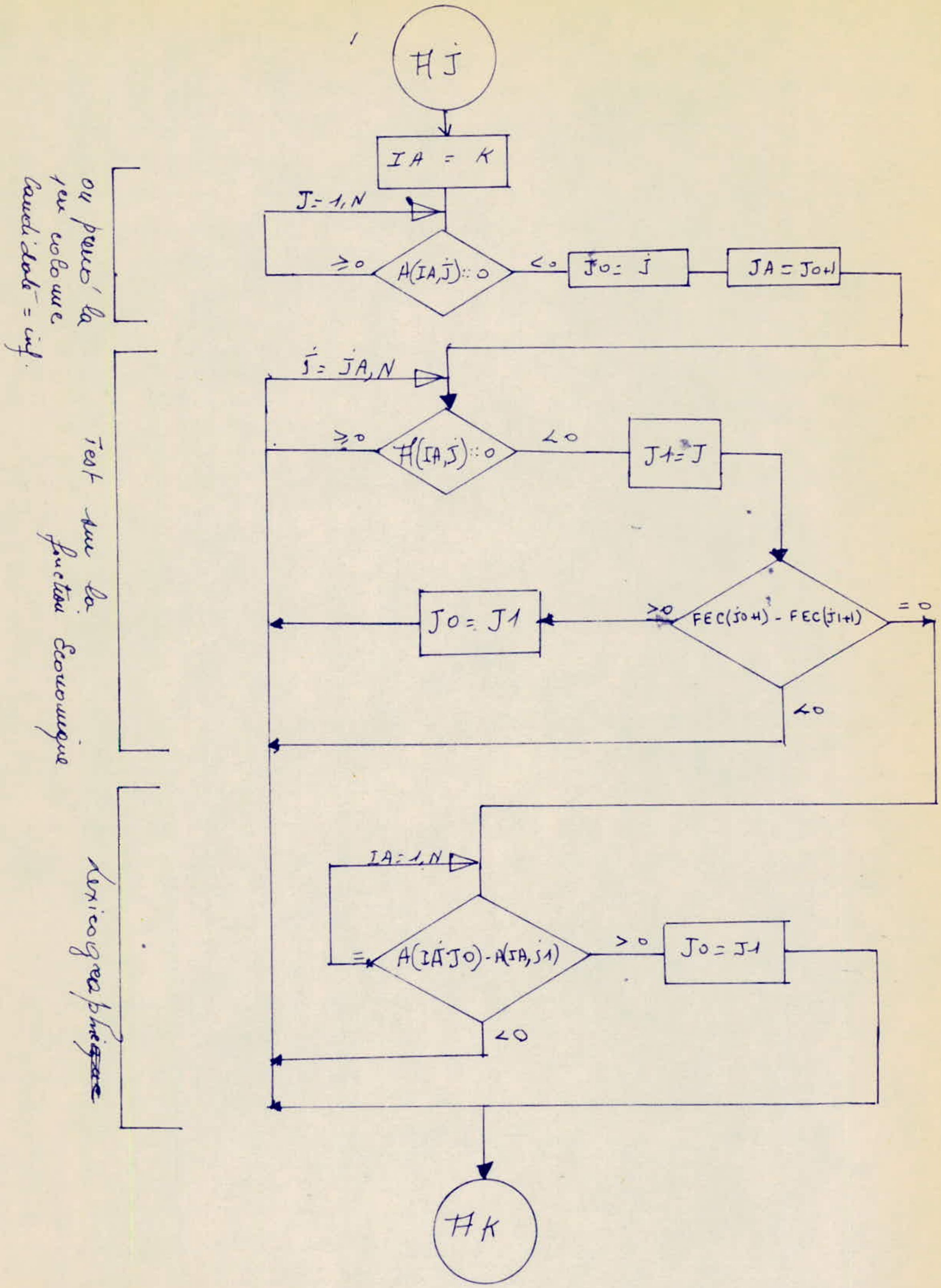
Recherche de la 1ère Base Majorante

Choix de la ligne



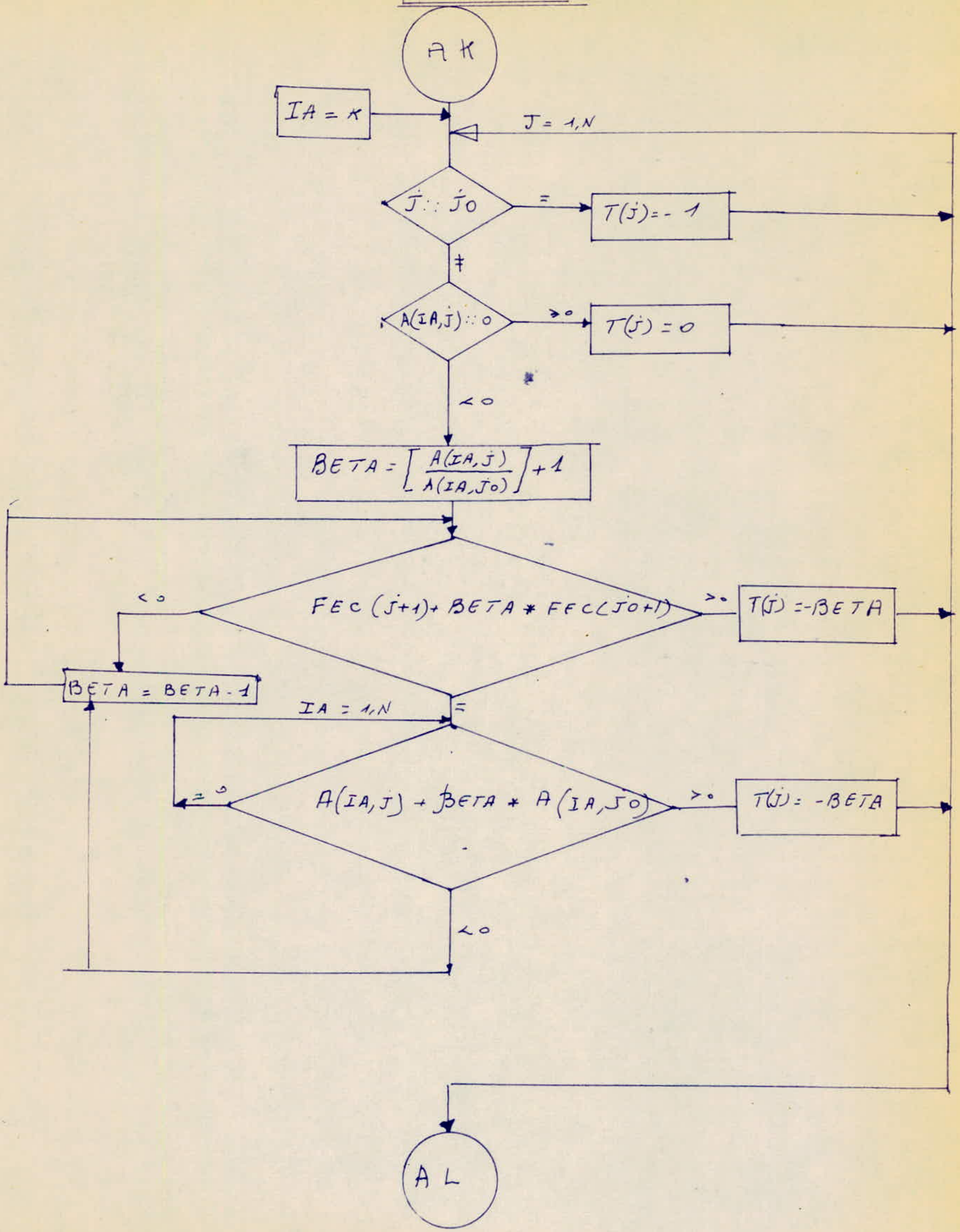
Recherche de la 1re Base Réalisante

Choix de la colonne



