

RÉPUBLIQUE DU SÉNÉGAL



ÉCOLE POLYTECHNIQUE DE THIÈS

GC 1224

PROJET DE FIN D'ETUDES

EN VUE DE L'OBTENTION DU DIPLOME D'INGÉNIEUR DE CONCEPTION

TITRE

*GÉOMATIQUE ET SYSTÈME
D'INFORMATION A RÉFÉRENCE
SPATIALE*

DATE : JUIN 1990

AUTEUR : VIGNON AMEDEGNATO
DIRECTEUR : MOUSTAPH M'BAYE
CO-DIRECTEUR :

A ma famille...

REMERCIEMENTS

C'est il me soit permis de saisir cette occasion pour témoigner ma gratitude à tous ceux qui, de quelque manière que ce soit ont contribué à l'élaboration de ce projet de fin d'études.

Les remerciements iront principalement à :

- M^r Aboustaph Abbaye, professeur à l'EPT.*
- M^r Amadou Djim, technicien à l'EPT.*
- M^r Komi Edem Akagbé, Étève-ingénieur à l'EPT.*
- M^r Papa Xuma, Étève-ingénieur à l'EPT.*

SOMMAIRE

Ce document est une étude des logiciels IDRISI, TIMOR et SURFER en vue de l'établissement d'un informatique les rendant compatibles. En cinq chapitres, cette étude portera d'abord sur une définition des notions de géomatique et système d'information à référence spatiale et une description des logiciels, ensuite sur la présentation du programme mis au point et d'une application pratique de système d'information à référence spatiale. Ce travail fera l'objet dans l'avant dernier chapitre de discussions et de recommandations. Enfin, une conclusion générale sera dégagée.

TABLE DES MATIERES

Remerciements	i
Sommaire	ii
CHAPITRE I - INTRODUCTION	1
CHAPITRE II - GEOMATIQUE ET SYSTEME D'INFORMATION A REFERENCE SPATIALE	4
II-1 Système d'information à référence spatiale	5
II-2 Géomatique	6
CHAPITRE III - PRESENTATION DES FICHIERS TIMOR, IDRISI, SURFER ET DU PROGRAMME DE COMPATIBILITE	8
III-1 Les fichiers images	9
III-1-1 Les fichiers images de TIMOR	9
III-1-1-1 Le premier type de format	10
III-1-1-2 Le deuxième type de format	12
III-1-2 Les fichiers images de IDRISI	13
III-2 Les fichiers vectoriels	14
III-2-1 Les fichiers vectoriels IDRISI	14
III-2-1-1 Les fichiers "point"	14
III-2-1-2 Les fichiers "ligne"	15
III-2-1-3 Les fichiers "polygone"	16
III-2-2 Les fichiers vectoriels de SURFER	17
III-3 Les fichiers "attribut" de IDRISI	18

III-4	Le programme de conversion des fichiers vectoriels IDRISI en fichiers vectoriels SURFER et vice versa	18
III-4-1	Le module du menu	19
III-4-2	Le module de création de fichiers vectoriels	19
III-4-3	Le module de stockage des fichiers	20
III-4-4	Le module de rappel de fichier	21
III-4-5	Le module de conversion de fichier	21
CHAPITRE IV APPLICATION PRATIQUE DE CREATION DES I.R.S.		22
IV-1	INTRODUCTION	23
IV-2	MENU.PRG	23
IV-3	ECRAN0.PRG	24
IV-4	ECRAN1.PRG	24
IV-5	ECRAN2.PRG	24
IV-6	ECRAN3.PRG	24
IV-7	ECRAN4.PRG	24
IV-8	Acquisition des données	25
IV-8-1	Traitement des données	25
CHAPITRE V CONCLUSION ET RECOMMANDATIONS		28
BIBLIOGRAPHIE		31
ANNEXES		31
ANNEXE1 - PROGRAMME SIRS (LISTING)		32
ANNEXE2 - PROGRAMMES MENU.PRG		35

CHAPITRE I

INTRODUCTION

INTRODUCTION

On assiste depuis quelques années, à une accélération de l'évolution de la technologie et de la croissance des besoins d'information reliés à la connaissance, l'administration, l'aménagement, l'utilisation et au développement des territoires. Des investissements financiers et en ressources humaines de plus en plus importants sont nécessaires pour se procurer et tenir à jour les informations géographiques nécessaires au bon fonctionnement des multiples institutions publiques et privées de nos pays. Pour bien organiser ce travail, nous avons recours à la géomatique dont le champ d'études englobe certaines activités autour desquelles s'articulent le captage, la manipulation et le traitement, la structuration et le stockage, l'analyse et l'interprétation, la présentation et finalement

la gestion des données et de l'information à référence spatiale.

Ce rapport porte sur la conception d'un programme en langage Basic permettant la compatibilité des logiciels TIMOR, IDRISI, SURFER utilisés en géomatique. Cette compatibilité nous permet de réaliser un système d'information à référence spatiale.

CHAPITRE II

GEOMATIQUE ET SYSTEME
D'INFORMATION A REFERENCE
SPATIALE

II-1/ Système d'information à référence spatiale.

Un système d'information à référence spatiale permet le rattachement d'une information issue d'un système d'information, à une position géographique sur un territoire.

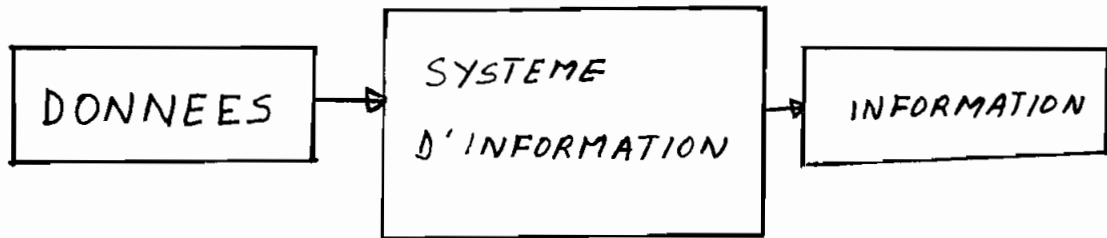
On peut définir un système d'information comme un ensemble organisé globalement comprenant des éléments (données, équipements, procédures, usagers) qui coordonnent pour concourir à un résultat (information). Les quatre principales fonctions sont :

- 1/ La saisie des données;
- 2/ Leur conservation;
- 3/ Leur traitement;
- 4/ Une restitution du résultat fournissant l'information;

Un système d'information est donc créé en vue de fournir l'information nécessaire à celui qui la requiert et ce dans le délai voulu. Il doit faire régulièrement l'objet de contrôles et de

mises à jour.

