

Version au 4 avril 1988: annule et remplace le texte du 17 novembre 1987

**NEPTUNE, LOGICIEL DE TRAITEMENT DE DONNEES DESCRIPTIVES:
MISE EN PLACE ET TRAITEMENT D'UNE APPLICATION**

**Anne-Marie AUBRY
Centre Orstom de Bondy, F 93140 Bondy**

RESUME

Le système NEPTUNE est destiné à traiter les données descriptives entrées en langage proche du langage naturel; ce qui implique un apprentissage très léger de la part des utilisateurs.

Afin d'expliquer et de montrer la mise en place d'une application utilisant le logiciel Orstom Neptune, nous avons traité complètement, dans un but didactique, une application à la flotte des compagnies aériennes mondiales. A chaque étape, son emploi détaillé est illustré par des exemples.

On montre comment, à partir de la réflexion sur la structure des informations qu'on veut traiter, on constitue d'abord un répertoire (ou nomenclature), dans un module qui est indépendant de la banque elle-même, contenant la "description des données" (au sens SGBD). C'est en se référant à ce répertoire que le système reconnaîtra, vérifiera et structurera les données qui lui sont fournies.

Chaque type d'exploitation, y compris la constitution du répertoire (déjà cité), les mises à jour et les interrogations de la banque, sont expliqués sur des exemples commentés. Les entrées et sorties d'ordinateur correspondant à chaque type de processus sont fournies en annexe, et permettent de comprendre les résultats que l'on peut obtenir grâce à un tel système.

On montre dans quelle mesure ce système reste "ouvert" et on donne des indications pratiques, à la faveur de l'expérience acquise par des exploitations "en vraie grandeur" (fichiers Bénin et Sénégal) qui ont montré la fiabilité et l'utilité de ce système.

La liste des programmes sources écrits en PL1, utilisables sur le Centre de Calcul du CNRS-CIRCE, est fournie en annexe.

PLAN

- I PRESENTATION
 - II TRAITEMENT D'UNE APPLICATION
 - III COMPLEMENTS DE MODE D'EMPLOI
 - IV ENVIRONNEMENT D'UNE APPLICATION
 - V INFORMATIONS SUR L'EXPLOITATION ACTUELLE DU SYSTEME
 - VI CARACTERISTIQUES TECHNIQUES
 - VII CONCLUSION
 - VIII REFERENCES BIBLIOGRAPHIQUE
 - IX ANNEXES
-

I PRESENTATION

Le logiciel présenté ici est destiné à traiter des descriptions faites par un observateur utilisant les mots et expression d'une TERMINOLOGIE PROCHE DU LANGAGE NATUREL (terminologie usuelle), qu'il a lui-même définie et qu'il peut enrichir comme il le désire.

Il n'est pas lié à un domaine déterminé.

Les données peuvent être qualitatives, ou qualitatives ordonnées, ou peuvent aussi être assorties de quantités entières.

Le système traite des données présentant deux niveaux hiérarchiques (arborescence), le niveau unité et le niveau sous-unité (structure père-fils de l'entité famille). La sous-unité est, à l'intérieur de l'unité, un sous-ensemble identifiable de données, dont on rencontre plusieurs occurrences. Le nombre de sous-unités par unité et le nombre de variables de chacun des niveaux varient selon les applications.

Le système est modulaire, constitué de trois ensembles en interaction:

- Le répertoire ou un des répertoires équivalents
- les programmes en binaire exécutable
- les données elles-mêmes

C'est sur les données et sur le répertoire que l'utilisateur agit.

D'abord, c'est lui-même qui constitue le REPERTOIRE, fichier séquentiel où il décrit ses données (sens SGBD, Système de Gestion de Banque de Données) et ce qu'il autorise à entrer dans la banque.

Il y donne les domaines sémantiques des variables (domaines où elles prennent leur valeur).

Quant au fichier des DONNEES, c'est le système qui se charge de le formater et d'y enregistrer les codes - de façon transparente à l'utilisateur - en se servant en même temps du REPERTOIRE qui lui est désigné et des DONNEES SAISIES par l'utilisateur DANS LE LANGAGE USUEL choisi, SANS ORDRE IMPOSE et SANS FORMAT : le point est utilisé comme séparateur de données et il n'y a pas d'autres contraintes d'écriture .

Entrée des données

Les données sont introduites par deux clefs logiques correspondant à la structure unité et sous-unité (dans l'unité). Ces clefs sont deux variables choisies par l'utilisateur dont les données entières identifient respectivement l'unité à l'intérieur de la banque et, relativement à cette unité, la sous-unité à l'intérieur de l'unité.

Le schéma d'introduction des données est le suivant:

clef logique de l'unité,
données dans un ordre quelconque,
information additionnelle éventuelle,
ensuite, clef logique de sous-unité, données, information additionnelle éventuelle,
autant de fois qu'il y a de sous-unités, leur nombre étant fixé lors de la constitution du répertoire.

Il est à noter que si on entre plusieurs données valides pour une variable, c'est la dernière qui est enregistrée. D'autre part, on n'entre que les données effectivement présentes.

Validation

La validation des données saisies est réalisée par le SYSTEME, AUTOMATIQUEMENT par comparaison avec le répertoire auquel on fait référence. Le système vérifie en particulier que les données appartiennent bien aux domaines de valeurs qui sont précisées dans le répertoire, et il ANNOTE, en sortie, les données non reconnues.

Mises à jour

Pour faire des mises à jour, on peut agir à la fois sur le REPERTOIRE et sur les DONNEES.

- S'il s'agit d'erreurs de saisie, il y a deux MODES de correction: un mode "local" qui permet de modifier une seule donnée si l'on veut, un mode "global" qui permet de remplacer toutes les données d'une unité et de ses sous-unités (cf II,§5,6 et annexe p.25).

Dans le mode local, on saisit uniquement la donnée correcte introduite par sa clef d'unité, et de sous-unité quand c'est le cas. On peut utiliser les deux modes conjointement au cours du même traitement.

IL N'Y A PAS DE LANGAGE DE MANIPULATION A APPRENDRE POUR ENTRER ET METTRE A JOUR LES DONNEES. IL SUFFIT DE CONNAITRE LES MOTS-CLEFS "GLOBAL" ET "LOCAL" QUI INTRODUISENT LE MODE DE SAISIE, ET LE VOCABULAIRE DU REPERTOIRE.

- On peut aussi mettre à jour le répertoire, en y ajoutant des données qu'ensuite il reconnaitra. Les variables quantitatives y ont leur bornes (entiers relatifs) maximum et minimum: ces bornes peuvent être modifiées. Les variables qualitatives y ont toutes leurs valeurs (expressions sous forme de chaînes de caractères) admissibles, ces listes peuvent être enrichies de termes supplémentaires.

Interrogation et extraction sélective

Il n'y a PAS non plus A CONNAITRE UN LANGAGE SPECIAL POUR INTERROGER la banque. L'expression de sélection ne fait appel qu'à la TERMINOLOGIE USUELLE répertoriée, aux OPERATEURS RELATIONNELS stricts (sous la forme $< > =$), et aux OPERATEURS LOGIQUES ET, OU, NON (sous la forme $\& ! ^$). De plus, on peut utiliser les parenthèses pour condenser son écriture. Enfin celle-ci est écrite sans format.

L'expression élémentaire est de la forme $A > B$, $A < B$, ou $A = B$, où, dans le cas le plus général (cf II,§8), A est une variable du répertoire et B est une donnée ou un entier. L'utilisation des opérateurs $\&$, $!$, $^$ et "vide" sur ces opérandes permettent, avec les règles de priorité classique et les parenthèses de formuler toutes les questions.

L'interrogation fournit l'impression, en ordre croissant, des numéros d'enregistrements (unités) répondant à la question. Ces enregistrements peuvent être consultés à leur tour, constituant un fichier de mêmes caractéristiques physiques et logiques que le fichier principal. On peut donc poursuivre les opérations de recherche de façon répétée (nested search). A n'importe quelle étape, on peut éditer le fichier extrait.

Impression/édition

Au cours de tous ces traitements, ce qui est entré au clavier est imprimé tel quel, sauf dans le cas où, pour faire repérer l'anomalie détectée, il remplace le point terminal de la donnée par un point d'interrogation.

On peut aussi éditer le fichier, avec mise en page, totalement ou en donnant une liste de numéros.

Synopsis

Le système fournit un synopsis graphique compact de quantité d'information stockée et, d'autre part, les fréquences d'utilisation des données.

Les fichiers d'accès séquentiel obtenus à partir de la banque peuvent être traités par les logiciels statistiques du centre de calcul (CIRCE) où elle est implantée.

En résumé, le système est indépendant de la terminologie choisie, déchargeant l'utilisateur d'une codification fastidieuse et de l'apprentissage d'un langage de commande le plus souvent lié au SGBD quel qu'il soit. L'utilisateur doit essentiellement

définir la terminologie utilisée dans son domaine, qui est enregistrée dans le répertoire et sert de référence pour tous les traitements.

II TRAITEMENT D'UNE APPLICATION

1. INTRODUCTION

On se propose de montrer comment se fait la mise en service de NEPTUNE sur des données de compagnies aériennes mondiales. On a collecté la plupart de ces données dans une revue d'avionique dont on voit ci-dessous l'extrait concernant Air Guinée et Air Mauritius (Interavia 10/85).

1 F27 Mk200 / 1 F27 Mk400 / 1 DO228-100 / 2 L-382*

TRAFFIC

Passengers(000): 54.5. RPK(m): 19.4. Pax Vt: 59.3%. CRTK(m): 0.1. Total RTK(m): 1.9. Weight Vt: 58.5%.

PRINCIPAL ROUTES

Domestic, Africa.

AIR GUINEE (NR)

6 Av. de la République, B.P. 12, Conakry, Guinée.

☎ 429 80. SITA: OKYDGGI

COMPANY DATA

Employees: 400 approx.

FLEET

1 A300C4 / 1 B727 / 1 B737-200 / 2 Il-18 / 1 Yak-40 / 1 An-12 / 3 An-24.

AIR MAURITIUS

MRU 13 MK

1st Floor, Rogers House, 5 President John Kennedy Street, Port Louis, P.O. Box 44 Mauritius.