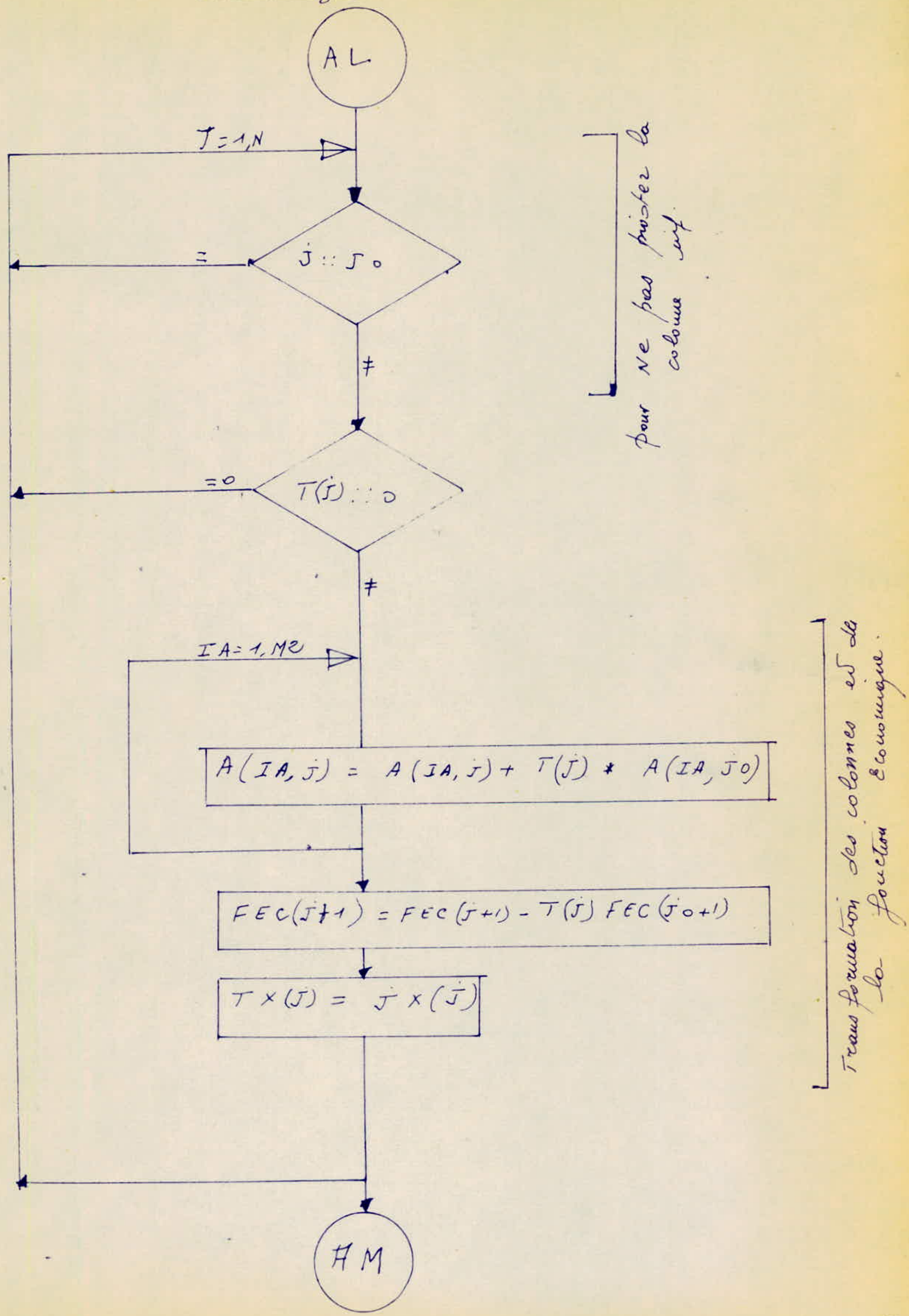
Recherche de la 1ère Base Majorante

Troncature

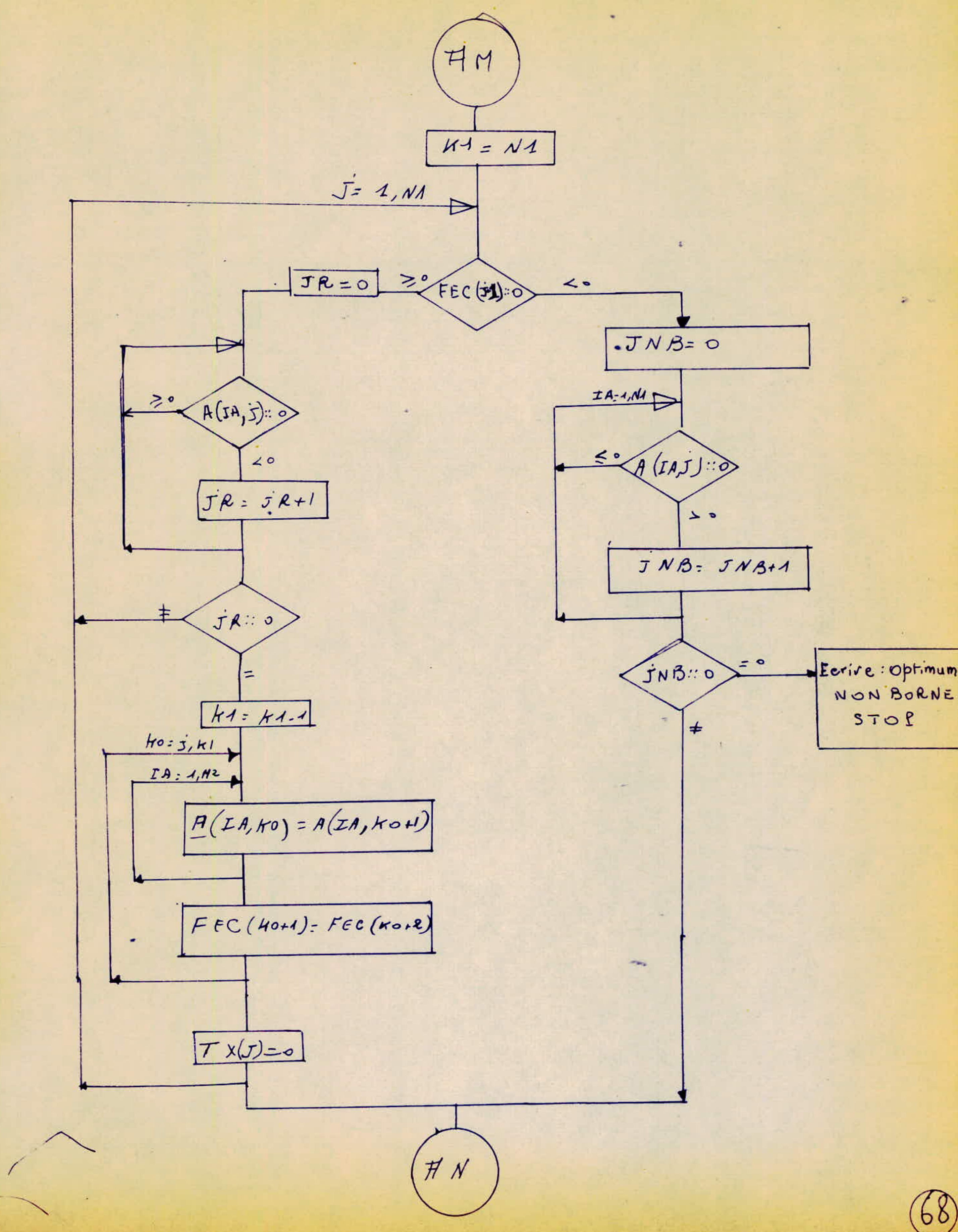


Recherche de la 1ère Base Majorante