FIGURE II-1-1



Nous pouvons enfin, à partir de la définition de système d'information dire qu'un système d'information à référence spatiale est un ensemble organisé globalement et comprenant des éléments (données, équipements, procédures, usagers) qui se coordonnent, à partir d'une base géographique commune, pour concourir à un résultat (information).

II-2 Géomatique :

La géomatique est une science qui englobe un ensemble de disciplines reliées à la production des données et de l'information à référence spatiale. Les disciplines sont le droit, les mathématiques, la physique, l'informatique, la topométrie, la gestion foncière, la cartographie, la géodésie, la photogram-

métrie et la télédétection.

En général, le terme géomatique étudie des données sur le territoire plutôt que les données sur les personnes, objets et événements qui, tout en étant localisables géographiquement, ne sont pas des éléments intrinsèques du territoire.

Il faut bien différencier géomatique de système d'information à référence spatiale. Alors que tout système utilisant la localisation géographique peut être qualifié de système d'information à référence spatiale, tous ne font pas appel aux différentes disciplines de la géomatique au même degré. La géomatique intègre, donc, les disciplines spécifiques permettant d'obtenir les données de base sur le territoire pour un système d'information à référence spatiale.

CHAPITRE III

PRESENTATION DES FICHIERS
TIMOR, IDRISI, SURFER ET DU
PROGRAMME DE COMPATIBILITE

III-1 Les fichiers images.

III-1-1 Les fichiers images de TIMOR.

Le logiciel TIMOR (Traitement d'Images par Micro-ordinateur Orstom) regroupe trois logiciels : TIMORSS, STEPPE, TROPICA.

Ce groupe de logiciels permet de faire des traitements d'images, des analyses statistiques sur des parcelles d'entraînement et de produire des images couleurs sur micro-ordinateur IBM Ac.

Les fichiers images TIMOR se présentent sous forme de tableau de plusieurs pixels disposés en lignes et colonnes. La première ligne comporte des renseignements sur le fichier. La suite est constituée de 256 lignes et 256 ou 128 colonnes. A l'intersection de chaque ligne et colonne se situe une case appelée pixel. Chaque pixel contient un nombre qui représente la radiance spectrale de la portion de sol qu'il décrit.

Les fichiers images sont sauvegardés sur

des disquettes de 5 pouces 1/4 double faces, double densités ou sur disque dur d'un ordinateur. Ils sont enregistrés suivant deux types de format. Le premier type concerne les images de 256 colonnes et 256 lignes comportant quatre canaux. Le deuxième type a trait aux images de 128 colonnes sur 256 lignes ayant 7 canaux.

III-1-1-1 Le premier type de format.

Dans ce cas, l'image de 256×256 pixels comporte quatre canaux. Ils correspondent aux canaux 4, 5, 6, 7 des scènes Landsat, aux trois canaux d'une scène SPOT avec, pour des raisons de commodité, un redoublement du troisième, ou bien à quatre canaux d'une scène T.M. 256×256 . Un premier enregistrement correspond aux indications relatives à la portion de scène consignée sur la disquette : nature des données, origine, lignes, colonnes. Le format est le suivant : A51, A28, 4I4.

L'enregistrement suivant de format $256A_2$ contient sous forme d'une succession de caractères ASCII, les données du canal 4 (ou

x51) de la première ligne, suivies par les données du canal 5 (ou x52) de cette même ligne, puis celles du canal 6 (ou x53), enfin par celles du canal 7 (ou x53). Les quatre canaux d'une ligne utilisent 1024 octets, soit 4×256 octets. Le canal 4 (ou x51) de la deuxième ligne est placé sur l'enregistrement 5, le canal 5 de cette ligne sur l'enregistrement 6 et ainsi de suite. Sous forme de tableau on a :

PREMIER ENREGISTREMENT.

FORMAT	A51	A28	4 I 4
VARIABLES	TEX1	TEX3	L ₁ , L ₂ , C ₁ , C ₂

TEX1 : date, trace et rang de la zone.

TEX3 : origine des données.

L₁, L₂, C₁, C₂ : lignes et colonnes de la scène.

ENREGISTREMENTS SUIVANTS.

FORMAT	256 A1	N° d'enregistrement
C4 (ou x51) de la ligne 1		2
C5 (ou x52) de la ligne 1		3
C6 (ou x53) de la ligne 1		4
C7 (ou x53) de la ligne 1		5
C4 (ou x51) de la ligne 2		6
C5 (ou x52) de la ligne 2		7
C6 (ou x53) de la ligne 2		8
⋮		⋮
C6 (ou x53) de la ligne 256		1024
C7 (ou x53) de la ligne 256		1025

L'ensemble du fichier occupe 264 289 octets dont 262 144 par les enrégistrement des quatre canaux de l'image.

Les données multiplexées (sur plusieurs canaux ici 4) peuvent être remplacées pour des raisons de rapidité dans le travail, par des fichiers mono-canal. Le format est alors le suivant:

Le premier enrégistrement a la même structure que dans le cas précédent (A51, A28, 4I4). Il est suivi de 256 enrégistrement de format 256A1 où sont consignées les lignes successives de l'image. Le fichier mono-canal occupe 66 145 octets dont 65 536 par les enrégistrement du canal.

III-1-1-2 Le deuxième type de format.

Ce format est utilisé pour les données Thematic Mapper. La disquette 5 pouces 1/4 contient les sept canaux d'une scène de 128 colonnes sur 256 lignes, occupant 233 057 octets environ dont 229 376 utilisés pour les enrégistrement des sept canaux. Le premier enrégistrement a la même structure que dans le cas précédent, à savoir le format A51, A28, 4I4.

Les enrégistrement suivants (de 2 à 1793), de format 128A1, suivent une même logique, à cela près que chaque ligne compte sept enrégistrement successifs correspondant aux canaux 1, 2, 3, 4, 5, 6 et 7 de Thematic Mapper. Ainsi, par exemple, la ligne 1 occupera les enrégistrement 2 à 8 inclus.

III - 1-2 Les fichiers images de IDRISI.