☎ 087700 ☎ 44151 SITA: MRUMK

EXECUTIVES

Chmn & MD: H. K. Tirvengadam. Gen Mgr Planning, Development & Pools: H. L. Sikk.

Gen Mgr Finance & Admin: L. G. Tyack. Asst to Chmn-Comm: S. Seegobin.

COMPANY DATA

Capital: Mrs2.6m. State holding: 51%. Employees (3/85): 530.

FLEET

1 B747SP* / 2 B707 / 1 B737* / 1 BAe 748 / 2 Twin Otter. On order: 1 ATR 42.

TRAFFIC (3/85)

Passengers(000): 157.4. RPK(m): 513. Pax Vt: 71.35%. CRTK(m): 10.3. Total RTK(m): 56.4. Weight Vt: 68.63%.

PRINCIPAL ROUTES

Europe, Africa, Asia, Domestic.

Abréviations utilisées pour les fonctions: Administration: Admin; Assistant: Asst; Chairman: Chmn; Chief Executive: C. Exec; Chief Executive Officer: CEO; Chief Operating Officer: COO; Commercial: Comm; Deputy: Dep; Director: Dir; Director General: DG; Engineering: Engrg; Executive Vice President: Exec. VP; Flight Operations: Flt Op; General Manager: Gen Mgr; International: Intl; Maintenance: Maint; Manager: Mgr; Managing Director: MD; Marketing: Mktg; Operations: Ops; President: Pres; Secretary: Secy; Secretary General: Secy Gen; Senior Vice President: Sen VP; Technical: Tech; Traffic: Tlc; Vice President: VP. **Autres abréviations:** * exploité en location; + loué à l'autre transporteur; n/a (not available) – non disponible; Pax – passagers; RPKm – passagers-km payants; CRTK – t/km payantes fret; RTK – t/km payantes; LF – coefficient de chargement. L'abréviation NR (no reply) indique que, n'ayant pas reçu de réponse à notre questionnaire, nous avons tiré les informations de *Interavia ABC 1985* et *IATA World Transport Statistics 1984*.

1088

INTERAVIA 10/1985

Chaque compagnie a donné des renseignements d'ordre général (localisation géographique, réseau couvert, etc.) et administratif (effectif, trafic, etc.) ainsi que la composition de sa flotte, en indiquant la quantité de chaque type d'appareil en service ou en commande.

On va montrer comment procéder pour constituer le fichier, pour le mettre à jour directement, pour l'interroger sur les valeurs des variables de façon à extraire automatiquement les compagnies présentant telle caractéristique ou combinaison de caractéristiques.

On a mis en place cet exemple dans un but didactique, pour montrer comment on procédait, et on l'a choisi car il est d'un domaine pouvant intéresser un public non spécialisé en géographie physique et plus particulièrement en analyse typologique des paysages, domaine sur lequel le logiciel est exploité actuellement (voir le chapitre V). En outre, le sujet n'est pas habituellement traité par les médias.

2. DONNEES A ENREGISTRER ---> CONSTITUTION DU REPERTOIRE

C'est l'utilisateur, comme on l'a dit dans la présentation, qui décrit ses données (au sens SGBD) pour le système dans le répertoire. Il y indique d'abord quelles données il veut enregistrer, sous quelle forme il pourra lui-même les saisir et sous quel code (nombre entier relatif) il charge le système de les enregistrer sans que lui-même intervienne. On commence donc par étudier les DONNEES SOURCES.

On voit qu'on dispose, pour chaque compagnie, d'un certain nombre de renseignements. On les désigne sous le nom de VARIABLES: en effet, pour chaque compagnie aérienne, chacune d'elles prend une valeur (DONNEE). L'ensemble de ces données constitue une UNITE LOGIQUE CARACTERISANT LA COMPAGNIE AERIENNE. On va s'intéresser plus particulièrement à un certain nombre de ces variables pour en enregistrer les données de façon à pouvoir y accéder directement par la suite, non seulement pour les actualiser, mais aussi pour faire des rapprochements entre les compagnies en interrogeant le fichier.

Ces variables sont les suivantes:

- _ le nom de la compagnie aérienne,
- _ la ville où se trouve le siège administratif,
- _ le nombre d'employés,
- _ le pourcentage de l'état dans le capital
- _ le réseau,
- _ le nombre de passagers en milliers,
- _ le pourcentage d'occupation des sièges
- _ le nombre de passagers-kilomètre payants, en millions,
- puis, pour la composition de la flotte:
- _ le nombre de EMB120 en service,
- _ le nombre de EMB120 en commande,
- _ le nombre de CL50 en service,
- CL50 en commande,
- _ ... etc.
- _ le nombre de MD87 en service,
- _ le nombre de MD87 en commande

Ce qu'on remarque dès à présent, c'est que:

a) parmi ces variables, certaines sont QUALITATIVES (ou "nominales"), par exemple le nom de la compagnie, celui de la ville du siège, ou encore le réseau. Leur donnée s'exprime à l'aide de mots. Les autres sont QUANTITATIVES (ou "ordinales"), par exemple le nombre d'employés ou le nombre de Boings 707 en service. Leur donnée est un nombre entier.

b) pour chaque variable il y AU PLUS UNE donnée. C'est évident pour la flotte, les quantités d'appareils d'un type déterminé ne sont données que lorsqu'elles sont positives. Si, par exemple, il n'y a pas de A310 commandés ou en service, ces deux variables seront vues comme ayant des données MANQUANTES ou absentes. Et, pour ce qui concerne les informations générales, des données manquent parce que certaines compagnies n'ont répondu qu'incomplètement au questionnaire.

c) pour une compagnie, un type d'appareil donné peut revenir deux fois: dans les variables quantité en service et quantité commandée. Ceci est vrai pour toutes les variables quantité d'un type d'appareil donné.

3. STRUCTURATION DES DONNEES EN UNITE/SOUS-UNITES

On a dit dans la présentation que le logiciel est prévu pour restituer une structure arborescente à deux niveaux observée dans les données. C'est le cas de l'exemple qu'on a choisi.

Toutes les variables d'ordre général, géographique, administratif, etc., n'apparaissent qu'une fois et caractérisent directement la compagnie, elles sont du niveau d'entrée dans l'unité (niveau 1).

Pour le sous-ensemble des variables quantités d'un type d'appareil, en service d'une part et en commande d'autre part, une approche consiste à le subdiviser en deux SOUS-ENSEMBLES (sous-unités) de STRUCTURE IDENTIQUE, composés des variables "quantité de EMB120" à "quantité de MD87", dont l'un caractérisera la flotte en service, et l'autre la flotte en commande. Par conséquent les variables quantités des différents types d'avion possibles définissent la sous-unité. Une variable clef d'accès pour cette sous-unité permettra de l'identifier de façon unique DANS L'UNITE.

En entrée des données, la saisie de la valeur de la clef d'accès à cette sous-unité, (on choisit par exemple "F=1" pour la flotte en service et "F=2" pour la flotte en commande), permettra d'accéder DIRECTEMENT à la sous-unité désignée, c'est-à-dire, par exemple, au sous-ensemble des quantités des différents types d'appareils en service.

C'est l'approche qu'on a retenue pour traiter cet exemple, mais on verra plus loin qu'on aurait pu faire un autre choix, avec les conséquences que cela entraîne au niveau de l'interrogation.

4. REPERTOIRE

Ayant déterminé la structure des données, on constitue la nomenclature (ou répertoire) où on va décrire les données (sens SGBD), c'est-à-dire énumérer pour le système ce qu'on veut enregistrer (quelles variables), dans l'ordre où on veut qu'il les enregistre, respectivement au niveau unité puis sous-unité (la structure), quelles "valeurs" pourront prendre ces variables (domaine sémantique), et sous quelle(s) forme(s) elles pourront être saisies. (Voir en annexe pp.1-5 le contenu de la nomenclature).

En pratique, la constitution du répertoire est excessivement simple (cf annexe p.5bis pour un extrait):

On donne d'abord tout ce qui concerne l'unité (pp.1-3), variable par variable, en commençant par ce qu'on choisit comme clef d'accès, on procède ensuite de même pour la sous-unité (pp.3-5). Ces deux ensembles sont introduits respectivement par les mots-clefs UNIT et SUBUNIT.

On met toutes les variables qu'on a citées plus haut (en 1) en leur donnant un nom et on y décrit le domaine où elles prennent leurs valeurs, à savoir:

- pour une qualitative, la liste de toutes les données rencontrées, à raison d'une donnée par ligne, 1 à 56 caractères alphanumériques,
- pour une quantitative, les valeurs minimum et maximum entre lesquelles elle pourra varier.

On a droit à deux noms synonymes pour chaque variable. Un signe "=" les séparant marque les quantitatives.

Remarque: on tient compte de la façon dont on saisira les données ultérieurement. En effet, pour les quantitatives, on saisit le nom de la variable (sous une des deux formes) suivi de "=" puis de l'entier représentant la valeur prise par la variable. Et pour les qualitatives, on saisit une des données (sous une de ses formes synonymes) de la liste qui suit le nom de la variable, sans saisir ce nom.

Les clefs d'accès à l'unité (annexe p.1), d'une part, à la sous-unité (p.3), d'autre part, se présentent sous la forme de n'importe quelles variables quantitatives puisqu'elles sont définies par la donnée d'un numéro (entier positif): c'est leur place en première position après les mots de contrôle "unit", respectivement "subunit", qui leur donne automatiquement ce rôle particulier. La clef d'accès à l'unité d'enregistrement (= enregistrement logique), à savoir la "compagnie aérienne" sera la variable de nom "C" (qui prendra ses valeurs entre "C=1" et "C=53" dans ce cas précis où l'on va rassembler les données de 53 compagnies). Puis le nom de la compagnie sera dans la variable "COMPAGNIE" de forme synonyme abrégée "C\$" (p.1), qu'on fait suivre de la liste des noms des 53 compagnies présentes, avec leurs synonymes, les codes internationaux.

On remarque que, à cette étape, on donne aussi au système le code entier (relatif) sous lequel il stockera (archivera) la chaîne de donnée qualitative saisie. On attribue aux données synonymes le même code.