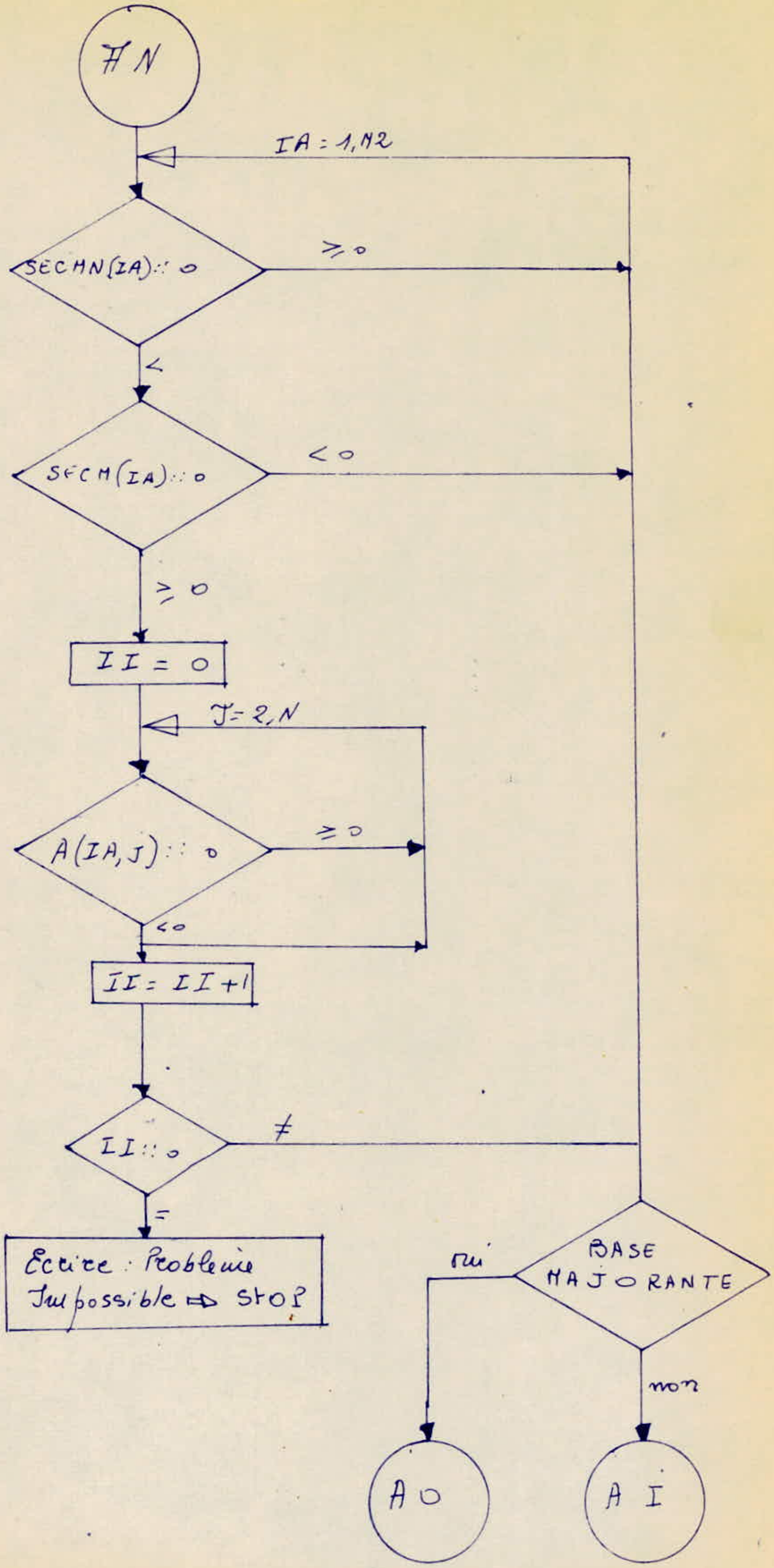
Pivotage :



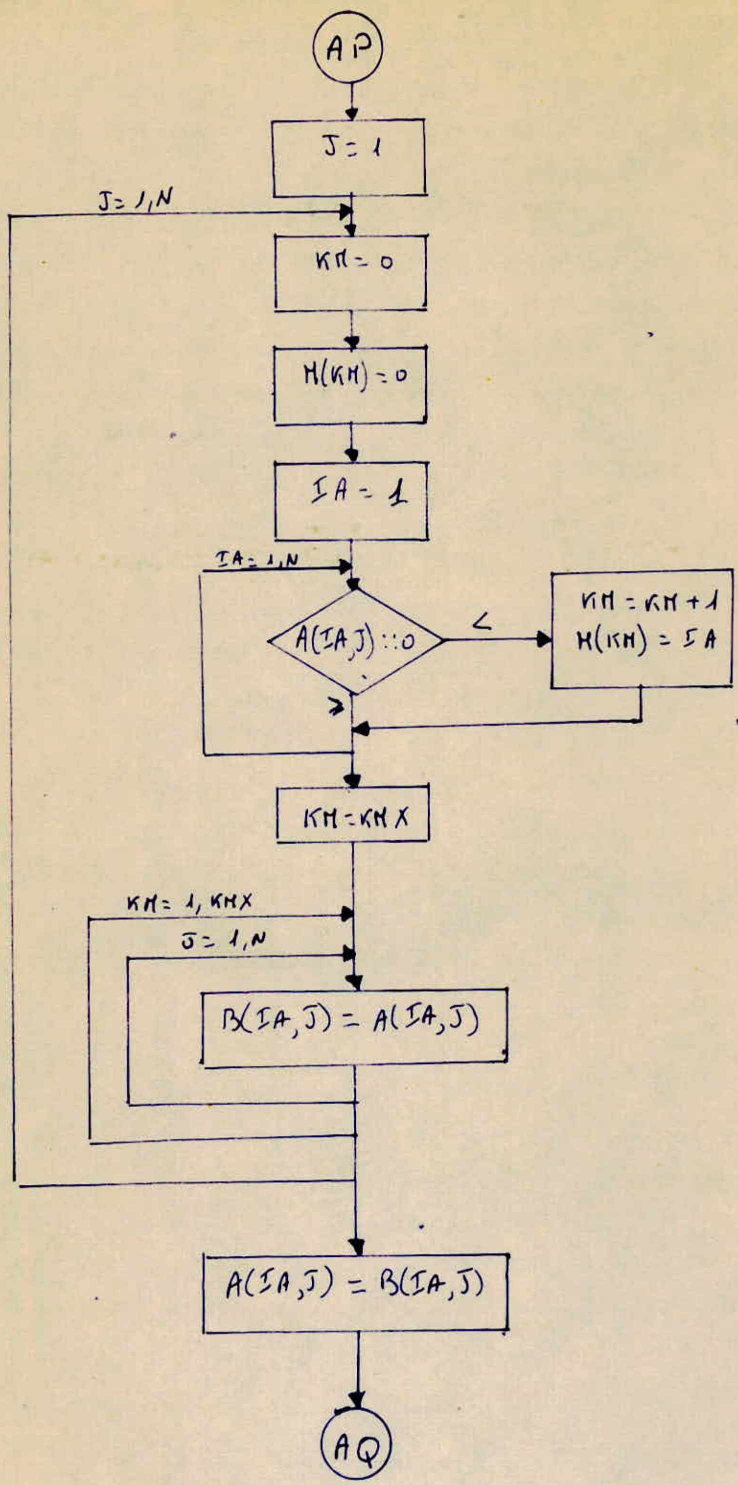
SP/ Colonnes Redondantes et Non Bornées



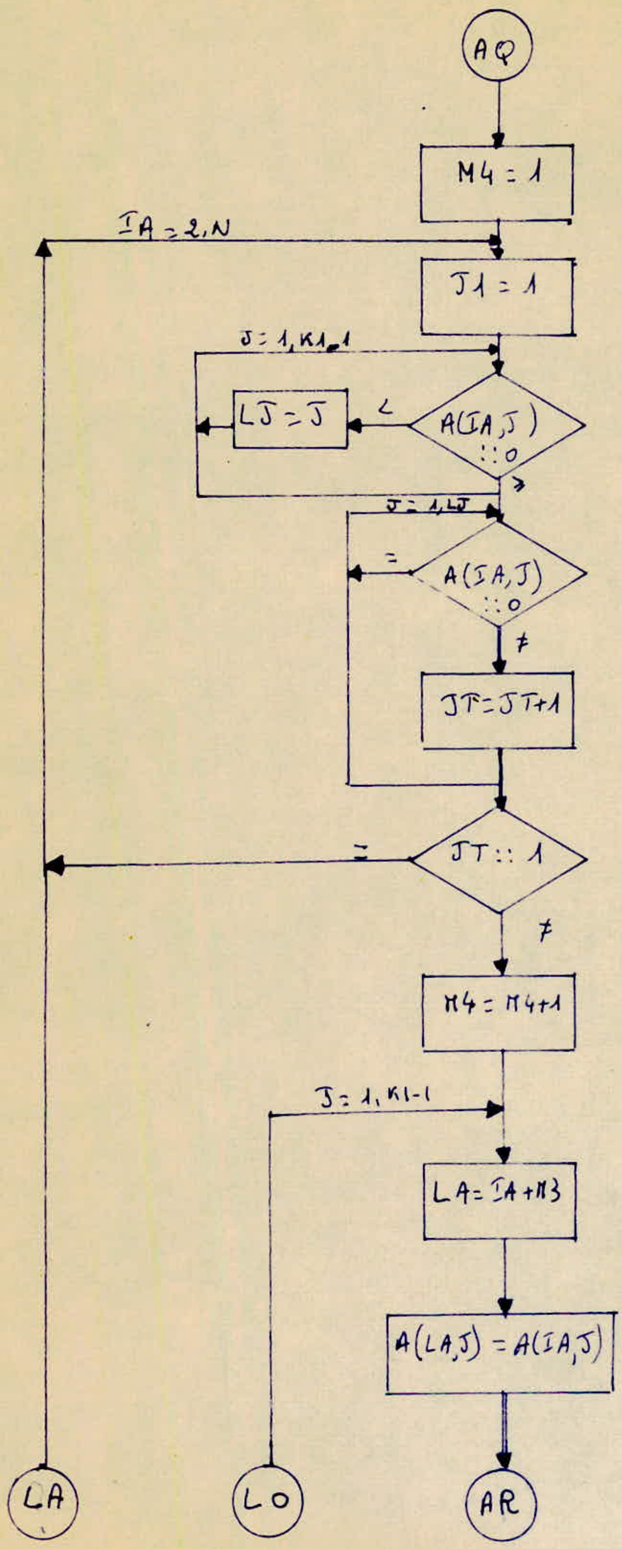
Lignes Impossibles :