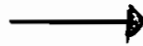
Les fichiers images IDRISI se présentent sous forme d'un tableau de plusieurs pixels disposés en lignes et colonnes. Le nombre de lignes ou colonnes est très variable suivant les images. A l'intersection de chaque ligne et colonne se situe un élément appelé aussi pixel. Chaque élément contient un nombre qui caractérise un phénomène donné sur le terrain. Le nombre peut être soit un entier de -32768 à +32767, soit un byte (entier positive de 0 à 255) ou un réel. Les nombres des éléments bien que disposés sous forme de tableau sur l'image sont enrégistrés en une seule colonne.

Le logiciel IDRISI qui est un système d'analyse géographique basé sur l'utilisa-

tion de pixel pour la représentation des phénomènes, utilise aussi des fichiers vectoriels et "attribut".

FICHIER IMAGE IDRISI (EXEMPLE)

$x_{1,1}$	$x_{1,2}$	$x_{1,3}$
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$



- $x_{1,1}$
- $x_{1,2}$
- $x_{1,3}$
- $x_{2,1}$
- $x_{2,2}$
- $x_{2,3}$
- $x_{3,1}$
- $x_{3,2}$
- $x_{3,3}$

Sur l'image .

Enregistrement des données du fichier image.

III-2 Les fichiers vectoriels.

III-2-1 Les fichiers vectoriels de IDRISI.

Les fichiers vectoriels de IDRISI sont de trois types: les fichiers "point", "ligne" et "polygone".

III-2-1-1 Les fichiers "point".

Structure.

- a_1 1
- x_1 y_1
- a_2 1
- x_2 y_2
- a_3 1
- x_3 y_3
- ...
- a_m 1
- x_m y_m
- 0 0

Les a_i sont des nombres réels appelés identificateurs. Ils représentent la valeur assignées aux points de coordonnées x_i, y_i . La valeur de a_i représente la valeur du phénomène étudié au point (x_i, y_i) . Le nombre 1 indique que c'est le seul point de coordonnées x_i, y_i qui suit a_i dans le fichier qui est représenté par a_i . Les deux zéro indiquent la fin du fichier.

III-2-1-2 Les fichiers "ligne":
structure.

a_1	b_1
x_{1-1}	y_{1-1}
x_{1-2}	y_{1-2}
x_{1-3}	y_{1-3}
⋮	⋮
x_{1-b_1}	y_{1-b_1}
a_2	b_2
x_{2-1}	y_{2-1}
⋮	⋮
x_{2-b_2}	y_{2-b_2}
a_3	b_3
⋮	⋮
a_i	b_i
x_{i-1}	y_{i-1}
⋮	⋮
a_n	b_n
x_{n-1}	y_{n-1}
x_{n-b_n}	y_{n-b_n}
0	0

Les a_i sont des nombres réels et représentent l'identificateur d'un point de coordonnées x_{i-j}, y_{i-j} . Les b_i sont des nombres entiers positifs et représentent le nombre de points x_{i-j}, y_{i-j} qui constituent la ligne i et comme pour les fichiers "point", deux zéro marquent la fin du fichier.

III-2-1-3 Les fichiers "polygone".

Ils ont la même structure que les fichiers "ligne" à la seule différence que le premier x_{i-1}, y_{i-1} est le même que le dernier x_{i-b_i}, y_{i-b_i} . Dans ce type, l'identificateur a_i représente la valeur à assigner à tous les points à l'intérieur du polygone décrit par les points du fichier. Exemple: soit le fichier polygone

a_i	b_i
x_{i-1}	y_{i-1}
x_{i-2}	y_{i-2}
⋮	⋮
x_{i-b_i}	y_{i-b_i}
x_{i-1}	y_{i-1}
0	0

... tous les points situés à l'intérieur du polygone décrit par les b_i points, auront comme

valeur a_i .

III-2-2 Les fichiers vectoriels de SURFER:

Structure:

x_1	y_1	z_1
x_2	y_2	z_2
...
x_i	y_i	z_i
...
x_m	y_m	z_m

Le logiciel SURFER permet une représentation sous forme de courbes en 2 ou 3 dimensions de phénomènes telles l'altitude d'un point (courbe de niveau), la pluviométrie.

Les fichiers vectoriels de SURFER ont une structure très simple. Ils sont constitués d'une série de coordonnées x_i, y_i, z_i comme le montre leur structure. Les x_i, y_i sont des nombres réels représentant les coordonnées d'un point A_i dans un système de coordonnées spécifique. Le z_i peut être un nombre réel ou un symbole et représente et représente un phénomène situé au point A_i .

III-3 Les fichiers "attribut" de IDRISI.

Ce type de fichier permet de lier une caractéristique à un nom donné.

Structure.

<u>Premier fichier</u>		<u>Deuxième fichier</u>	
1	x_1	1	y_1
2	x_2	2	y_2
3	x_3	3	y_3
⋮	⋮	⋮	⋮
i	x_i	i	y_i
⋮	⋮	⋮	⋮
n	x_n	n	y_n

Si on considère les x_i comme les caractéristiques, x_i sera la caractéristique du nom y_i . Par exemple si les x_i représentent des densités de population et les y_i des noms de localité, x_i sera la densité de populations de la localité y_i .

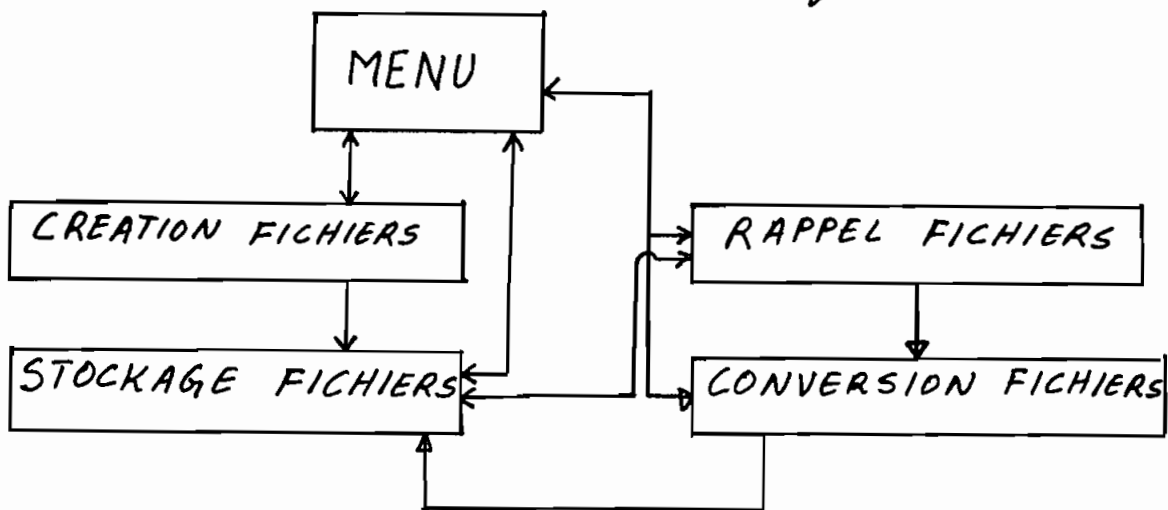
III-4 Le programme de conversion des fichiers vectoriels IDRISI en fichiers vectoriels SURFER et vice versa:

Le programme nommé SIRS et exécutable sous ce nom, est écrit en langage BASIC. Il permet de convertir les fichiers vectoriels

IDRISI en fichiers vectoriels SURFER et vice versa.

Le programme est constitué de cinq modules :

- Le module du menu.
- Le module de création de fichiers vectoriels (IDRISI ou SURFER).
- Le module de stockage de fichiers vectoriels.
- Le module de rappel de fichiers d'une unité donnée de l'ordinateur.
- Le module de conversion des fichiers.



III-4-2/ Le module du menu :

Le menu permet l'affichage à l'écran et le choix d'une des quatre autres options ou d'une sortie vers le système d'exploitation (DOS).

III-4-2 Le module de création de fichiers vectoriels : Il vous donne le moyen de créer

des fichiers vectoriels IDRISI ou SURFER. Pour les fichiers vectoriels IDRISI, les nombres sont entrés suivant deux colonnes. Ces nombres sont des réels, à part le nombre indiquant la quantité de points contenus par une entité. La fin du fichier est marquéé par deux zéro dont l'apparition arrête automatiquement l'entrée des données et entraîne le retour au menu. Les fichiers vectoriels SURFER sont entrés suivant trois colonnes de données. Les données doivent être des réels. Le programme vous permet de spécifier le nombre de points à entrer et arrête automatiquement l'écriture des points dès que ce nombre est atteint.

III-4-3 Le module de stockage des fichiers:

Les fichiers créés à partir du programme doivent être obligatoirement stocker tout juste après la fin des entrées des données sinon ils seront perdus. Le stockage de fichiers concerne donc spécialement les fichiers créés à partir du programme. Mais on peut stocker un fichier existant,

sous un autre nom après l'avoir appelé?

III-4-4 Le module de rappel de fichier:

Ce module est incorporé au module de conversion de fichier. Un fichier rappelé est donc converti par la suite. Ce module permet de rappeler un fichier d'une unité donnée de l'ordinateur.

III-4-5 Le module de conversion de fichier:

Il nous permet de transformer les fichiers du type deux colonnes en fichiers du type trois colonnes (SURFER). Dans la conversion, l'identificateur d'un point est transformé en la coordonnée Z du point dans le cas d'une conversion IDRISI à SURFER et l'inverse se produit dans le cas d'une transformation de fichiers vectoriels SURFER en fichiers vectoriels IDRISI.

CHAPITRE IV

APPLICATION PRATIQUE
DE CREATION DE S.I.R.S.

IV-1 INTRODUCTION:

Cette partie concerne l'établissement d'une banque de données à partir des résultats obtenus en utilisant les logiciels IDRISI et SURFER pour le traitement des fichiers vectoriels de la région administrative de Ehiès.

La banque de données est gérée par un programme de gestion de banque de données exécutable en option DBASEIII.

Pour pouvoir exécuter le programme, on entre en mode DBASEIII puis on tape la touche F2. On entre après le nom du programme (MENU) suivi de l'instruction DO. Ce programme est constitué de six modules: MENU.PRG, ECRANO.PRG, ECRAN1.PRG, ECRAN2.PRG, ECRAN3.PRG et ECRAN4.PRG.

IV-2 MENU.PRG:

A partir de MENU.PRG, tous les autres modules peuvent être exécutés. Il constitue la porte d'entrée pour l'écriture et la lecture de donnée dans la banque.

IV-3 ECRAN 0. PRG:

C'est avec ce module qu'on enregistre les données d'occupation et la pente du sol pour une position géographique donnée.

IV-4 ECRAN 1. PRG:

Ce module nous permet de rechercher les données d'occupation et la pente du sol d'un élément (pixel) en donnant sa position géographique (latitude et longitude). Il nous affiche à l'écran les résultats.

IV-5 ECRAN 2. PRG:

Il nous permet de rechercher les données d'occupation et la pente du sol d'un élément à partir du nom de la localité située sur l'élément.

IV-6 ECRAN 3. PRG:

Ce module nous affiche à l'écran de l'ordinateur l'état de la route reliant deux localités en faisant enregistrer les noms de ces deux localités.

IV-7 ECRAN 4. PRG:

Il permet l'enregistrement des données

sur les routes reliant les localités.

IV-8 Acquisition des données:

Pour notre application, nous avons choisi la région administrative de THIES. Les données utilisées sont captées à partir de cartes géographiques à l'échelle 1/200 000. Les types de données utilisées sont l'occupation et l'utilisation du sol et les courbes de niveau.

IV-8-1 Traitement des données:

A partir des données d'occupation et d'utilisation du sol, nous avons créé des fichiers vectoriels "point" IDRISI du nom de PXYZ1.VEC. Du fichier PXYZ1.VEC, nous avons créé un fichier PXYZ1.DVC avec le module DOCUMENT V de IDRISI. Le fichier PXYZ1.VEC a pour fichier document le fichier PXYZ1.DVC. Ce fichier document permet aux autres modules de IDRISI de prendre les renseignements nécessaires à leurs traitements sur le fichier PXYZ1.VEC. Avec le module INITIAL de IDRISI, une image vierge est créée et le module POINTRAS transforme

en fichier image PXYZ1.IMG le fichier vectoriel PXYZ1.VEC par une transposition sur l'image vierge créée par INITIAL.

Le fichier image PXYZ1.IMG s'accompagne d'un fichier PXYZ1.DOC qui donne les renseignements nécessaires aux traitements sur l'image.