On dresse ainsi une sorte d'"inventaire" des variables, en présentant leurs deux noms équivalents, toutes les données qualitatives rencontrées, avec leurs synonymes et les valeurs maximum et minimum entre lesquelles varient les quantitatives.

On choisit, pour les variables retenues, les noms suivants:

Pour le niveau unité,

C	(clef d'accès) (p.1)	
COMPAGNIE		C\$ (synonyme abrégé)
SIEGE	S\$	
EMPLOYES	E	
CAPITAL % ETAT		C%
RESEAU	R\$	
PASSAGERS EN 10**3	P	
OCCUPATION DES SIEGES	P%	P%
PASSAGERS KM PAYANTS EN 10**6	PKM	

Pour le niveau sous-unité,

F	(clef) (p.3)	
EMB120		EMB
CL50		C50
CL215		C215
... etc.		
MD87		MD87

Comme on l'a dit, QUAND ON SAISIRA "F=1" ou "F=2", ces données de clef d'accès à la sous-unité à l'intérieur d'une unité déterminée, permettront d'accéder directement à la sous-unité désignée :

F=1 introduira la flotte en service et F=2 la flotte en commande.

Au niveau 2, on n'a que des variables quantitatives. On a donc fait suivre chaque nom de type d'appareil de la fourchette des valeurs permises. "Nombre d'appareils de type X" est représenté directement par "X". Ce qui en saisie donnera par exemple "EMB120=5." ou "EMB=5" pour "nombre de EMB120 égale 5".

On donne aussi à cette étape le code du "manquant" pour le système. Pour sa part, l'utilisateur aura l'occasion de s'en servir quand il interrogera le fichier, avec des conditions portant sur la présence des variables (cf §8)

Ce code est ici choisi égal à "-1". En effet, la valeur "0" ne peut pas être retenue parce que c'est une valeur qui peut être réellement prise par certaines variables (par exemple C%). On remarque enfin qu'on a choisi pour minimums des quantités des différents types d'appareils la valeur "1" et non pas "0", car une quantité nulle d'un type d'avion correspond à son absence et on ne saisit pas ce qui manque ou n'existe pas.

5. SAISIE-VALIDATION

Faisant suite au contenu de la nomenclature, on a en annexe (pp.6-7) une image de la saisie faite sur le micro-ordinateur portable et autonome Epson HX20 dont l'imprimante incorporée a des lignes de 24 caractères.

On a saisi les informations sur 53 compagnies aériennes du Bassin Méditerranéen et d'Afrique. On attribue à chaque compagnie un NUMERO DE CLEF, de 1 à 53, qui les IDENTIFIE DE FACON UNIQUE

FORME DES ELEMENTS D'INFORMATIONS - OU "DONNEES"- QUE L'ON SAISIT

1) Pour les variables qualitatives, c'est la chaîne de caractères constituant la donnée que l'on saisit, sans fournir le nom de la variable. On saisit indifféremment un des synonymes.

Exemple: on voit page 6 que, pour C=1, on a saisi "AH", synonyme de "AIR ALGERIE" (p.1), donnée de la variable "COMPAGNIE"..

2) Pour les variables quantitatives, c'est le nom de la variable (sous une des deux formes synonymes) suivi du signe "=" et de l'entier qu'on saisit. Dans ce cas, c'est cet ensemble qui constitue la donnée.

Exemple: page 6, pour C=1, on a saisi "P=4061", donnée de la variable "PASSAGERS EN 10**3, de synonyme "P" (p. 3), signifiant que le nombre de passagers (en milliers) est 4061.

REMARQUE IMPORTANTE: la saisie d'une donnée, telle qu'on vient de la définir, qualitative ou quantitative, doit être immédiatement SUIVIE de la saisie d'UN POINT qui la délimite pour le système.

Procédé pratique

La saisie est faite en continu, caractère par caractère, donnée après donnée, sans format (chaque donnée étant terminée par un point évidemment). On ne saisit rien pour les données manquantes.

Si on utilise des terminaux du CIRCE ou du Mini6 Orstom -terminal du CIRCE- qui ont des écrans de 80 colonnes, on peut faire la saisie des données "au kilomètre", pour employer le terme imagé utilisé en Traitement de Texte.

Dans le cas d'une saisie de données sur le terrain, on a besoin d'enregistrer les données sur un matériel léger, robuste, au fur et à mesure qu'elles sont observées (pas d'erreurs de transcription). On utilise actuellement le micro-ordinateur portable autonome Epson HX20 (cf rapport de mission au Sénégal en novembre 86).

Sur le HX20, la saisie des données se fait par tranches de 1 à 255 caractères maximum. C'est-à-dire qu'on introduit dans le micro, par "return", des chaînes de moins de 255 caractères. Le respect de cette limite n'offre pas de difficulté car on utilise, pour la saisie sur ce matériel, un écran virtuel de 255 caractères de large (cf le mode d'emploi de saisie pour NEPTUNE sur HX20). Dans ce cas, pour être traitées par NEPTUNE, les chaînes sont transférées sur le Mini6 sous forme de lignes de 80 caractères, le point restant le séparateur des données.

Contenu

On a saisi les données pour chaque compagnie, compagnie par compagnie, ici dans l'ordre non imposé croissant des numéros de clefs qu'on a attribués, "C=1" à "C=53", en commençant toujours par la clef d'accès à l'enregistrement qui est saisie comme n'importe quelle quantitative.

Quand on a fini d'entrer les informations générales (variables "COMPAGNIE" à "PASSAGERS KM PAYANTS EN 10**6"), on fournit la clef d'accès à la sous-unité en indiquant sa valeur "F=1" (flotte en service) ou "F=2" (flotte commandée), puis les données concernant la sous-unité.

Exemple: page 6, pour C=1 et F=1, on a saisi "A300=8" et "B737=6", signifiant qu'il y a 8 appareils A300 (Airbus) et 6 appareils B737 (Boeing) en service.

Ce qui peut faciliter la saisie, indépendamment d'autres propriétés caractéristiques dont nous parlerons plus loin, c'est que le système ignore

- une suite de points ou des blancs entre deux points,
- des blancs devant une donnée, "donnée" ayant l'acception qu'on a définie, qu'elle soit qualitative ou quantitative, ce qui autorise le passage à la ligne suivante après un point,
- pour les quantitatives, des blancs de part et d'autre du signe="".

Sur cet exemple, on peut voir un certain nombre de propriétés de la saisie.

Ainsi, après la clef d'accès à l'enregistrement, les informations générales sont entrées dans n'importe quel ordre.

Exemple: pour C=11, (p.6), la donnée de la variable "RESEAU", c'est-à-dire "AF,EU", est entrée la dernière, alors que l'on voit dans le répertoire, p.3, qu'elle se trouve entre les variables "E" et "P".

De même, les sous-unités peuvent être entrées dans n'importe quel ordre, ainsi que les données à l'intérieur de celles-ci;

Exemples: p.6, pour C=34, "F=2" est saisi avant "F=1"

pour C=6, dans F=1, les avions ne sont pas dans l'ordre du répertoire (pp.4-5); les deux Iliouchine 18, le Yak 40 sont saisis avant les Boeing, tandis que l'Antonoff 12 et les trois Antonoff 24 le sont après les Iliouchine et le Yak.

Validation

En annexe, p.8, il y a ensuite la partie SORTIE IMPRIMÉE (ici en format réduit, comme toutes les sorties d'ordinateur) du programme de SAISIE-VALIDATION exécuté sur les données saisies..

C'est l'image exacte de l'entrée sur 80 colonnes, à ceci près:

des "?" remplacent les "." finals des données non reconnues. Ce sont les "ANNOTATIONS".

à la rencontre d'une nouvelle clef d'accès à l'enregistrement, le système imprime sur la gauche de la feuille, et en retournant à la ligne, le nombre d'annotations pour l'enregistrement précédent, s'il y en a. A la suite, il imprime le message "ECRITURE SUR LE FICHIER SORTIE".

Exemple: pour C=5, on a le message "1 ANNOTATION(S) ?", que l'on repère immédiatement (P5=66?), puis "ECRITURE SUR LE FICHIER SORTIE".

Parallèlement à cette sortie imprimée, le système enregistre dans le fichier les données valides sous forme de codes entiers relatifs (qu'on lui a indiqués dans le Répertoire). Ce fichier est sur le site central de l'ordinateur-hôte du CIRCE.

Le système y a enregistré toutes les données valides, c'est-à-dire toutes celles qu'il n'a pas imprimé annotées par "?".

Dans un but didactique, on donne en annexe pp.10-15, l'édition des enregistrements immédiatement après ce traitement, y compris ceux pour lesquels il y a eu des annotations: les numéros 5,10,13,26,27,29,35 que l'on rééditera après la mise à jour. On voit sur l'édition que les données qui avaient été annotées n'ont pas été enregistrées.

Sur la sortie imprimée de mise à jour présentée en annexe p.8, on va étudier en détail les différentes causes des annotations:

_ pour C=5, 10 et 13, ce sont des erreurs de saisie.

_ pour C=26, dans la sous-unité F=1, la donnée "DC10=6" est annotée, bien que sa forme soit correcte, parce que, dans le répertoire, on a prévu "5" comme valeur maximum pour la variable quantité de DC10 ("DC10="). En examinant les données de base, on voit que c'est cette limite qui a été mal choisie dans le répertoire, et qu'il y a bien six DC10 en service chez UTA.

_ pour C=27 et C=28, c'est exactement le même genre de cause.

_ pour C=35, c'est une erreur de saisie, mais cette fois sur la clef de sous-unité (on a saisi "F2=2", cette donnée est annotée car la variable "F2=" n'est pas reconnue), qui a fait que le système n'a pas pu identifier la donnée "A320=4" qui est de niveau 2. Elle aurait dû être introduite par une clef "F=1" ou "F=2" pour être reconnue et enregistrée.

On voit donc que les annotations ne sont pas dues aux mêmes causes, et cela aura de l'influence sur les mises à jour.

6. MISES A JOUR

Pour mettre à jour les données, on a la possibilité d'agir à la fois sur le répertoire et sur le fichier des données. Ce sont les deux interlocuteurs de l'utilisateur.

Mise à jour du répertoire

Dans notre exemple, on commence par mettre à jour le répertoire puisque les maximums autorisés pour les 3 variables quantités de DC10, A310 et A320 se sont montrés insuffisants. On remplace ces maximums dans le répertoire et on le recharge dans son ensemble. Il efface et remplace le précédent.