CLASSEMENT

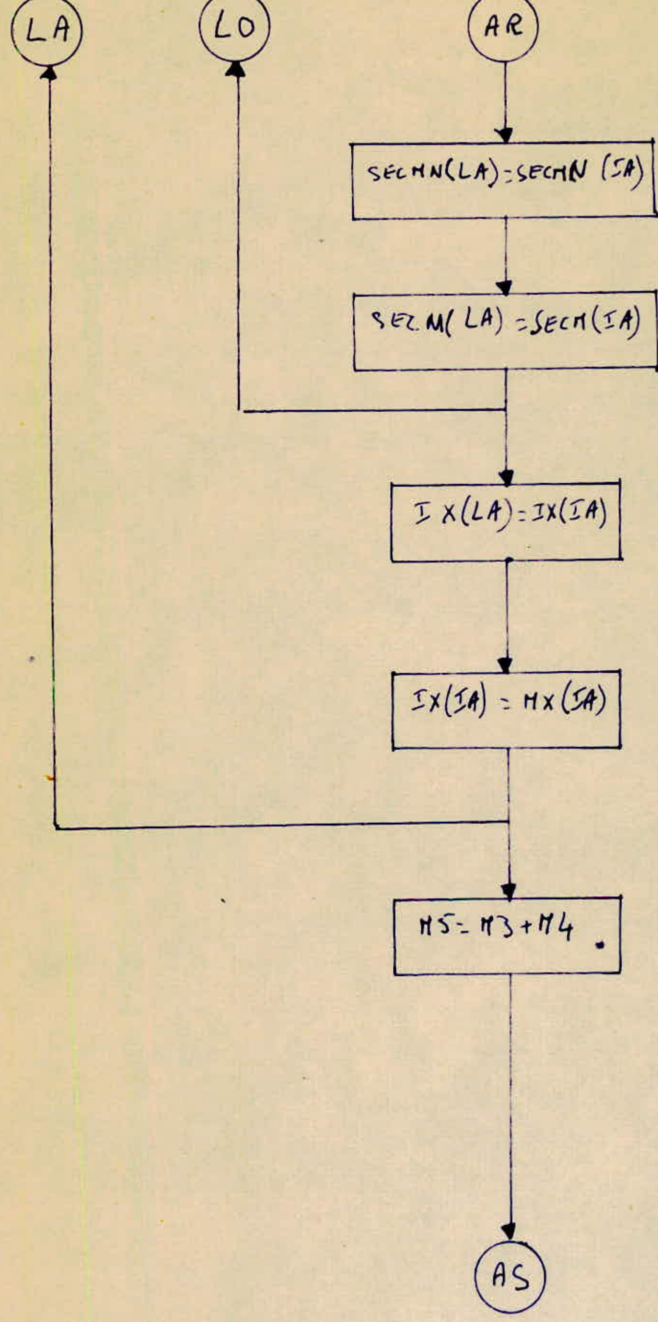

AP
J = 1
J = 1,N
KM = 0
M(KM) = 0
IA = 1
IA = 1,N
A(IA,J) :: 0
<
>
KM = KM + 1
M(KM) = IA
KM = KMX
KM = 1, KMX
J = 1,N
B(IA,J) = A(IA,J)
A(IA,J) = B(IA,J)
AQ

RANGEMENT

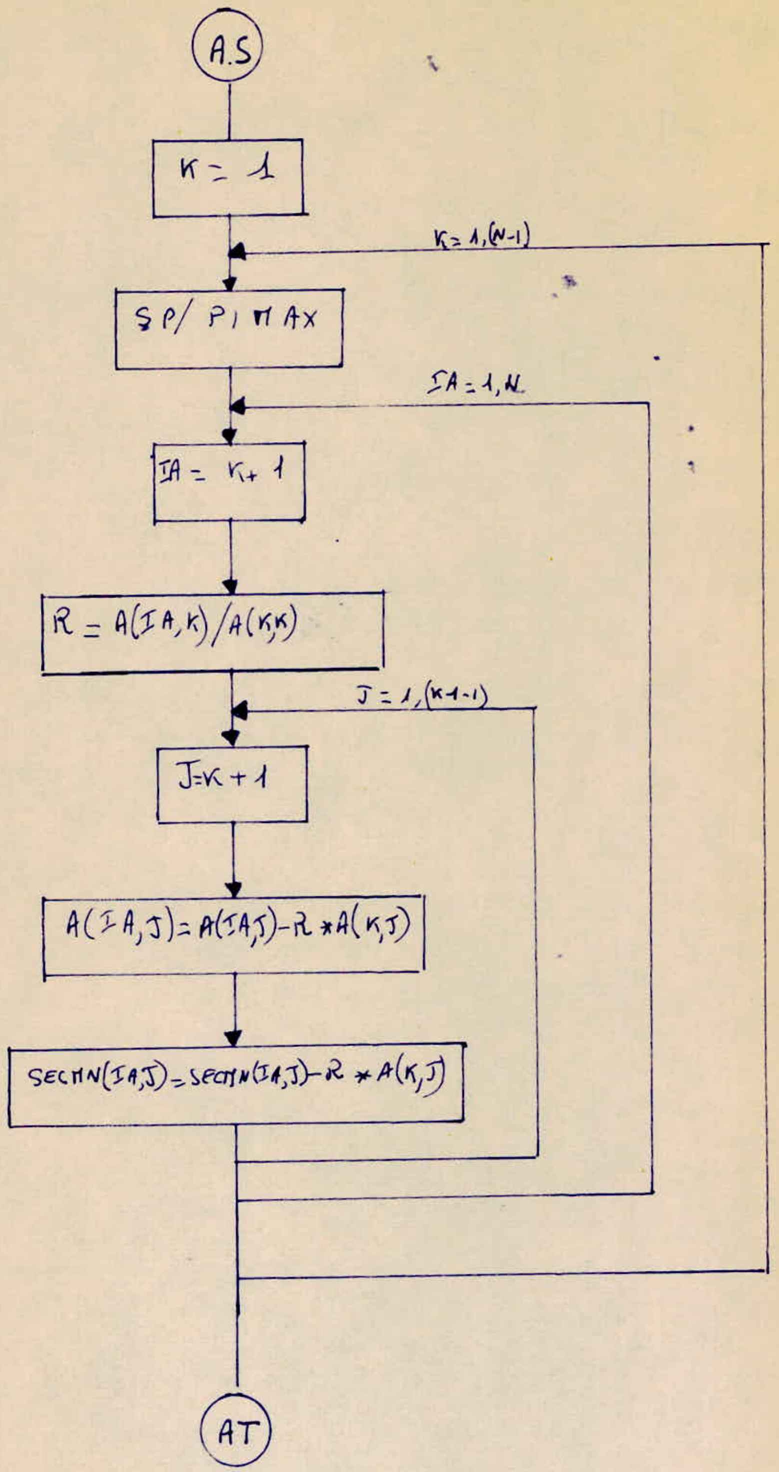


AQ
M4 = 1
IA = 2, N
J1 = 1
J = 1, K1-1
A(IA,J) :: 0
<
LJ = J
>
J = 1, LJ
A(IA,J) :: 0
=
≠
JT = JT+1
JT :: 1
=
≠
M4 = M4+1
J = 1, K1-1
LA = IA + M3
A(LA,J) = A(IA,J)
LA
LO
AR