On crée aussi un fichier PXYZ1.VAL qui nous permet d'assigner par le module ASSIGN des valeurs et des noms de localité aux différents éléments ou pixels de l'image PXYZ1.IMG.

Le module VIEW nous permet de visualiser la valeur assignée à chaque pixel à partir de l'image PXYZ1.IMG. C'est ces valeurs qui vont constituer les données pour la banque de données.

Les données géodésiques (x, y, z) où x, y représentent les coordonnées d'un point dans un repère cartographique et z son altitude en mètre (m) forme un fichier du type SURFER (fichier PXYZ2.DAT). Le fichier PXYZ2.DAT est converti en fichier PXYZ2.VEC de IDRISI. Avec successi-

vement les modules DOCUMENT, INITIAL, POINTRAS, nous avons obtenu un fichier image IDRISI PXYZ2.IMG, avec son fichier document, PXYZ2.DOC. Le module SURFACE de IDRISI nous permet d'avoir la pente du sol pour chaque pixel. Les pentes en pourcentage sont visualisées avec le module VIEW.

Les pentes, les altitudes en mètres des éléments ainsi que les occupations et utilisations du sol de chaque pixel nous permettent de constituer une base de données liée à la position géographique des éléments. C'est cette base de données que le programme de gestion en DBASE^{III} utilise. L'ensemble base de données et programme de gestion de cette base nous donne un système d'information à référence spatiale pour la région de THIES.

CHAPITRE V

CONCLUSION ET
RECOMMANDATIONS

Cette étude nous a permis de mettre au point un programme en langage BASIC nommé SIRS, programme qui rend compatibles les logiciels IORISI et SURFER. Le programme de compatibilité entre les logiciels TIMOR et IORISI n'a pas eu les résultats attendus à cause d'un manque d'images satellitaires utilisables par TIMOR.

Nous avons aussi mis au point un système d'information à référence spatiale, constitué par une base de données géographiques de Thiès et un programme en DBASE III pour la gestion de cette base. Ce système d'information à référence spatiale nous informe sur l'utilisation et la pente moyenne du sol sur des éléments de 4 km². Le nombre limité d'informations et l'imprécision relative de ces informations au niveau de la région sont des inconvénients pour l'utilisation de ce S.I.R.S. dans l'établissement d'un plan d'aménagement de la région. Nous recommandons donc

que ce projet soit continué pour une plus grande diversification des informations du système et une augmentation de la précision des pixels utilisés.

BIBLIOGRAPHIE

- IDRISI VERSION 3.0
 J. RONALD EASTMAN
 CLARK UNIVERSITY
 GRADUATE SCHOOL OF GEOGRAPHY

- TOPOMETRIE GENERALE
 ERNEST P. LAUZON ET ROGER DUQUETTE

- DBASE IV *et III*
 BERNAR FRALA

- TIMOR
 LOGICIEL DE TELEFECTION POUR MICRO-ORDINATEUR
 COMPATIBLE PC/CAT
 JEAN-FRANCOIS PARROT
 HERUE CHEVILLOTTE

- TRAITEMENT D'IMAGE PAR CLASSIFICATION
 C. MORZIER, P. MEYLAN, A. MUSY.

- *Systèmes d'information à référence spatiale*
 Recueil de lectures
 FACULTÉ DE FORESTERIE ET DE GÉODÉSIE
 UNIVERSITÉ LAVAL

PROGRAMME DE CONVERSION FICHIERS

```

10 KEY OFF
11 ON KEY(1) GOSUB 9996
12 KEY(1) ON
13 ON KEY(2) GOSUB 9998
14 KEY(2) ON
20 CLS:DIM A(255),B(255),C(255),H(255),G(255)
30 ON ERROR GOTO 990
40 LOCATE 5,5,1:PRINT "CREER UN FICHER----->"
50 LOCATE 7,5,1:PRINT "CONVERSION D'UN FICHER----->"
60 LOCATE 9,5,1:PRINT "STOCKER UN FICHER----->"
80 LOCATE 11,5,1:PRINT "ALLER EN DOS----->"
90 CDS=5
100 LOCATE CDS,39,1
110 ON ERROR GOTO 990
120 GOSUB 320
130 VD$=INPUT$(1)
140 IF VD$<>"2" AND VD$<>"8" AND VD$<>CHR$(13) THEN BEEP:GOTO 100
150 IF VD$="8" AND CDS=5 THEN CDS=11:GOTO 100
160 IF VD$="2" AND CDS<>11 THEN CDS=CDS+2:GOTO 100
170 IF VD$="8" AND CDS<>5 THEN CDS=CDS-2:GOTO 100
180 IF VD$="2" AND CDS=11 THEN CDS=5:GOTO 100
190 IF CDS=5 THEN GOSUB 1020
200 IF CDS=7 THEN GOSUB 400
210 IF CDS=9 THEN GOSUB 750
220 IF CDS=11 THEN GOSUB 9991
230 'IF CDS=13 THEN GOSUB 4000
240 GOTO 100
250 END
260 LOCATE 5,5,1:PRINT "CREER UN FICHER----->"
270 LOCATE 7,5,1:PRINT "CONVERSION D'UN FICHER----->"
280 LOCATE 9,5,1:PRINT "STOCKER UN FICHER ----->"
285 LOCATE 11,5,1:PRINT "ALLER EN DOS----->"
310 RETURN
320 GOSUB 260
330 IF CDS=5 THEN COLOR 0,7:LOCATE 5,5,1:PRINT "CREER UN FICHER-----
->":COLOR 7,0
340 IF CDS=7 THEN COLOR 0,7:LOCATE 7,5,1:PRINT "CONVERSION D'UN FICHER-----
->":COLOR 7,0
350 IF CDS=9 THEN COLOR 0,7:LOCATE 9,5,1:PRINT "STOCKER UN FICHER -----
->":COLOR 7,0
351 IF CDS=11 THEN COLOR 0,7:LOCATE 11,5,1:PRINT "ALLER EN DOS-----
-->":COLOR 7,0
380 LOCATE CDS,39,1
390 RETURN
400 LOCATE 19,1,1:PRINT "1 -pour un fichier VECTORIEL IDRISI convertir"
410 LOCATE 20,1,1:PRINT "2 -pour un fichier .dat de SUFER convertir  "
420 LOCATE 22,12,1:PRINT "VOTRE CHOIX ?" :LOCATE 22,27,1:INPUT "",CF$
430 IF CF$="1" THEN GOSUB 460
440 IF CF$="2" THEN GOSUB 640
450 IF CF$<>"2" AND CF$<>"1" THEN BEEP:LOCATE 22,27,1:PRINT "
      ":GOTO 420
460 LOCATE 19,1,1:PRINT "
470 LOCATE 20,1,1:PRINT "
480 LOCATE 22,12,1:PRINT "
490 LOCATE 19,1,1:PRINT "NOM DU FICHER A CONVERTIR (Unit:NOM.EXT) "
500 LOCATE 20,20,1:INPUT "",NOM$
510 A(1)=12 :B(1)=12 :J=1 :K=1
520 OPEN "i",#1,NOM$
530 I=1
540 'IF EOF(1) THEN CLOSE #1
550 INPUT #1,A(I),B(I)
560 IF A(I)<>0 OR B(I)<>0 THEN I=I+1: GOTO 540 'J=I :K=K+1: G(K)=B(I):H(K)=A(
570 'wend
580 'P=1:Q=1
590 CLOSE #1
600 GOSUB 9000 'P=1:Q=1