La mise à jour du répertoire peut se montrer nécessaire aussi pour réactualiser les données. Par exemple, des appareils en commande vont être mis en service ou de nouveaux appareils sont commandés. Le processus pratique est évidemment le même.

Mise à jour des données

La mise à jour des données consiste en partie en correction des erreurs de saisie signalées par le système.

On utilise le mode de saisie "local", en saisissant ce mot-clef comme une donnée.

On suit sur l'annexe p.16 l'image de la saisie.

On ne saisit que les données à insérer, introduites par la(les) clef(s) les identifiant.

Ainsi, pour la donnée "A320=4", il faut préciser s'il s'agit d'appareils en service ou en commande, c'est-à-dire à quelle sous-unité on a affaire, "F=1" ou "F=2", et, d'abord, bien entendu de quelle compagnie il s'agit, donc on précise l'unité à laquelle appartiennent ces données. On entre "C=35.F=2.A320=4."

La sortie obtenue après validation de la saisie par le système est à la suite sur la même page d'annexe. On voit qu'il n'y a pas d'annotation sur ce traitement. Les données correctes sont enregistrées dans la banque. L'édition des enregistrements ainsi mis à jour le montre en annexe p.17.

Remarques:

1) Dans cet exemple, ce sont en fait des insertions qu'on a dû faire pour mettre à jour les données. Mais dans le cas de remplacement de données, le procédé reste le même: on ne saisit que les données de remplacement correctement introduites par la clef désignant l'unité (et celle désignant la sous-unité quand c'est le cas): elles viendront automatiquement prendre la place de celles enregistrées auparavant.

2) On n'a pas illustré sur un exemple la propriété du système de rejeter les modifications proposées en mode local pour un enregistrement si seulement l'une d'elles est annotée. Dans ce cas, on n'a pas sur le listing de sortie d'impression le message "ECRITURE" Ceci a été choisi par sécurité.

Exemple fictif d'actualisation des données.

Supposons que la compagnie CYPRUS AIRWAYS reçoive deux A320 sur les quatre commandés. La mise à jour se fera en mode local, en entrant seulement, après la clef d'accès à l'enregistrement: "F=1.A320=2.F=2.A320=2." On remplacera ainsi "F=2.A320=4."

7. EDITION

On donne, en annexe, des exemples d'édition: pages 10 à 15, 17 et 19.

Au moment où on demande l'édition d'un enregistrement, le système imprime son contenu TEL QU'IL EST A L'INSTANT DE LA DEMANDE, en ignorant l'"historique" de sa saisie (mises à jour "globales" alternées avec mises à jour "locales", etc.).

L'impression est faite sur 132 colonnes. Les données sont imprimées dans l'ordre où on a mis les variables dans le répertoire: il y a d'abord celles du premier niveau, puis, successivement, celles de chaque sous-unité, dans l'ordre croissant des numéros de sous-unité. Cet ordre est respecté, quels que soit l'ordre et la façon (modes global ou local) dans lesquels les données ont été saisies.

Exemples:

pour C=11, p.10, la donnée du réseau reprend sa place,
pour C=6, p.10, pour F=1, les avions sont imprimés dans l'ordre des pays donné dans le répertoire,
pour C=34, p.13, F=1 est imprimé avant F=2, saisi avant lui.

Le système imprime les données quantitatives en prenant comme nom de variable la forme non abrégée. Par exemple, la variable nombre d'employés est imprimée "EMPLOYES=", alors qu'on l'a saisie "E=".

Il imprime les données qualitatives en sortant pour donnée le premier des synonymes fournis dans le répertoire quand c'est le cas (c'est en général le plus long, que l'on réserve à l'édition), et cela, quel que soit le synonyme qu'on a utilisé en saisie.

Par exemple, le nom de la compagnie a toujours été saisi sous son code international et est imprimé en clair.

Il retourne à la ligne à chaque nouvelle sous-unité.

Cependant on peut très bien, en constituant le répertoire (ou même seulement plus tard en le mettant à jour) lui demander de retourner à la ligne pour subdiviser les variables en sous-ensembles formant des paragraphes

L'édition peut être globale, c'est-à-dire que tous les enregistrements du fichier sont imprimés en ordre croissant de leur numéro de clef, quelle que soit l'historique de la saisie et, dans ce cas, on n'indique pas les numéros (annexe pp.10-15), ou locale: on donne une liste de numéros d'enregistrements à imprimer, en format libre, dans l'ordre où on les veut faire imprimer, qui peut ne pas être séquentiel croissant. Ils sont séparés par des virgules et une séquence de numéros, de n à m, s'écrit à l'aide de la préposition "A" entourée d'au moins un blanc;

Exemple: "3, 5 A 8, 15, 12, 20 A 22."

Voir les annexes p.17 et 19

8. SELECTION/INTERROGATION

On peut interroger la banque de données à l'aide d'une expression de sélection.

On peut voir (annexe p.18) des exemples élémentaires de telles interrogations. (La taille - réduite volontairement - de ce fichier aurait presque permis de faire ces traitements "à la main", ce qui est impensable dans des conditions normales d'utilisation).

L'expression de sélection est une EXPRESSION booléenne qui est VRAIE ou est FAUSSE sur CHACUN des ENREGISTREMENTS du fichier.

Le résultat est la LISTE DES NUMEROS d'enregistrement sur lesquels l'expression, tout d'abord listée, est VRAIE. Elle est suivie de l'indication de leur nombre, et du nombre total d'enregistrements explorés.

Commentaire des exemples présentés p.18.

- 1: Compagnies dont le siège est à Madrid.
- 2: id ayant des A320 (en service ou commandés)
- 3: id ayant des Airbus (indépendamment du modèle)
- 4: id ayant à la fois des Airbus (A300, A310 ou A320) et des Beoing (B707,B727, B737 ou B747)
- 5: id ayant à la fois des Airbus A320 et des Boeing B747 en commande
- 6: id dont le pourcentage de l'état dans le capital est inférieur à 30, cette donnée étant effectivement présente, (non égale manquante) et ayant plus de 5000 employés. (La réponse est la seule UTA).

Des questions plus pertinentes et significatives peuvent être posées. Il suffit d'appliquer les règles de constitution de l'expression de sélection exposées ci-dessous.

Une alternative à la seule interrogation est l'EXTRACTION SIMULTANEE DES ENREGISTREMENTS REpondant A LA QUESTION.

Le fichier ainsi constitué peut à son tour être interrogé, on recommence le processus, etc.: ce sont les extractions imbriquées (nested search).

A toute étape, on peut imprimer le contenu du fichier extrait, en tout ou en partie: voir le chapitre édition et, en annexe, l'édition p.19 où on voit les compagnies ayant des A320 (ce sont Air France, Cyprus Airways à Nicosie et Inex à Liubiana).

CONSTITUTION DE L'EXPRESSION DE SELECTION

L'expression de sélection va se suffire à elle-même: il n'y a PAS DE LANGAGE DE MANIPULATION à apprendre.

Elle est saisie SANS FORMAT.

Les ELEMENTS qu'elle peut contenir, à l'exclusion de tout autre, sont les suivants:

- a) les OPERATEURS de comparaison stricte (ou relationnels)
 - > pour "supérieur à",
 - < "inférieur à",
 - = "égal à",
- b) les OPERATEURS logiques
 - ^ pour "non",
 - ! "ou",
 - & "et",
- c) les PARENTHESSES ouvrante et fermante () ,
- d) les noms des VARIABLES , sous une des deux formes où ils figurent dans le répertoire,
 - les DONNEES des qualitatives, indifféremment sous une des formes synonymes,
 - les ENTIERS relatifs, y compris les codes de stockage fournis dans le répertoire pour les données et le "manquant",
 - les noms de VARIABLES DE SOUS-UNITE AFFECTES D'UN INDICE de sous-unité mis entre parenthèses. Par exemple, "A320(2)" signifie "variable quantité de A320 dans la sous-unité 2, c'est-à-dire en commande".

Dans ces conditions, l'EXPRESSION ELEMENTAIRE (et minimale) est de la forme:

$X \text{ op } Y$, ou

"op" est un opérateur de comparaison de a) et
 "X", "Y" sont des éléments de d),
 avec les possibilités suivantes (et leurs symétriques)
 (on note U pour "de l'unité", SU pour "de sous-unité",
 SUi et SUj pour des indices i et j, numéros de sous-unité):

Variable_U	op	Donnée_U
Variable_SU	op	Donnée_SU
Variable	op	Entier de Z
Variable_SU(SUi)	op	Variable_SU(SUj)
Variable_SU(SUi)	op	Entier de Z
Variable_SU(SUi)	op	Donnée_SU

Sur ces expressions peuvent porter les opérateurs

unaires ^ (non) et nul,
 binaires & (et) et ! (ou),

avec les règles de l'Algèbre de Boole que l'on rappelle en annexe p.20.

L'expression de sélection est saisie sans format.

Les blancs en nombre quelconque entre les différents ELEMENTS définis précédemment (en a, b, c, d) sont permis, ce qui peut faciliter la relecture.

Ce sont les blancs et les caractères spéciaux utilisés (">", "<", "=", "~", "&", "!", "(", ")") qui servent de séparateurs au système.

L'expression n'est pas limitée en longueur: elle peut tenir sur plusieurs lignes et peut être constituée de 100 expressions (conditions) élémentaires, ce qui ne constitue pas vraiment une limitation. Le POINT doit apparaître une et une seule fois pour indiquer au système la FIN de l'expression.

Il faut noter que:

- quand il y a une variable de sous-unité sans indice dans une expression, on a choisi que la condition est vraie si elle vraie pour une -au moins une - sous-unité.

exemple: "A320 > 0" équivaut à:

"A320(1) > 0 ! A320(2) > 0" et non pas à

"A320(1) > 0 & A320(2) > 0"

- dans les expressions de sélection, on utilise indifféremment les données sous une des formes synonymes ou leur code de stockage, entier relatif.

exemple: "SIEGE = MADRID" équivaut à:

S\$ = 29

SIEGE = 29

SIEGE=MAD

S\$ = MADRID etc.