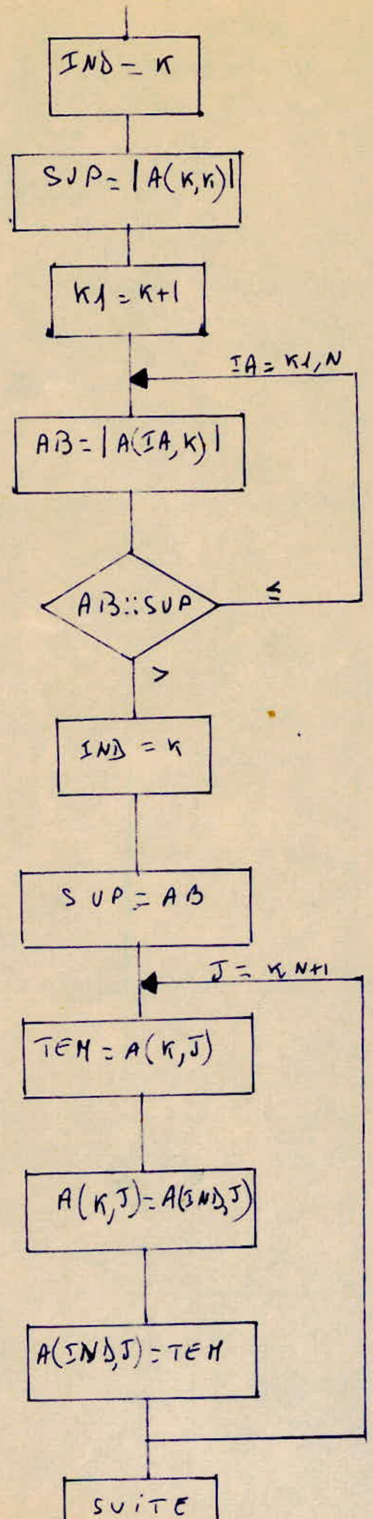
RANGEMENT - SUITE



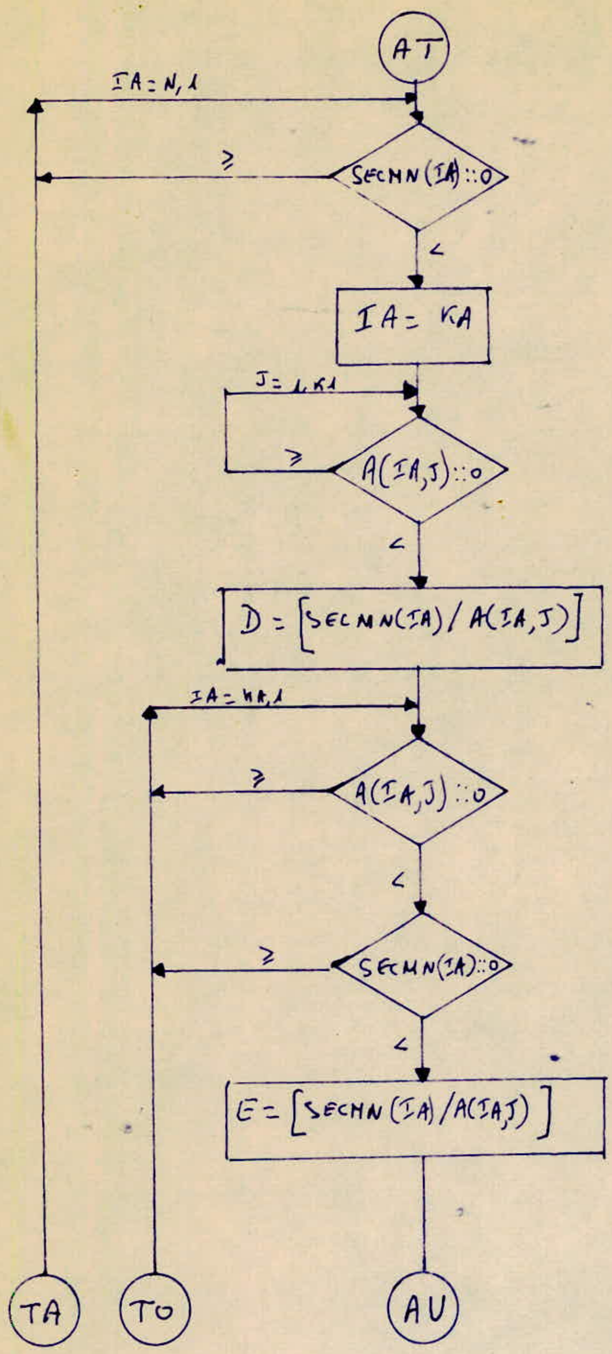
- TRIANGULARISATION -



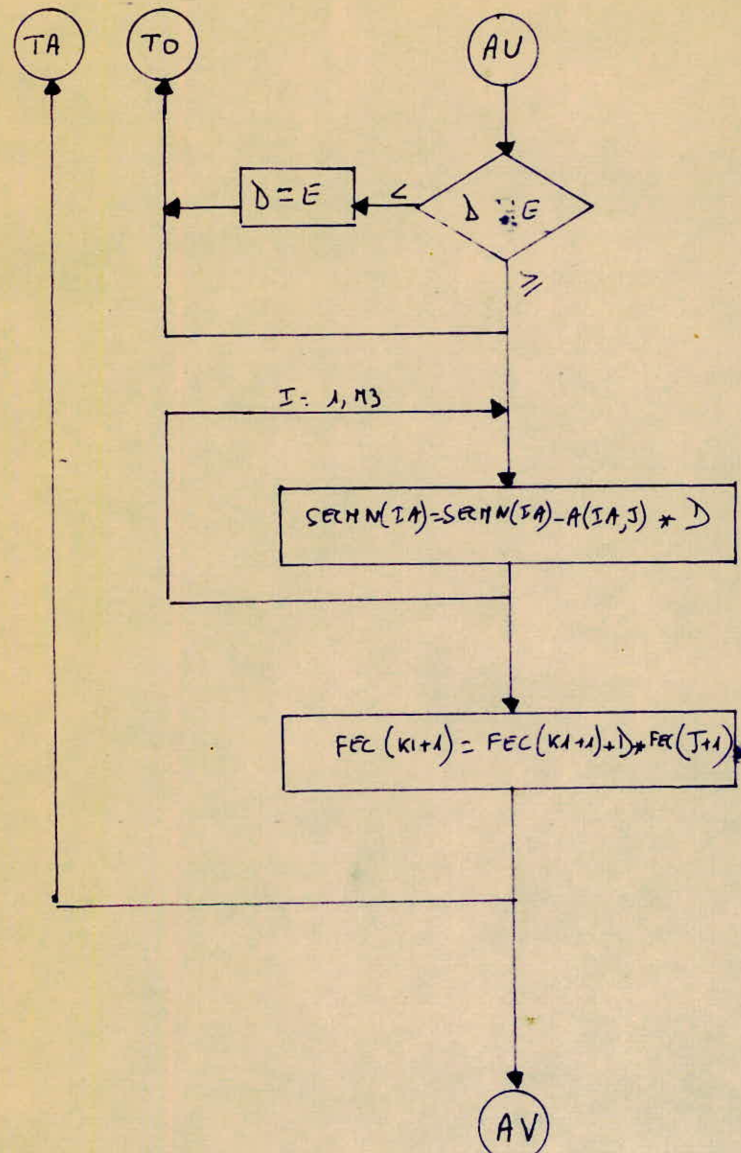
PIVOT MAXIMUN - SP/PIMAX



ITÉRATIONS

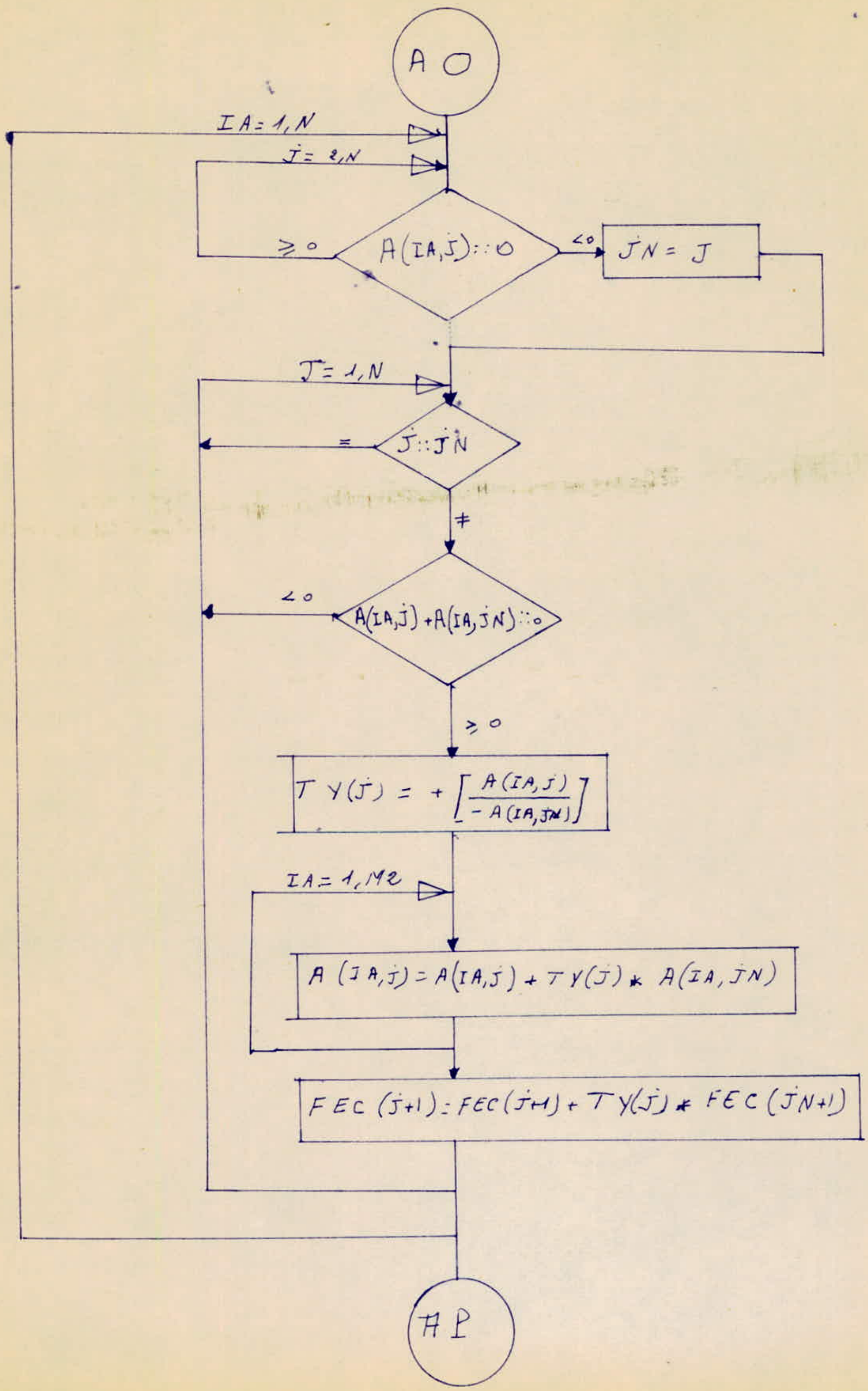

AT
IA = N,1