```

```

610 'FOR I=Q+1 TO G(P)+1 :C(I)=H(P):NEXT
620 'IF P<K THEN P=P+1:Q=Q+G(P)+1:GOTO 610
630 RETURN
640 LOCATE 19,1,1:PRINT "
650 LOCATE 20,1,1:PRINT "
660 LOCATE 22,12,1:PRINT "
670 LOCATE 19,1,1:PRINT "NOM DU FICHER A CONVERTIR (Unit:NOM.EXT) "
680 LOCATE 20,20,1:I=1:INPUT "",NOM$
690 OPEN "i",#2,NOM$
700 IF EOF(2) THEN END
710 INPUT #2,A(I),B(I),C(I)
720 I=I+1:GOTO 700
730 FOR J=1 TO I-1:P(J)=A(J) :Q(J)=B(J):R(J)=C(J):NEXT
740 FOR J=1 TO 2*(I-1):IF J MOD(2)=1 THEN A(J)=C((J-1)/2+1):B(J)=1 ELSE A(J)=
5*J):B(J)=Q(.5*J):NEXT
745 RETURN
750 LOCATE 19,1,1:PRINT "1 -pour un fichier VECTORIEL IDRISI STOCKER"
760 LOCATE 20,1,1:PRINT "2 -pour un fichier .dat de SUFER STOCKER "
770 LOCATE 22,12,1:PRINT "VOTRE CHOIX ?" :LOCATE 22,27,1:INPUT "",CF$
780 IF CF$="1" THEN GOTO 810
790 IF CF$="2" THEN GOTO 900
800 IF CF$<>"2" AND CF$<>"1" THEN BEEP:LOCATE 22,27,1:PRINT "
":GOTO 770
810 LOCATE 19,2,1:I=1:PRINT "
"
820 LOCATE 20,1,1:PRINT "
830 LOCATE 22,12,1:PRINT "
840 LOCATE 19,1,1:PRINT "NOM DU FICHER A STOCKER (Unit:NOM.EXT) "
850 LOCATE 20,22,1:INPUT "",NOM$
860 OPEN "o",#1,NOM$
870 WHILE A(I)<>0 OR B(I)<>0
880 PRINT #1,A(I),B(I)
882 I=I+1
883 IF A(I)=0 AND B(I)=0 THEN PRINT #1,A(I),B(I)
885 WEND
890 CLOSE #1 :RETURN
900 LOCATE 19,2,1:PRINT "
910 LOCATE 20,2,1:PRINT "
920 LOCATE 22,12,1:PRINT "
930 LOCATE 19,1,1:PRINT "NOM DU FICHER A STOCKER (Unit:NOM.EXT) "
940 LOCATE 20,22,1:I=1:INPUT "",NOM$
950 OPEN "o",#2,NOM$
960 WHILE A(I)<>0 OR B(I)<>0
970 PRINT #2,A(I),B(I),C(I)
975 I=I+1
976 WEND
980 CLOSE #2 :RETURN
990 SOUND 450,1:IF ERR=53 THEN COLOR 0,7:LOCATE 23,2,1:PRINT "fichier introuv
e :appuyez une touche pour continuer":LOCATE 23,58,1:VR$=INPUT$(1):COLOR 7,0
1000 LOCATE 23,2,1:PRINT "
"
1010 RESUME 100
1020 LOCATE 19,1,1:PRINT "1 -pour un fichier VECTORIEL IDRISI crer "
1030 LOCATE 20,1,1:PRINT "2 -pour un fichier .dat de SUFER crer "
1040 GOSUB 1380
1050 LOCATE 22,12,1:PRINT "VOTRE CHOIX ?" :LOCATE 22,27,1:INPUT "",CF$
1060 IF CF$="1" THEN GOTO 1080
1070 IF CF$="2" THEN GOTO 1250
1080 LOCATE 4,48,1:PRINT "ENTRER ICI LE FICHER "
1090 LOCATE 5,50,1:PRINT " VECTORIEL IDRISI "
1100 LOCATE 19,1,1:PRINT "
1110 LOCATE 20,1,1:PRINT "
1120 LOCATE 22,12,1:PRINT "
1130 GOSUB 1230
1140 LOCATE 6,50,1:COLOR 0,7:PRINT " X Y N½ " :COLOR 7,0
1150 G=0: A(G)=111:B(G)=111:J=1