- pour extraire les enregistrements dans lesquels une variable est présente, on peut se servir du code du manquant en écrivant: " ^variable = manquant " (ici le code est "-1")
 exemple: " ^C% = -1 " équivaut à
 " la variable CAPITAL % ETAT est présente " (donc a été enregistrée)

Dans le cas précis de l'exemple, quand la variable C% (CAPITAL % ETAT) est présente, elle est positive ou nulle, donc pour extraire les compagnies où C% est inférieur à 30, on peut aussi bien dire $C\% < 30$ mais n'est pas négatif, soit écrire
 " $C\% < 30 \ \& \ ^C\% < 0$ ", qui équivaut à " $C\% < 30 \ \& \ ^C\% = -1$ "

9. SYNOPSIS1 ET SYNOPSIS2, UTILITAIRES DU LOGICIEL

On a donné en annexe, pages 21 à 24, des extraits des sorties de ces programmes qui ont exploré la banque après les mises à jour.

Toutefois on peut les utiliser quel que soit l'état du fichier (par rapport aux mises à jour). Ils en donnent des images instantanées et des statistiques de quantité d'information enregistrée.

SYNOPSIS1

Ce programme donne un synopsis graphique compact de la quantité d'information collectée. Il affiche en parallèle -d'où son nom- les fréquences d'utilisation des variables. On peut noter que son utilisation est plus démonstrative sur un fichier plus volumineux que celui de notre exemple.

Présentation de la sortie

L'affichage se fait par tranches de 100 ENREGISTREMENTS, mis en ENTREES COLONNES d'un tableau rectangulaire dont les ENTREES LIGNES sont les VARIABLES des deux niveaux successifs et les éléments sont les FREQUENCES d'utilisation correspondantes. On CONDENSE au maximum l'information étant donné les conditions de l'environnement matériel (132 colonnes et 66 lignes par page imprimables en sortie du système, sur papier paravent):

- une colonne seulement correspond à une entrée colonne,
- une ligne à une entrée ligne.

Par conséquent, à l'intersection d'une ligne et d'une colonne, UNE SEULE POSITION de caractère d'impression pour noter la FREQUENCE. Sur la largeur de la feuille, on a mis 100 entrées colonnes représentées sur 100 colonnes. Les entrées lignes sont imprimées en continu (sur des feuilles adjacentes, si elles tiennent sur plus d'une feuille).

Pour insérer la sortie en annexe, on a fait artificiellement une coupure horizontale. Ici on a un seul tableau (de 95 lignes).

Dans le cas général où il y a plusieurs tableaux, on peut les afficher (au sens courant du terme) côte à côte parallèlement, pour obtenir un tableau unique dont les entrées lignes sont les variables et les entrées colonnes tous les enregistrements du fichier.

Le numéro d'enregistrement est affiché naturellement sur une seule colonne, donc verticalement. Pour faciliter la lecture, on n'affiche que la partie unité du nombre tant qu'elle seule change.

Exemple: le numéro de l'enregistrement 2401 sera imprimé

2
4
0
1

en entête de la colonne correspondante.

Les entrées colonnes sont affichées sur les bords supérieurs et inférieurs du tableau. Les entrées lignes sont identifiées par les noms abrégés des variables et leur numéro d'ordre (ici 1 à 9 pour l'unité, 1 à 86 pour la sous-unité). Les noms sont affichés sur les bords gauche et droit à l'extérieur du tableau.

CONTENU

1. A l'intersection d'une ligne et d'une colonne une fréquence non nulle est représentée, on l'a dit, par un caractère d'impression.

Si la variable (entrée ligne) a été utilisée dans l'enregistrement (entrée colonne), la fréquence "1" est représentée par un "." pour les variables du niveau unité.

Pour le niveau sous-unité, une variable peut être renseignée autant de fois qu'il y a de sous-unité donc autant de fois que la variable-clef est présente (a une donnée) donc la fréquence peut atteindre le nombre de sous-unités de l'application, aussi est-elle représentée par les chiffres, 1 à 9, puis les lettres A à Z, quand c'est le cas. Ici, on a au plus une fréquence égale à 2, quand la variable de sous-unité est présente dans les 2 sous-unités de l'enregistrement (cas où la fréquence d'utilisation de la clef "F" est 2).

Exemples:

Dans les colonnes 54 à 100, la variable C manque. Comme il s'agit de la clef, cela permet de voir immédiatement où des enregistrements manquent, ce n'est pas en général évident comme ici.

Pour les colonnes 9 et 10, la variable E (EMPLOYES) manque.

Pour l'enregistrement numéro 13, la fréquence de la variable-clef F est 2: on a deux sous-unités. On voit immédiatement quels appareils sont présents dans cet enregistrement, en examinant toute la colonne.

2. Pour chaque tranche de 100 enregistrement, sur la même feuille, on a, par ligne (variable), la fréquence cumulée imprimée sur la ligne à l'extérieur du tableau, à droite du nom de variable.

Exemples: sur la ligne de la variable C (la clef), la fréquence est 53, (nos 53 enregistrements "compagnies"), alors que pour la variable E (nombre d'employés), la fréquence retombe à 47. Donc seulement 47 compagnies ont fourni ce renseignement et on voit précisément lesquelles à l'intérieur du tableau là où il y a des blanc. Pour la variable C% (CAPITAL % ETAT), la fréquence n'est que de 31. On revoit aussi les variables fictives (fréquence nulle) insérées dans le répertoire en vue de réactualiser la flotte, EMB, C50 etc..

Par colonne, on imprime aussi à l'extérieur inférieur du tableau le nombre de données enregistrées qui se lit évidemment verticalement.

3. La dernière tranche demandée par l'utilisateur affiche en plus sur la même feuille, par ligne, à droite de la fréquence cumulée sur ces 100 enregistrements, celle cumulée sur tous les enregistrements affichés dans les tableaux précédents, qui peut être la fréquence totale. (Dans notre cas, comme il n'y a qu'un tableau, car moins de 101 enregistrements, cela n'apporte aucune information supplémentaire)..

Enfin, en bas de ce dernier tableau, le nombre total de données enregistrées est imprimé précédé de "STT=".

INTERET IMMEDIAT

On peut voir, D'UN COUP D'OEIL, les "TROUS" du fichier : où des données manquent et si des variables ne sont pas utilisées du tout, fictives ou non. On peut repérer D'UN COUP D'OEIL des INCOHERENCES: si des variables doivent être logiquement présentes (renseignées) en même temps, elles auront la même fréquence, si des variables doivent être présentes dans toutes les sous-unités, leur fréquence doit être égale à celle de la clef de sous-unité, etc. Cela permet donc de sérier les questions à poser à la banque afin d'obtenir la constitution d'un fichier fiable en employant les mises à jour.

SYNOPSIS2

Ce programme calcule les fréquences d'utilisation de chaque DONNEE et les récapitule par variable.

Le résultat est une liste imprimée dont on donne, en p. 22 et 23 de l'annexe des extraits.

On y trouve, POUR CHAQUE VARIABLE SUCCESSIVEMENT (ordre du répertoire), y compris pour les clefs d'accès à l'unité et à la sous-unité, d'abord et dans tous les cas le NOM de cette variable sous la forme non abrégée.

Exemple: "EMPLOYES" (p.23)

Puis DEUX CAS se présentent:

1. Cette variable n'a PAS été UTILISEE ("de façon valide"), alors on passe tout de suite à la suivante.

Exemple: "EMB120" (p.23)

2. Cette variable a été UTILISEE, on a deux cas:

C'est une QUANTITATIVE. Alors les MINIMUM et MAXIMUM REELLEMENT ATTEINTS sont imprimés sur deux lignes successives précédés respectivement de "MIN" et "MAX".

Exemple: pour la variable EMPLOYES (p.23), le minimum réel est de 150, le maximum réel est de 35232, et il y a 47 enregistrements renseignés sur cette variable.

C'est une QUALITATIVE. Alors on a la LISTE DES SEULES DONNEES qui ont été UTILISEES. Chaque donnée, si elle a été utilisée au moins une fois, est imprimée sur une ligne, avec, à sa gauche, le code entier de stockage, et, à sa droite sa FREQUENCE. Quand elle a des synonymes, l'information est répétée pour chaque synonyme.

Exemple: pour la variable SIEGE, la donnée BELGRADE (code de stockage 37, synonyme BEG) a une fréquence égale à 2: deux compagnies ont leur siège à Belgrade.

Puis est récapitulée la FREQUENCE TOTALE d'utilisation de la variable. Ce résultat est aussi donné dans SYNOPSIS1, en dernière colonne du tout dernier tableau demandé.

INTERET

Dans cette sortie, si toutes les variables sont rappelées, même non utilisées, par contre n'apparaissent QUE LES DONNEES qualitatives REELLEMENT UTILISEES, avec leur fréquence, et les MINIMUM et MAXIMUM REELLEMENT ATTEINTS par les quantitatives.

L'analyse de cette sortie peut conduire par exemple à alléger le répertoire de données qualitatives superflues (on sait qu'on peut les réinsérer si le besoin s'en fait sentir plus tard) et d'y resserrer la fourchette permettant de valider plus strictement les quantitatives.

Autre exemple: si, pour une application, deux DONNEES qualitatives particulières de deux variables doivent être logiquement associées, on voit immédiatement si leur fréquence est égale.

C'est l'observateur connaissant le sujet traité qui sera à même d'exploiter au mieux ces informations.

L'association de ce traitement avec les interrogations permet des mises à jour efficaces.

9. ALTERNATIVE A LA STRUCTURE D'ENREGISTREMENT RETENUE

Au paragraphe 2 on a vu les informations dont on parlait.

On aurait pu choisir une autre structure pour les données, c'est à dire se définir autrement la sous-unité.

Si les informations générales sur la compagnie ne peuvent que rester naturellement au niveau d'entrée dans l'enregistrement, niveau unité, le sous-ensemble des informations sur la flotte peut être subdivisé autrement.

En effet, on peut considérer un type d'avion présent dans la compagnie comme une sous-unité pour laquelle il y aura les deux variables "quantité en service" et "quantité commandée". On identifie chaque type d'avion présent, sous-unité, par une clef, de nom "A" par exemple.