SECMN(IA)::0
≥
<
IA = KA
J = 1,K1
A(IA,J)::0
≥
<
D = [SECMN(IA)/A(IA,J)]
IA = KA,1
A(IA,J)::0
≥
<
SECMN(IA)::0
≥
<
E = [SECMN(IA)/A(IA,J)]
TA
TO
AU

ITÉRATIONS - SUITE

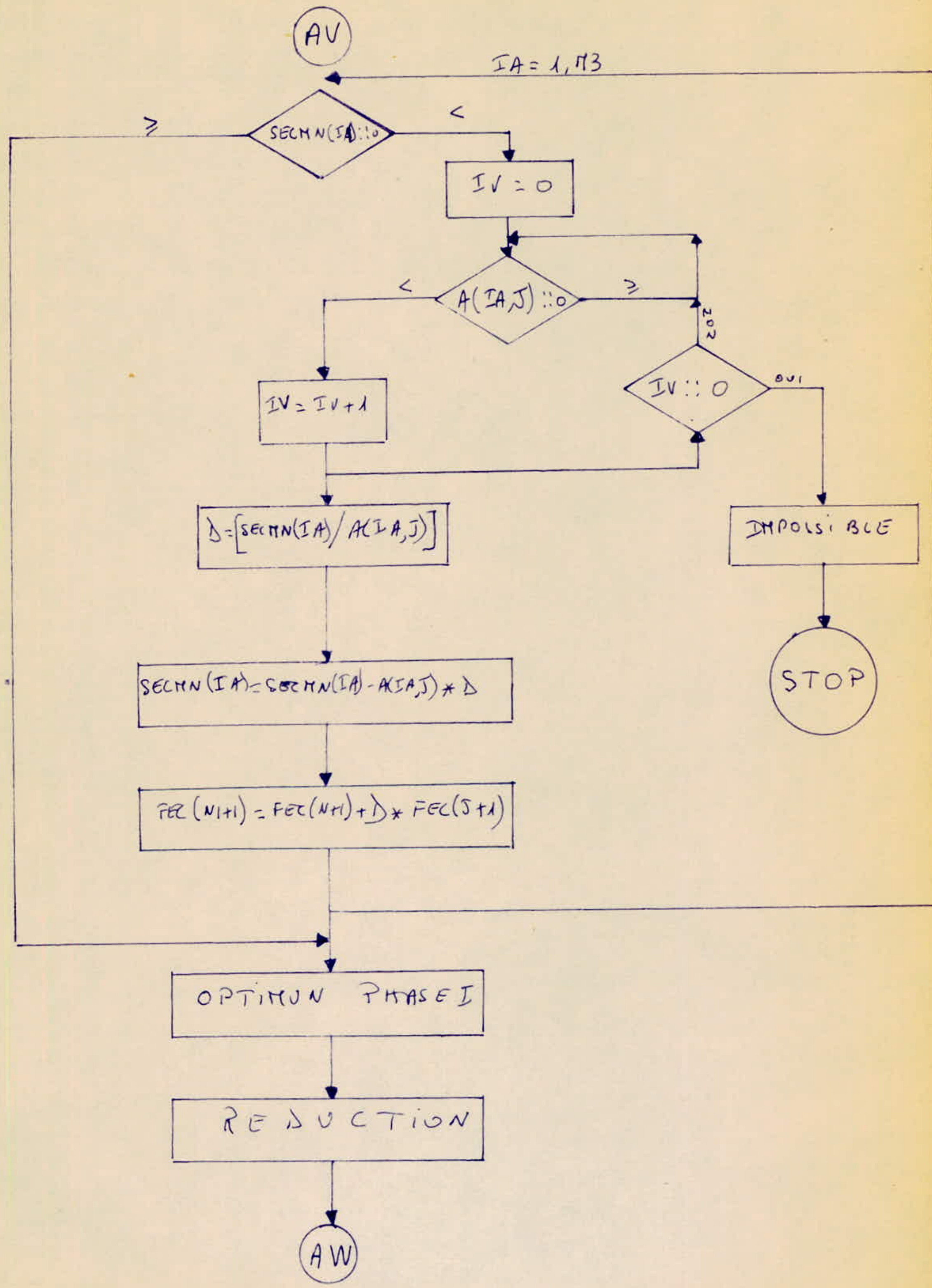


Recherche de la 1re Base Majorante

Programme « NEG »