```

```

1160 WHILE A(G)<>0 OR B(G)<>0
1170 G=G+1 :LOCATE J+7,72,1:COLOR 0,7:PRINT G :COLOR 7,0
1180 LOCATE J+7,52,1:INPUT "",A(G):LOCATE J+7,67,1:INPUT "",B(G)
1190 DXS=CSRLIN:J=J+1
1200 IF DXS=23 THEN J=1 ELSE GOTO 1220
1210 FOR F=7 TO 23:LOCATE F,48,1:PRINT " " :NEXT
1220 WEND
1230 FOR I=3 TO 23 :COLOR 0,7:LOCATE I,46,1:PRINT " " : COLOR 7,0:NEXT
1240 RETURN
1250 LOCATE 6,50,1:COLOR 0,7:PRINT " X Y Z N½ " :COLOR 7,
1251 LOCATE 4,48,1:PRINT "ENTRER ICI LE FICHIER "
1252 LOCATE 5,50,1:PRINT " VECTORIEL SUFER "
1260 LOCATE 19,1,1:PRINT "
1270 LOCATE 20,1,1:PRINT "
1280 LOCATE 22,12,1:PRINT " "
1290 GOSUB 1230
1300 J=1:LOCATE 22,2,1:INPUT "Nombre de points entrer ?",NBRE
1309 LOCATE 22,2,1:PRINT " "
1310 FOR G=1 TO NBRE
1320 LOCATE J+7,73,1:COLOR 0,7:PRINT G :COLOR 7,0
1330 LOCATE J+7,52,1:INPUT "",A(G):LOCATE J+7,60,1:INPUT "",B(G):LOCATE J+7,6
:INPUT "",C(G):J=J+1 :DXS=CSRLIN
1340 IF DXS=23 THEN J=1 ELSE GOTO 1360
1350 FOR F=7 TO 23:LOCATE F,48,1:PRINT " " :NEX
1355 'IF G=NBRE THEN SOUND 450,2
1360 NEXT
1370 RETURN
1380 FOR P=1 TO 23:LOCATE P,47,1:PRINT " " :NEX
1390 RETURN
8000 -
9000 L=1:F=1:N=0
9001 G=1+B(L)
9010 C(L+1+N)=A(F)
9020 IF A(F)=0 AND B(F)=0 THEN GOTO 9055
9030 L=L+1
9040 IF L<G THEN GOTO 9010
9050 F=F+B(F)+1:G=G+B(F):N=N+1:GOTO 9010
9055 'FOR O=1 TO L:PRINT A(O),B(O),C(O):NEXT
9060 NB=0
9080 INPUT "nom du fichier converti (UNUIT:NOM.EXT)?",NOM$
9090 OPEN "o",#2,NOM$
9095 NB=NB+1
9100 FOR I=NB TO NB+B(NB)
9110 IF I<>NB THEN PRINT #2,A(I),B(I),C(I)
9120 NEXT
9130 IF A(I)=0 AND B(I)=0 THEN ELSE :NB=NB+B(NB)+1:GOTO 9100
9140 CLOSE#2
9150 CLEAR
9160 GOTO 10
9991 LOCATE 22,1,1:PRINT "tes vous srs? (N/O)"
9992 LOCATE 22,30,1:INPUT "",O$
9993 IF O$<>"o"AND O$<>"O" AND O$<>"N" AND O$<>"n" THEN BEEP:GOTO 9992
9994 IF O$="o" OR O$="O" THEN SYSTEM ELSE LOCATE 22,1,1:PRINT "
":RETURN
9996 LOCATE 20,1,1:SOUND 450,2:PRINT "vous allez sortir de sirs":GOSUB 9991
9997 RETURN
9998 CLEAR:GOTO 10

```


MENU.PRG

```

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE On
SET CONFIRM ON
SET SCOREBOARD OFF

```

```

SELECT 1
USE A:PROJET
SELECT 2
USE A:ROUTE

```

```
DO WHILE .T.
```

```
* ---Display menu options, centered on the screen.
```

```
* draw menu border and print heading
```

```
CLEAR
```

```
@ 1, 20 TO 10,62 DOUBLE
```

```
@ 2,32 SAY [M E N U   T H I E S]
```

```
@ 3,21 TO 3,61 DOUBLE
```

```
* ---display detail lines
```

```
@ 4,26 SAY [1. DONNEES OCCUPATION DU SOL]
```

```
@ 5,26 SAY [2. DONNEES ETAT ROUTES]
```

```
@ 6,26 SAY [3. RECHERCHE PAR LAT. ET LONG.]
```

```
@ 7,26 SAY [4. RECHERCHE PAR LOCALITE]
```

```
@ 8,26 SAY [5. RECHERCHE SUR LES ROUTES]
```

```
@ 10, 26 SAY '0. EXIT'
```

```
STORE 0 TO selectnum
```

```
@ 15,33 SAY " select      "
```

```
@ 15,42 GET selectnum PICTURE "9" RANGE 0,5
```

```
READ
```

```
DO CASE
```

```
  CASE selectnum = 0
```

```
    SET BELL ON
```

```
    SET TALK ON
```

```
    CLEAR ALL
```

```
    RETURN
```

```
  CASE selectnum = 1
```

```
    * DO DONNEES OCCUPATION DU SOL
```

```
    SELECT 1
```

```
    do ECRANO
```

```
  CASE selectnum = 2
```

```
    * DO DONNEES ETAT ROUTES
```

```
    SELECT 2
```

```
    do ECRAN4
```

```
  CASE selectnum = 3
```

```
    * DO RECHERCHE PAR LAT. ET LONG.
```

```
    SELECT 1
```

```
    do ECRAN1
```

```
  CASE selectnum = 4
```

```
    * DO RECHERCHE PAR LOCALITE
```

```
SELECT 1
do ECRAN2
```

```
CASE selectnum = 5
* DO RECHERCHE SUR LES ROUTES
SELECT 2
do ECRAN3
```

```
ENDCASE
```

ECRAN0.PRG ET ECRAN1.PRG

```
ENDDO T
```

```
RETURN
```

```
* EOF: MENU.PRG
```

```
@ 11, 0 clear
```

```
do while .T.
```

```
@ 13, 0 to 22, 75
```

```
append blank
```

```
@ 12, 23 SAY "ENREGISTREMENT DONNEES"
```

```
@ 16, 2 SAY "Localit :"
```

```
@ 16, 14 GET PROJET->NOM
```

```
@ 16, 44 SAY "Latitude :"
```

```
@ 16, 57 GET PROJET->LATIT
```

```
@ 18, 44 SAY "Longitude :"
```

```
@ 18, 57 GET PROJET->LONGIT
```

```
@ 20, 2 SAY "Occupation du sol :"
```

```
@ 20, 22 GET PROJET->SOL
```

```
read
```

```
wait "Avez vous d'autres donnees O/N " to reponse
```

```
@ 11, 0 clear
```

```
if upper(reponse)="O"
```

```
loop
```

```
else
```

```
exit
```

```
endif
```

```
enddo
```

```
return
```

```
store " " to latite
```

```
store " " to longite
```

```
do while .T.
```

```
@ 11, 0 clear
```

```
@ 12, 1 to 19,76
```

```
@ 24, 10 SAY "Donner la latitude de la localit "
```

```
@ 14, 4 SAY "Latitude Nord :"
```

```
@ 14, 25 GET LATITE
```

```
Read
```

```
@ 24, 10 say "Donner la longitude de la localit "
```

```
@ 16, 4 SAY "Longitude Ouest :"
```

```
@ 16, 25 GET LONGITE
```

```
Read
```

```
@ 24, 10 clear
```

```
locate for latit=latite .and. longit=longite
```

```
IF EOF()
```

```
@ 24, 10 say " Les donnees sont invalides , taper une touche "
```

```
wait
```

```
loop
```

```
endif
```

```
@ 18, 4 SAY "Occupation du sol :"
```

```
@ 18, 25 GET PROJET->SOL
```

```
@ 24, 10 SAY "une touche pour continuer"
```

```
wait
```

```
exit
```

```
enddo
```

```
@ 11, 0 clear
```

```
return
```