Dans ces conditions, il y a autant de sous-unités pour une compagnie donnée que de types d'appareils différents constituant sa flotte. Et ce nombre varie d'une compagnie à l'autre. IL faut prévoir un maximum de 48 sous-unités au moins.

Dans cette sous-unité, on a alors, outre la clef, trois variables pour prendre en compte les informations et on les décrira dans le répertoire:

- 1) la clef d'accès déjà citée, quantitative entière, "A",
- 2) le nom du type d'avion, qualitative, (constructeur+type+série), variable "AVION", par exemple.

On y met comme données les 48 types présents recensés par le programme Synopsis1 dans la structure précédemment traitée. On pourrait même aller jusqu'à la série qui est parfois fournie dans les données d'origine.

- 3) le nombre en service, quantitative, variable "S", minimum 1, maximum 25 (cf sortie de synopsis1 donnant directement le chiffre).

- 4) le nombre en commande, quantitative, variable "C", minimum 1, maximum 25.

Avec cette structure, les informations concernant la flotte s'exprimeraient de la façon suivante, à l'entrée, par exemple pour la compagnie CIPRUS AIRWAYS,

A=1. A310. S=3. (3 A310 en service)
 A=2. A320. C=4. (4 A320 commandés)
 A=3. BAC1-11. S=3.
 A=4. B707. S=3.

Pour la compagnie AIR FRANCE,

A=1. A300. S=18.

A=2. A310. S=5. C=1.

A=3. A320. S=25.

A=4. CONCORDE. S=7.

A=5. B727. S=29.

A=6. B737. S=12.

A=7. B747. S=31. C=1.

L'interrogation qui donnera les compagnies ayant des A320 en service ou en commande s'écrira:

AVION = A320.

Mais on voit qu'on ne peut plus extraire de façon immédiate et séparée les compagnies ayant les A320 en service, par exemple, puisque le numéro de la sous-unité concernée n'est pas le même d'une compagnie à l'autre.

Pour la saisie, la liste des types d'avion (variable qualitative "AVION") sera facilement mise à jour en l'allongeant dans le répertoire, à condition d'avoir prévu un nombre de sous-unités suffisant, alors qu'avec la structure qu'on a choisie pour notre traitement, en vue des mises à jour, on a mis au départ dans le répertoire des variables virtuelles quantitatives de types d'avion (elles ont été intercalées pour rester dans l'ordre des pays constructeurs), exemple "MYSTERE10".

On voit que le choix de la structure décrite dans le répertoire a une influence sur les accès ultérieurs aux données et, réciproquement, que c'est en fonction de la façon dont on veut pouvoir y accéder qu'on choisira la structure.

III COMPLEMENTS

1. AUTRE PROPRIETE DE SAISIE-VALIDATION

On a vu que le mode d'entrée "local" ou "global" et la clef d'accès à l'unité étant donnés, on peut entrer les données du niveau unité dans n'importe quel ordre, et que, si la clef de sous-unité est donnée, celles de la sous-unité peuvent également être saisies dans n'importe quel ordre.

Mais il y a une autre propriété importante, dont l'intérêt est renforcé par son association avec la précédente et valable à chaque niveau: si on entre, à un niveau donné, plusieurs données "acceptables" (correctes) pour une même variable, c'est la dernière entrée que le système garde et enregistre (et il n'annote pas les précédentes).

Exemples: le mode de saisie ayant été donné, si on entre

"C=3. P=1307. P%=65. P=300.",

c'est "P=300" qui sera enregistré (pas 1307)

puis, si on entre

"F=1. B727=4. B737=8. B727=10.,

c'est "B727=10" qui sera enregistré (pas 4)

D'un point de vue pratique, ceci peut être intéressant sur le terrain: si, en entrant ses données, on s'aperçoit d'une erreur ou d'un oubli à l'intérieur du niveau d'enregistrement que l'on est en train de saisir, il n'y a pas besoin de remonter ou reculer sur l'écran pour effacer ou insérer une donnée: on entre alors la donnée correcte. Mais si on a quitté l'unité ou la sous-unité, on y retourne en mode local, comme on l'explique ci-dessous.

On donne en annexe (p.25) un résumé des propriétés de saisie-validation sur un exemple imagé.

2. MISE A JOUR DU REPERTOIRE - REPERTOIRES EQUIVALENTS

MISE A JOUR DU REPERTOIRE

Quand une donnée est annotée sans qu'il y ait une erreur de saisie (directement de la donnée ou au niveau de sa définition, mode et clefs), on met à jour le répertoire de façon à ce qu'elle soit acceptée quand on la proposera à nouveau.

On peut mettre à jour le répertoire en modifiant la fourchette de validité d'une quantitative (cas de l'exemple) ou en allongeant les listes des qualitatives, par synonymie ou non.

Dans l'exemple, après mise à jour du répertoire, ces données ont été entrées en mode "local", donc sans toucher aux autres.

Toutefois, il y a l'autre solution de tout recharger globalement comme à la première étape, avec, localement, les corrections dues aux seules erreurs de saisie. Cette possibilité est intéressante quand il s'agit de GROS FICHIERS et que la donnée qu'on vient d'ajouter au répertoire y est fréquente.

REPERTOIRES EQUIVALENTS

Au lieu d'insérer les synonymes, par exemple réservés à la saisie, dans le répertoire, on peut créer un autre répertoire de même structure (mêmes variables aux deux niveaux) ne comportant que les termes à utiliser en saisie, et auquel on se référera uniquement pour cet usage. On peut aussi en créer un autre ne comportant, par exemple, que les termes convenant à des publications, et auquel on fera appel pour cet usage, etc., chacun est plus léger et on peut aussi par ce moyen donner d'autres synonymes pour les quantitatives, sachant qu'on ne peut en fournir que deux par répertoire.

Ce qui est nécessaire, c'est que les mêmes variables soient aux mêmes endroits, représentant les mêmes entités, et que les codes archivés réellement dans la banque puissent avoir une TRADUCTION dans n'importe lequel des répertoires équivalents, le type (qualitatif ou quantitatif) de la variable important peu.

3. PRINCIPES DE L'ACCES AUX DONNEES - MODES

Les entrées fractionnées en modes différents sont totalement admises.

On peut revenir à tout moment, soit au cours du traitement, soit n'importe quand plus tard, sur un enregistrement pour ponctuellement (en mode local) lui fournir des données qui n'étaient pas encore enregistrées, pour en remplacer ou pour en supprimer quand elles ne sont pas simplement à remplacer.

Il est important de se souvenir qu'on peut:

- Effacer complètement un enregistrement (le vider de toutes les valeurs qui pouvaient y être alors) à condition d'y accéder en mode global. Il est prêt à recevoir les (nouvelles) données.

- Ponctuellement placer, ou remplacer ou effacer une ou des données d'un enregistrement, à condition d'y accéder en mode local.

En résumé, on peut dire qu'on accède directement aux enregistrements que l'on veut, en nombre que l'on veut, dans l'ordre que l'on veut, dans les modes d'accès que l'on change successivement comme on veut, et dans ces enregistrements, directement aux données que l'on veut (propriétés de SGBD).

Chaque donnée (au sens du logiciel) valide est entièrement déterminée (définie) par:

- le mode d'accès (global ou local),
- la clef d'accès à l'unité, si c'est une donnée du premier niveau (le numéro),
- la clef d'accès à la sous-unité, si c'est une donnée du deuxième niveau (le numéro)

La hiérarchie des définitions se fait comme suit:

- la donnée du mot-clef "GLOBAL" ou "LOCAL", entre comme n'importe quelle donnée met instantanément dans le mode en question. Par défaut, on y reste tant qu'on n'en fournit pas un autre. C'est une règle importante.
- La donnée de la clef d'accès à l'enregistrement, (unité d'enregistrement), le mode étant défini, met instantanément dans l'enregistrement en question. Par défaut, on y reste tant qu'on ne fournit pas une autre clef ou un autre mode.
- Enfin, la donnée de la clef d'accès à une sous-unité, le mode et l'enregistrement étant définis, met instantanément dans la sous-unité en question. Par défaut, on y reste tant qu'on ne fournit pas une autre clef de sous-unité, une autre clef d'unité ou un autre mode.

Ces règles doivent rester présentes à l'esprit. On est toujours implicitement dans la portée des clefs qui restent actives tant qu'on n'en fournit pas explicitement d'autres.

Dans le niveau de définition (unité ou sous-unité) où on est, le système va valider une donnée de la façon suivante:

- si la donnée contient un signe "=", il recherche la chaîne de caractères précédant ce signe dans les noms de quantitatives du répertoire, et, s'il le trouve, vérifie que l'entier après le signe "=" est dans la fourchette de validité correspondante.
- si la donnée ne contient pas de signe "=", il recherche la chaîne dans les données des qualitatives.

IV ENVIRONNEMENT D'UNE APPLICATION

Quand on veut constituer une banque de données sous NEPTUNE, on commence donc par constituer le répertoire, ce qui fixe la structure logique des données. On pourra ensuite paramétrer les programmes et initialiser la banque.

Le répertoire initial peut n'être que très léger et ne comporter qu'une sorte d'"ossature" des données, ceci pour les applications où la terminologie est en évolution.

CE QUI DOIT ETRE FIXE DES CE MOMENT-LA, ce sont le nombre de variables de l'unité (clef comprise), le nombre de variables de la sous-unité (clef comprise) et le nombre maximum de sous-unités envisagées. C'est ce qui ne devra plus changer par la suite.

A l'inverse, les fourchettes de validité des quantitatives peuvent être modifiées (dans les deux sens), et des données, en synonymie ou non, peuvent être ajoutées pour les qualitatives, n'importe quand plus tard. En ce sens, le référentiel est "OUVERT".

Par conséquent, si, au démarrage, on envisage la possibilité d'enregistrer dans le futur des variables pour lesquelles on n'a pas encore d'information, on réserve leur place dans le répertoire, en fournissant un nom et une donnée (minimum nécessaire et suffisant), même fictifs.

On mettra à jour le répertoire au fur et à mesure qu'on sera en mesure de préciser ces variables, puis la mise à jour des données se fera simplement, en mode "local", en fournissant seulement les données concernées, introduites par la (les) clef(s) adéquates. Elles viendront automatiquement compléter correctement l'enregistrement.