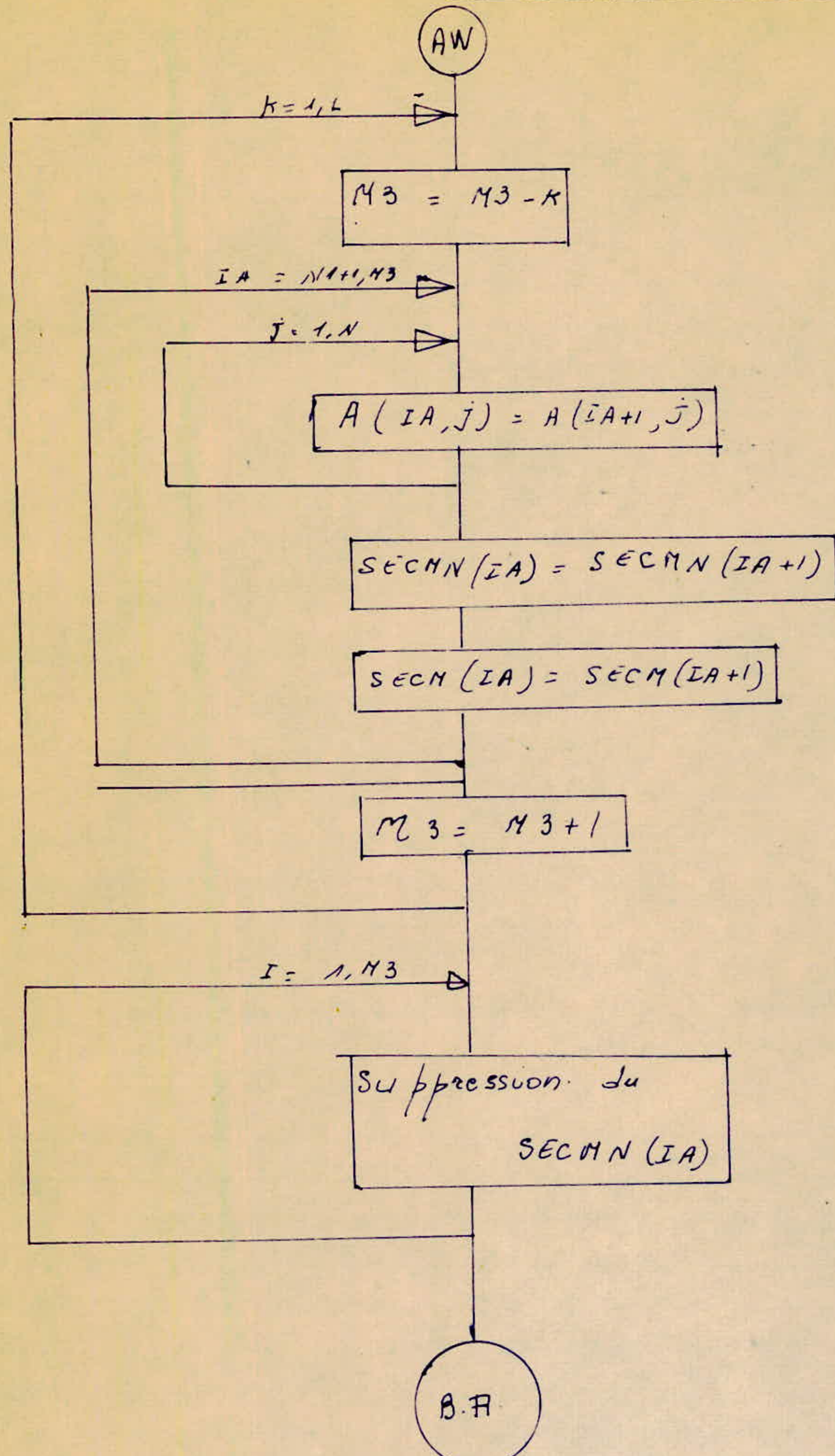


- FINITIONS -


AV
IA = 1, M3
SECMN(IA) :: 0
<
≥
IV = 0
A(IA,J) :: 0
<
≥
NON
IV :: 0
OUI
IV = IV + 1
IMPOSSIBLE
STOP
D = [SECMN(IA) / A(IA,J)]
SECMN(IA) = SECMN(IA) - A(IA,J) * D
FEC(N1+1) = FEC(N1+1) + D * FEC(J+1)
OPTIMUN PHASE I
REDUCTION
AW

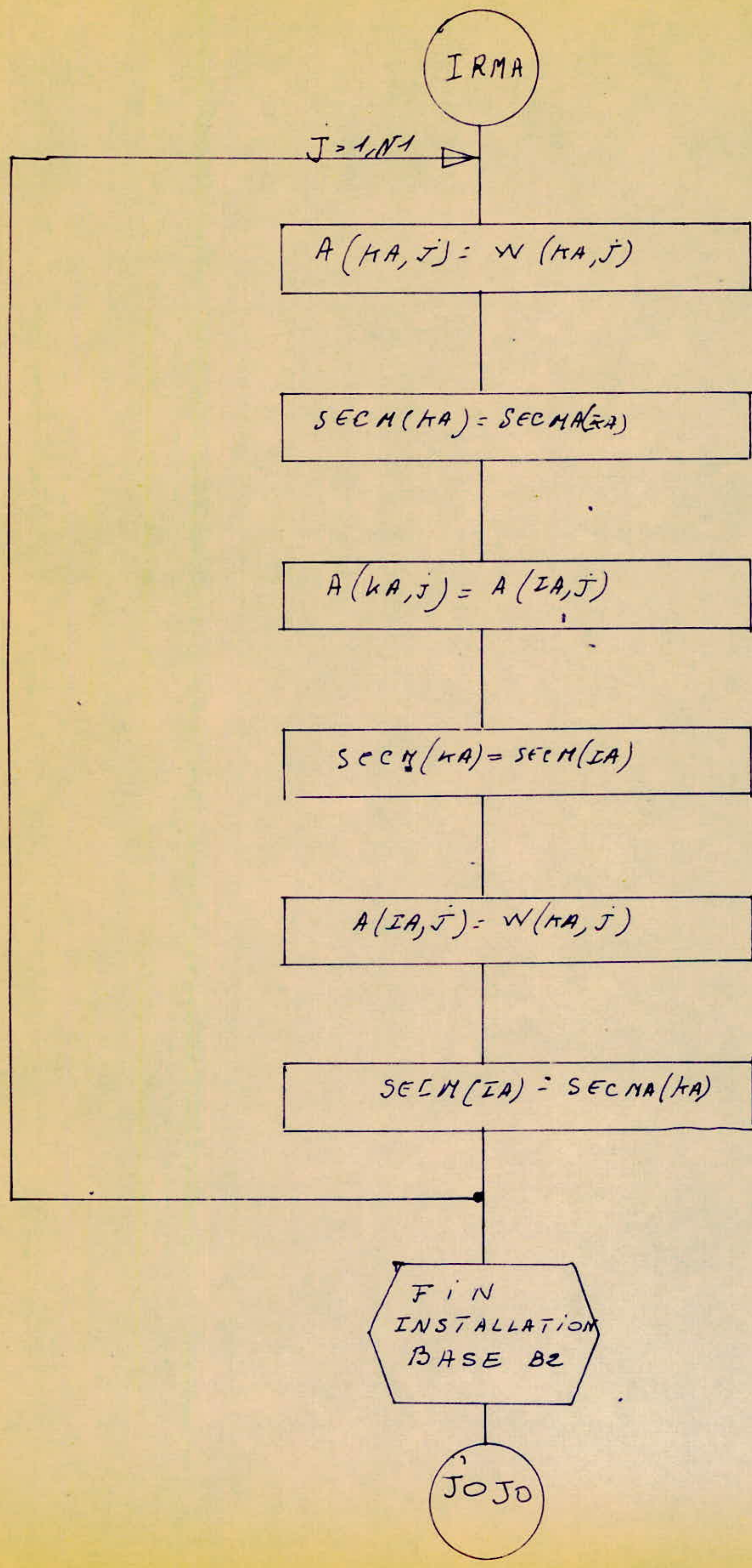
PHASE II

Reduction des Variables Artificielles:



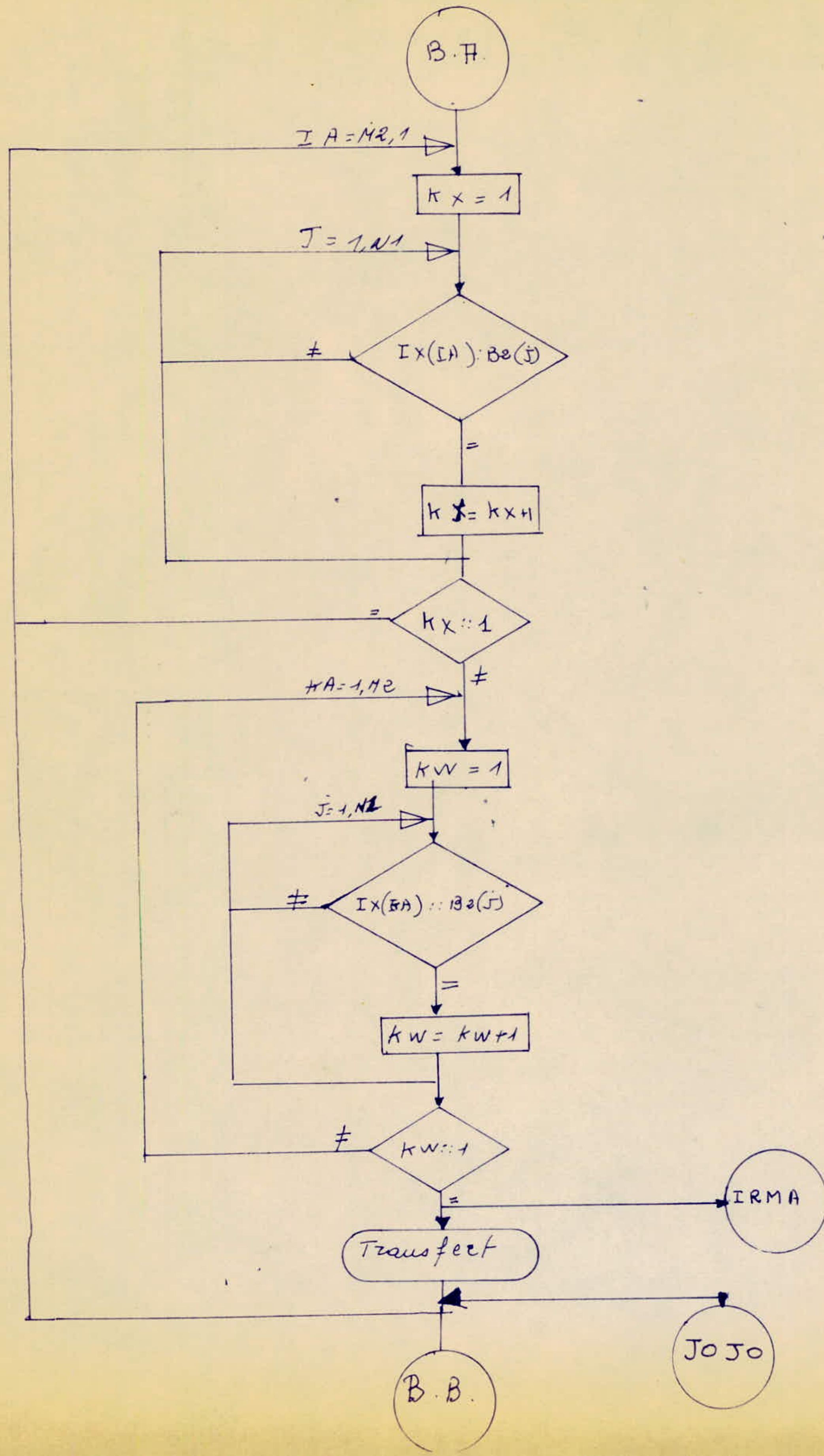
Phase II. (Transfert.)