ECRAN1.PRG (SUITE)

```

TORE " " TO NOME
DO WHILE .T.
@ 11, 0 CLEAR
@ 12, 0 to 19, 79
do while .t.
@ 24, 10 say "Donner la localite "
@ 14, 3 SAY "Localit : "
@ 14, 23 GET NOME
READ
@ 24, 10 CLEAR
locate for nom=nome
if eof()
@ 24, 10 say "localit inexistante , taper une touche "
wait
@ 24, 10 clear
loop
endif
@ 14, 40 SAY "Latitude Nord : "
@ 14, 60 GET PROJET->LATIT
@ 16, 40 SAY "Longitude Ouest : "
@ 16, 60 GET PROJET->LONGIT
@ 18, 3 SAY "Occupation du sol : "
@ 18, 23 GET PROJET->SOL
@ 24, 10 SAY "une touche pour continuer"
WAIT
exit
enddo
@ 11, 0 CLEAR
RETURN
STORE " " TO NOME
DO WHILE .T.
@ 11, 0 CLEAR
@ 12, 0 to 19, 79
do while .t.
@ 24, 10 say "Donner la localite "
@ 14, 3 SAY "Localit : "
@ 14, 23 GET NOME
READ
@ 24, 10 CLEAR
locate for nom=nome
if eof()
@ 24, 10 say "localit inexistante , taper une touche "
wait
@ 24, 10 clear
loop
endif
@ 14, 40 SAY "Latitude Nord : "
@ 14, 60 GET PROJET->LATIT
@ 16, 40 SAY "Longitude Ouest : "
@ 16, 60 GET PROJET->LONGIT
@ 18, 3 SAY "Occupation du sol : "
@ 18, 23 GET PROJET->SOL
@ 24, 10 SAY "une touche pour continuer"
WAIT
exit
enddo
@ 11, 0 CLEAR
RETURN

```

ECRANA.PRG

```
store "                " to latite
store "                " to longite
do while .T.
@ 11, 0  clear
@ 12, 1  to 19,76
@ 24, 10 SAY "Donner la latitude de la localit  "
@ 14,  4 SAY "Latitude Nord      :"
@ 14, 25 GET  LATITE
Read
@ 24, 10 say "Donner la longitude de la localit "
@ 16,  4 SAY "Longitude Ouest   :"
@ 16, 25 GET  LONGITE
Read
@ 24, 10 clear
locate for latit=latite .and. longit=longite
IF EOF()
    @ 24, 10 say " Les donnes sont invalides , taper une touche "
    wait
    loop
endif
@ 18,  4 SAY "Occupation du sol  :"
@ 18, 25 GET  PROJET->SOL
@ 24, 10 SAY "une touche pour continuer"
wait
exit
enddo
@ 11, 0 clear
return
```

ECRAN2:PRG

```
@ 11, 0 clear
do while .T.
@ 13, 0 to 22, 75
append blank
@ 12, 23 SAY "ENREGISTREMENT DONNEES"
@ 16, 2 SAY "Localit :"
@ 16, 14 GET PROJET->NOM
@ 16, 44 SAY "Latitude :"
@ 16, 57 GET PROJET->LATIT
@ 18, 44 SAY "Longitude :"
@ 18, 57 GET PROJET->LONGIT
@ 20, 2 SAY "Occupation du sol :"
@ 20, 22 GET PROJET->SOL
read
wait "Avez vous d'autres donnees O/N " to reponse
@ 11, 0 clear
if upper(reponse)="O"
loop
else
exit
endif
enddo
return
```

ECRAN3.PRG

```
STORE " " TO NOME1
STORE " " TO NOME2
DO WHILE .T.
@ 11, 0 CLEAR
@ 24, 10 say "DONNER LA PREMIERE LOCALITE"
@ 12, 1 TO 20,76
@ 14, 4 SAY "LA LOCALITE 1:"
@ 14, 25 GET NOME1
  READ
@ 24, 10 SAY "DONNER LA DEUXIEME LOCALITE"
@ 16, 4 SAY "LA LOCALITE 2:"
@ 16, 25 GET NOME2
  READ
@ 24, 10 CLEAR
LOCATE FOR NOM1=NOME1 .AND. NOM2=NOME2
IF EOF()
  LOCATE FOR NOM1=NOME2 .AND. NOM2=NOME1
ENDIF
IF EOF()
@ 24,10 SAY "LES DONNEES SONT INVALIDES , une touche... "
  Wait
  exit
ENDIF
@ 18, 4 SAY " ROUTE LES RELIANT : "
STORE ETAT TO ETAT1
@ 18, 25 GET ROUTE->ETAT
@ 24, 10 SAY "UNE TOUCHE POUR CONTINUER"
  WAIT
  EXIT
ENDDO
RETURN
```

ECRAN 4.PRG

```
@ 11, 0 CLEAR
DO WHILE .T.
@ 14, 0 TO 22, 75
  APPEND BLANK
@ 12, 23 SAY "ENREGISTREMENT DONNEES"
@ 16, 2 SAY "LOCALITE1:"
@ 16, 14 GET NOM1
READ
@ 18, 2 SAY "LOCALITE2:"
@ 18, 14 GET NOM2
READ
@ 20, 2 SAY " ETAT DE LA ROUTE : "
@ 20, 22 GET ETAT
  READ
  WAIT "AVEZ VOUS D'AUTRES DONNEES O/N" TO REPONSE
@ 11,0 CLEAR
  IF UPPER(REPONSE)="O"
  LOOP
  ELSE
  EXIT
  ENDIF
ENDDO
RETURN
```