Dans l'exemple, on a mis des variables types d'appareils pour lesquels on n'avait pas de données, avec des minimum et maximum fictifs, pour des mises à jour ultérieures du fichier.

"COMMENTAIRES" ENREGISTRES

On n'a pas parlé du texte libre qui peut faire partie intégrante de l'enregistrement - des "commentaires" -. Ils apparaissent toujours dans les éditions.

Ce sont des informations que l'on veut garder, mais que l'on ne peut pas ou que l'on ne veut pas répertorier (on ne peut pas interroger la banque sur les commentaires).

A la saisie, pour que le système les reconnaisse comme tels (sans avoir à chercher s'ils sont répertoriés), on les fait précéder d'un caractère spécial que l'on réserve à cet usage, et on marque la fin par un point. On entre ceux du niveau unité après la clef d'accès, et ceux relatifs à une sous-unité qu'après y avoir accédé avec sa clef. A l'intérieur des niveaux qu'ils commentent ils peuvent être saisis intercalés avec les données. Le système se charge de les concaténer. L'ouverture avec la clef en mode global les expulse avec le reste. En mode local, l'apport d'un commentaire l'ajoute à ceux qui se trouvaient déjà éventuellement dans l'enregistrement, caractéristique propre à ce type d'objets.

Au moment du paramétrage des programmes avec les nombres de variables de l'unité et de la sous-unité, on donne le nombre de caractères (la longueur de la chaîne) de commentaires maximum qu'on veut enregistrer au niveau unité et (autant de fois qu'il y a de sous-unités prévues) au niveau sous-unité.

CHOIX DES TERMES DU REPERTOIRE

On peut donner quelques conseils à ce sujet, à la lumière de l'expérience de saisie.

On choisit le nom de la variable (les noms) à la fois aussi clair et concis que possible: cela facilite la saisie et la relecture, puisque le nom intervient dans la saisie des données quantitatives et dans les interrogations.

Quant aux données qualitatives, on a intérêt à ce que leur expression soit suffisamment "CONCISE" mais "PARLANTE" en saisie et "EXPLICITE" en édition, au moins pour le synonyme prioritaire qui est édité, sans pour autant que sa lecture répétée devienne fatigante.

Il faut donc trouver un juste équilibre entre des abréviations que l'on peut retenir facilement et les termes qu'on a intérêt à laisser le plus clairs possible (pour ne pas retomber dans les inconvénients des codages, puisque c'est l'ordinateur qui s'en charge après avoir vérifié que ce que l'on rentre est correct). Une CONCISION EXPLICITE est souhaitable.

UNICITE DES TERMES DU REPERTOIRE

Quand on veut mettre à jour le répertoire en allongeant les listes des données ou en remplaçant par de "vraies" variables les variables "de réserve", il faut préserver l'unicité totale des éléments du répertoire: données des qualitatives et noms de toutes les variables.

On le conçoit aisément du fait que l'on ne saisisse que les données observées, sans format, que, de plus, on puisse les noter dans n'importe quel ordre, et qu'enfin on ne saisisse pas le nom de la variable pour les qualitatives. Le système doit, malgré cela, pouvoir IDENTIFIER SANS AMBIGUITE une donnée.

De toute façon, il est appréciable que deux informations différentes correspondent à deux expressions différentes: cela conduit à une terminologie descriptive RIGOUREUSE sémantiquement.

V INFORMATIONS SUR L'EXPLOITATION ACTUELLE DU SYSTEME

Le système est actuellement exploité dans l'étude de paysages du Sénégal.

Les données sont saisies sur le terrain avec un EPSON HX20.

La structure de l'enregistrement est celle qui a été mise au point lors de l'étude des paysages du Bénin, base de la thèse présentée par C.J. Houndagba à l'Université de Strasbourg en 1984. En effet la méthode d'appréhension du milieu naturel est la même. On y retrouve les concepts de diagnostics typologiques développés par J.F. Richard dans sa thèse (Paris, 1985) et qui ont été adoptés par des chercheurs de différentes universités africaines.

Les 3804 octets de l'enregistrement se répartissent comme suit:

- 300 variables (dont un certain nombre encore en réserve) et
- 800 caractères de commentaires autorisés pour le niveau unité,
- 20 sous-unités avec chacune
- 35 variables et
- 20 caractères de commentaires autorisés.

Le répertoire est mis à jour progressivement, au fur et à mesure que se constitue le fichier, avec des données dont certaines sont propres au Sénégal.

Ce système est particulièrement bien adapté à cette recherche en évolution, où l'on peut proposer la création (l'"invention") de nouveaux diagnostics qui peuvent venir compléter les listes.

Le vocabulaire est tel (il y a tellement de termes possibles) qu'on a dû éliminer l'utilisation de sortes de "menus", surtout qu'on saisit sur le terrain et sur portatif. La souplesse des règles de saisie de NEPTUNE (pas de format, modes alternés à volonté, accès direct, ordre quelconque dans un niveau, données en nombre quelconque) est intervenue dans cette décision.

Les deux niveaux permettent de traiter les strates perçues en un point où le chercheur fait un relevé (strates de végétation dans l'atmosphère et couches dans le sol), avec, pour chaque strate, les éléments diagnostiques qu'il y reconnaît (jusqu'à 20 par strate).

VI CARACTERISTIQUES TECHNIQUES DU SYSTEME

1. CARACTERISTIQUES MATERIELLES

Il a été mis au point sur l'ordinateur-hôte du CNRS-CIRCE d'architecture IBM 370 sur lequel il est implanté et qui est actuellement un biprocesseur NAS 9080 couplé avec un biprocesseur IBM 3090/200 sous MVS/XA + JES3.

On y accède en batch, via le Bull Mini6 du Centre Orstom de Bondy, terminal du CIRCE, sur lequel les procédures sont stockées, tandis que les fichiers NEPTUNE et les programmes en binaire sont sur le site central.

Les programmes ont été écrits en PL1 et compilés avec PL1 optimiseur.

Les données sont dans un fichier séquentiel indexé VSAM d'organisation KSDS (initialement le système a été mis au point pour utiliser des fichiers ISAM). Les programmes de sauvegarde séquentielle et restauration en VSAM à partir de la sauvegarde font partie du logiciel, les fichiers VSAM n'étant pas actuellement gérés par HSM.

On accède aux données
 en accès direct pour les stockages-mises à jour et les éditions ponctuelles,
 en accès séquentiel pour les éditions in-extenso et les interrogations/extractions.
 Les répertoires sont des fichiers séquentiels. Leur longueur de blocs ne leur permet pas
 non plus d'être gérés par HSM.

C'est un système mono-utilisateur: une seule personne à la fois peut accéder aux
 fichiers à un instant donné.

Pour la sélection/interrogation on a utilisé la notation polonaise ou post-fixée pour
 traiter l'expression de sélection.

2. CARACTERES AUTORISES POUR CONSTITUER LES REPERTOIRES

Les caractères autorisés pour constituer les répertoires sont ceux du PL1. Ce sont:

les 26 lettres majuscules A à Z

les 10 chiffres 0 à 9

£

\$

%

*

à

' (apostrophe)

, (virgule)

:

/

;

-

+

_ (souligné)

et le caractère blanc (espacement) qu'on utilise ordinairement à raison d'un seul
 pour séparer deux mots dans les noms de variables et les données qualitatives: il joue le
 même rôle que n'importe quel caractère à l'intérieur des termes du répertoire.

Parmi les caractères spéciaux qu'on vient d'énumérer, on en choisit deux à qui on
 attribue un rôle particulier en saisie-validation: l'un introduira les "commentaires" (cf
 chapitre IV) et l'autre une donnée à supprimer (cas qui se présente si elle n'est pas
 simplement à remplacer). Aussi les deux caractères choisis et indiqués au système au
 début du répertoire ne seront-ils plus utilisables pour en constituer les termes.

Les caractères suivants ne sont pas non plus utilisables car ils sont réservés au système
 pour la saisie-validation:

. évidemment puisqu'il marque la fin de la donnée,

= partie intégrante d'une donnée quantitative,

? puisqu'il remplace le point final d'une donnée annotée

Sont aussi réservés au système pour écrire les expressions de sélection:

< signe "inférieur à"

> "supérieur à"

! pour "OU"

& "ET"

^ "NON"

et les parenthèses.

Remarque: tous ces caractères sont cependant autorisés à l'intérieur des commentaires.

3. TEMPS DE CALCUL

Voici quelques temps de calcul dans l'exploitation actuelle du système sur les paysages du Sénégal (temps arrondis à la seconde supérieure):

- Il a fallu moins de 41 secondes d'unité centrale NAS9080 pour traiter les 3285 images-cartes obtenues par transfert à partir du HX20 des 262800 caractères saisis au clavier sur ce matériel et enregistrer 19439 données valides dans 741 enregistrements.

On se souvient que, dans cette application, l'enregistrement occupe 3804 octets: au niveau unité, il y a 300 variables et 800 caractères de commentaires, et il y a 20 sous-unités ayant chacune 35 variables et 20 caractères de commentaires.

- Pour charger un répertoire contenant 2000 données qualitatives et 35000 caractères de textes (compactés), il faut 7 secondes d'unité centrale.

- Pour éditer les 741 enregistrements, il faut moins de 4 secondes d'UC.

- Pour traiter le fichier avec le programme Synosis1, donnant la fréquence de chaque variable par enregistrement, puis par tranches de cent enregistrements, puis cumulée sur le fichier entier, il faut 3 secondes d'UC.

- Enfin la fréquence d'utilisation de chaque donnée qualitative et les maximums et minimums réellement atteints par les quantitatives (programme Synopsi2) prend 4 secondes d'UC.

Ces temps d'UC sont facturés au tarif CNRS. A ces temps s'ajoute la location des pistes de disques magnétiques sur lesquels sont enregistrés les programmes (binaire exécutable), les répertoires et la banque de donnée elle-même.

Les programmes et le fichier des données sauvegardé en séquentiel sont sur des disques gérés par le système HSM qui offre un tarif dégressif

Donc le coût dépend du stockage et du temps d'UC de chaque application.