S2/



Phase II

Installation de la Base optimale B2



ΦASE II

BB

FAIRE DANS le Programme

Recherche de BASES MAJORANTE

SECMN = SECM.

Recherche des 2 Bases Majorantes

Retour au programme Principal.

Pour les K1 Iterations on remplace FEC(N+1) par FEC(1)

A L'OPTIMUM.

REDACTION des RESULTATS.

Ecrire : IX (IA) = SECM (IA)

Ecrire : Fonction Economique Entiere = FEC(1)

STOP

- Commentaire.

- Chargement des données

Il se fera sous forme de trois matrices :

1. Une matrice de dimensions (M, N).
2. Une matrice colonne pour le second membre.
3. Une matrice ligne pour la fonction économique.

$JX(J)$: variables réelles

$IX(I)$: variables d'écart dont la liste est placée dans un tableau

CO -

BO : Base initiale

appelations : - SECM : second membre initial.

- SECMN : second membre auxiliaire.
- FEC : fonction économique

- Test de la J-admissibilité

nous installons un compteur L pour les négatifs se trouvant dans la fonction économique -

- Phase I.

La dimension des lignes sera augmentée de L (nombre de négatifs) - Quant à la dimension des colonnes, elle augmentera d'une unité ceci étant dû à l'introduction du second membre auxiliaire - $N(L)$ est un vecteur qui repèrera le numéro de la colonne possédant un négatif dans la fonction économique.

- Pivotage autour de la matrice unité

Nous rendons le tableau J-admissible.

IS : identificateur de manoeuvre.

- Test de redondance et d'impossibilité par ligne et par colonne - RESIL - RESIC

J1 : compteur des éléments négatifs par ligne.

J2 : " " " " et nuls par ligne.

I1 : " " " " par colonne.

I2 : " " " " et nuls par colonne.

KR : compteur nous permettant de revenir aux redondances lignes au cas où nous trouvions une redondance colonne.

- Continue

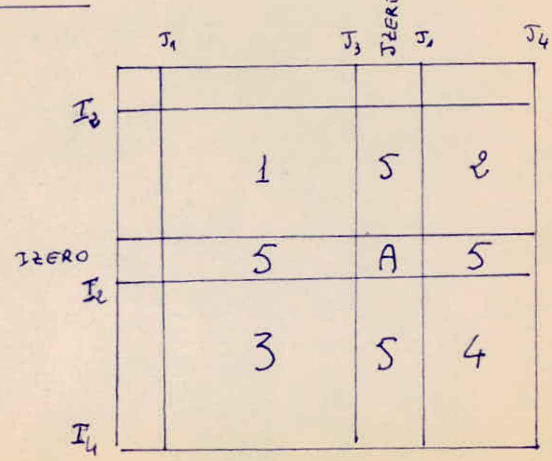
Programme nous permettant d'obtenir l'optimum du programme.

MT : compteur pour les négatifs du second membre

KG : " " " " pour les lignes

- Sous programme pivot - SP/PIVOT.

IM : identificateur de manoeuvre



Test d'optimalité

Pour passer à la phase II il est nécessaire de tester les variables artificielles. Celles-ci doivent être redescendues de la base et être nulles - aussi devons nous installer un compteur IL pour les variables d'écart nulles et un autre IK nous permettant de repérer le numéro des variables artificielles -

Classement

Nous classons les négatifs sur la diagonale en tenant compte des colonnes où existe plusieurs négatifs -

KM - compteur pour les négatifs

Rangement de la partie triangularisée au bas du tableau

Triangularisation -

K : indice permettant de délimiter la triangularisation .

Itérations

Les itérations sont faites de façon à satisfaire les négatifs de la même colonne -

IG : compteur pour les éléments négatifs de la ligne

FINITIONS

IV : compteur pour les éléments négatifs par ligne -

Commentaire : Resolution du Programme en Nombres Entiers

- **Chargement des données :**

- Compteur KJ
teste si les variables appartiennent à $B1$ ou à $B0$ ou bien a aucune des deux.

le chargement se fait en 4 temps :

* chargement en tête du tableau des variables qui appartiennent à $B1$ et qui n'appartiennent pas à $B0$.
* chargement des variables qui appartiennent à $B1$ et qui appartiennent à $B0$.
* chargement des variables qui n'appartiennent pas à $B1$ mais qui appartienne à $B0$.
* chargement du Reste des variables.

On insere pour la phase I L variables artificielles après l'installation du deuxième groupe de variables.

Bilan pour la phase I : Le Tableau a la longueur $1 + N + M + L$ $\quad 1 \leq IA \leq N+M+L$.

pour la Phase II. $1 \leq IA \leq N+M$

. <u>Programme choix de la ligne</u>:

- le Compteur NN(IA) indique le nombre de negatifs dans une ligne i_A.
- le Compteur NNS sera le plus grand nombre de negatifs trouvé jusqu'à present dans une ligne.
- K = IA permet de retenir le N° de la ligne Candidate.
- le Test <NNS-2> permet de savoir si l'on à atteint la Base Majorante

<u>Choix de la Colonne</u>:

- J_0 = colonne inf lexicographique.

3 etapes :

- parmi ttes les colonnes candidates on prend la 1ere comme inf.
- on compare a la colonne suivante.
- s'il y a egalité on fait appel à la lexicographie.

<u>Colonnes Redondantes : et Colonnes non bornées</u>

- K1 : indique le Nbre de variables Restant après suppression des colonnes Redondantes.
- le Programme Comprend deux parties:
  - ★ Detection des colonnes non bornées : ⇒ STOP

* Detection de colonnes Redondantes:
  - Affectation de la valeur zero à la variable.
  - reduction du tableau. (Transfert)

PHASE II

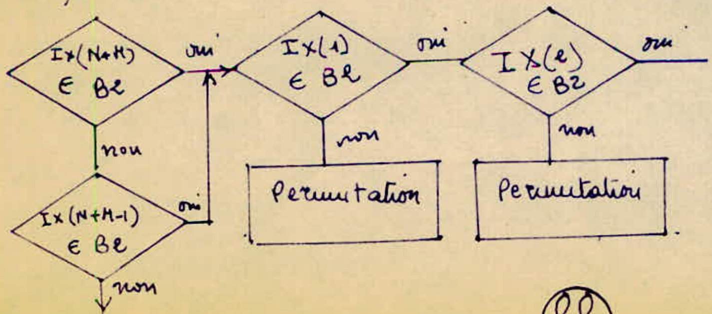
- Installation de la Base optimale B2:

Principe:

- Ayant le Tableau Entier reduit Comprenant toute la liste des variables on fait un test sur la dernière variable du tableau:

<u>si elle appartient a la Base B2:</u>
nous testons la 1ère variable. si celle ci appartient egalement à B2 on la laisse et on teste la 2ème variable si elle n'appartient pas à B2 on fait la permutation entre la dernière et la deuxième puis on teste l'Avant dernier variable et on refait le m travail.

<u>si la dernière variable n'appartient pas à B2:</u>
on passe à l'Avant dernière et on refait le même travail.



Qu'il nous soit permis, à la fin de ce travail de remercier M. AÏT OUYAHIA, professeur à l'Ecole Nationale Polytechnique, de nous avoir proposé ce sujet et de nous avoir guidé d'une manière judicieuse tout au long de cette étude.

Nous remercions également Mme BENNIKOUS pour l'apport de son précieux concours dans la réalisation de ce projet.

M.M. DIB et BERERHI.