VII CONCLUSION

Un tel système est un outil de recherche présentant divers avantages pour les sciences de l'environnement:

- Il s'agit d'abord d'une AIDE A LA CONSTITUTION D'UNE TERMINOLOGIE descriptive pour structures arborescentes à deux niveaux. Il permet de constituer cette terminologie PAR APPROCHES SUCCESSIVES (mises à jour des répertoires associées à celles des données), régies par des choix RIGOUREUX (unicité des termes).

- Il aide à l'APPRENTISSAGE de cette terminologie dans le cadre de formation et d'enseignement (annotation et rejet de ce qui n'y est pas conforme).

- Il présente en outre une certaine CONVIVIALITE grâce

- . à sa facilité d'utilisation sur le terrain due à la souplesse de la saisie, sans codes et sans format qu'on a détaillée dans le texte,

- . aux mises à jour locales, d'une donnée, pour une unité et sous-unité identifiées, en rentrant simplement la donnée.

- . au fractionnement à volonté de la saisie-stockage, avec les modes de saisie global/local alternés.

- . aux interrogations faciles à exprimer et aux extractions rapides

- . à l'absence générale de langage de manipulation, de format, de formalisme et à l'emploi d'un minimum de paramètres dans les procédures.

D'autre part il est reconnu que, dans un logiciel, les optimums de facilité d'utilisation et de fiabilité peuvent difficilement être atteints en même temps et qu'on doit faire un compromis.

Ici, la souplesse de saisie se trouve "encadrée" par la seule règle que, par défaut, le mode et la clef à chaque niveau restent actifs tant qu'on n'en introduit pas d'autres, sachant que toute donnée doit être "définie" pour le système (il doit savoir dans quel mode la traiter et avec quelle(s) clef(s) la localiser).

Il faut donc y prendre garde et cela demande le minimum de rigueur qui va de pair avec l'utilisation de l'informatique.

Le système a enfin l'avantage d'être opérationnel sur le Centre de Calcul du Circe(CNRS) à Orsay où l'on trouve un souci constant de la qualité du service offert aux utilisateurs, tant du point de vue matériel que logiciel. Une permanence d'assistance très compétente répond aux problèmes qu'on lui pose sur place ou par téléphone. Les logiciels statistiques en ligne permettent aussi de traiter les fichiers de données séquentiels obtenus à partir de la banque.

VIII REFERENCES BIBLIOGRAPHIQUES

INTERAVIA 10/85

AUBRY Anne-Marie (et al.) - 1984 - From POSEIDON to NEPTUNE : Software for environmental surveys. in Burrough P.A. and Bie S.W. (Eds) Soil Information Systems Technology, Bolkesjo, pp. 41-45

HOUNDAGBA Cossi-Jean - 1984 - Analyse typologique informatisée des paysages d'Abomey-Zagnanado (R.P. du Bénin). Exploitation d'un système de programmes PL1 NEPTUNE. Thèse de 3e cycle, Université Louis Pasteur, Strasbourg, 286 p.

Richard Jean-François - 1985 - Le paysage, analyse et synthèse. Contribution méthodologique à l'étude des milieux tropicaux (savanes et forêts de Côte d'Ivoire). Université PARIS VII (thèse d'Etat) 438 p.

IX ANNEXES

Contenu de la nomenclature (ou répertoire)	pp. 1-5
Données: image de l'entrée	pp. 6-7
Saisie-validation en mode global	pp. 8-9
Edition	pp. 10-15
Mise à jour en mode local :image de l'entrée et sortie imprimée	p.16
Edition des enregistrements mis à jour	p.17
Interrogations/extractions	p.18
Edition d'un fichier extrait	p.19
Règles de l'Algèbre de Boole	p.20
Sortie de Synopsis1	pp. 21-22
Sortie de Synopsis2 (extraits)	pp.23-24
Mises à jour globales et locales: résumé image	p.25-27
Programmes sources en PL1	pp.28-51

NOMENCLATURE

archivé

saisi

(synonymes)

1	AIR ALGERIE	AH
2	EGYPTAIR	MS
3	LIBYAN ARAB AIRLINES	LN
4	ROYAL AIR MAROC	AT
5	TUNIS AIR	TU
6	AIR GUINEE	GI
7	LAM	TM
8	AIR TANZANIE	TC
9	AIR ZAIRE	QC
10	AIR BOTSWANA	BP
11	AIR ZIMBABWE	WX
12	AIR MADAGASCAR	MD
13	AIR MAURITIUS	MK
14	ETHIOPIAN AIRLINES	ET
15	CAMEROON AIRLINES	UY
16	AIR MALAWI	QM
17	GHANA AIRWAYS	GH
18	NIGERIA AIRWAYS	WT
19	ROYAL SWAZI NATIONAL AIRWAYS	ZC
20	SIERRA LEONE AIRLINES	LJ
21	ZAMBIA AIRWAYS	QZ
22	TAAG	DT
23	KENYA AIRWAYS	KQ
24	AIR AFRIQUE	RK
25	AIR IVOIRE	VU
26	UTA	UT
27	AIR FRANCE	AF
28	MINERVE	MM
29	IBERIA	IB
30	AVIACO	AO
31	SPANTAX	BX
32	AIR MALTA	KM
33	ALITALIA	AZ
34	OLYMPIC ou OLYMPIC AIRWAYS	OA
35	CYPRUS AIRWAYS	CY
36	THY	TK
37	JAT	JU
38	INEX ou INEX ADRIA AIRWAYS	JP
39	AVIOGENEX	JJ
40	BALKAN	LZ
41	AIR PORTUGAL ou TAP	TP
42	TAROM	RO
43	MALEV	MA
44	CSA	OK
45	AUSTRIAN ou AUSTRIAN AIRWAYS	OS
46	ALIA	RJ
47	GULF AIR	GF
48	IRAQI AIRWAYS	IA
49	IRANAIR ou IRAN AIR	IR
50	MIDDLE EAST AIRLINES ou MEA	ME
51	SAUDIA	SV
52	KUWAIT AIRWAYS	KU
53	SYRIAN ARAB AIRLINES	RB

codes internationaux

manquants archivés -1

clé C= restant C min 1 max 53

variable COMPAGNIE abrégée C\$

archivé

saisi

(synonymes)

1	ALGER	ALG
2	LE CAIRE	CAI
3	TRIPOLI	TIP
4	CASABLANCA	CAS
5	TUNIS	TUN
6	CONAKRY	CKY
7	MAPUTO	MPM
8	DAR ES SALAM	DAR
9	KINSHASA	FIH
10	GABORONE	
11	HARARE	
12	ANTANANARIVO	TNR
13	PORT LOUIS	MRU
14	ADDIS ABEBA	ADD
15	DOUALA	DLA
16	BLANTYRE	BLZ
17	ACCRA	ACC
18	LAGOS	LOS
19	MANZINI	MTS
20	FREETOWN	FNA
21	LUSAKA	LUN
22	LUANDA	LAD
23	NAIROBI	NBO
24	ABIDJAN	ABJ
26	PARIS	PAR
29	MADRID	MAD
32	MALTE	MLA
33	ROME	ROM
34	ATHENES	ATH
35	NICOSIE	NIC
36	ISTANBUL	IST
37	BELGRADE	BEG
38	LJUBLJANA	LJU
40	SOFIA	SOF
41	LISBONNE	LIS
42	BUCAREST	BUH
43	BUDAPEST	BUD
44	PRAGUE	PRG
45	VIENNE	VIE
46	AMMAN	AMM
47	BAHREIN	BAH
48	BAGDAD	BGW
49	TEHERAN	THR
50	BEYROUTH	BEY
51	DJEDDAH	JED
52	KOWEIT	KWI
53	DAMAS	DAM

codes internationaux

```
variable EMPLOYES=          abrégée E=
                                min 100 à max 36000
```

variable CAPITAL % ETAT= abrégée C%=
min 0 à max 100

— 1 —

variable	RESEAU	abrégée R\$
	codes archivés	chaînes à saisir
2		I EN AF
5		I EN EU
10		I AF,EU
12		I AF,PO
14		I AM,EU
21		I EU,PO
24		I AF,AM,EU
27		I AF,AS,EU
31		AF,EU,PO
37		AS,EU,PO
47		AF,AM,EU,PO
50		AF,AS,EU,PO
54		AM,AS,EU,PO
58		AF,AM,AS,EU,PA
59		AF,AM,AS,EU,PO
63		I AM,AS,EU,PA,PO
64		I AF,AM,AS,EU,PA,PO ou MONDIAL

variable PASSAGERS EN 10**3= abrégée P= min 10 à max 15000

variable OCCUPATION DES SIEGES P%= abrégée P=

min 30 à max 80

variable PASSAGERS KM PAYANTS EN 10**6= abrégée PKM=
min 10 à max 30000

clé F=	restant F=	min 1 et max 2
		(1 pour achat ou location; 2 pour commande)
variable	EMB120=	abrégée EMB min 1 à max 5
variable	CL50=	C50 5
variable	CL215=	C215 10
etc	liste à la page suivante	
variable	MD87=	MD87 5

CL50	5
CL215	10
CL600	5
CL601	5
DHC5	5
DHC6	5
DASH7	5
DASH8	5
TWIN OTTER	15
LET410	5
LET610	5
Y7	5
Y12	5
DO228	15
CASA212	15
CN235	15
A300	15
A310	25
A320	15
MYSTERE10	35
MYSTERE20	5
MYSTERE50	5

CARAVELLE	5
MERCURE	5
NORD262	5
CONCORDE	7
ATR42	15
ATR72	15
SF600	15
F27	20
F28	10
F50	5
F100	5
ROMBAC1-11	10
ROMBAC141	5
AN12	5
AN24	40
AN26	10
AN30	5
AN400	5
IL18	15
IL62	15
IL76	20
IL86	5
TU134	20
TU154	20
YAK40	20
YAK42	20
SF340	5
BAC1-11	15
BAC125	5
BAE146	5
BAE748	10
JETSTREAM31	5
SHORTS	10
TRIDENT	10
VISCOUNT	10
BEECH99	5
BEECH1900	5
HS748	10
BN TRISLANDER	5

(suite)

B707	10
B720	10
B727	40
B737	25
B747	35
B757	5
B767	5
CV580	5
CV990	5
DC3	10
DC8	5
DC9	30
DC10	5
L100	5
L382	5
L410	5
L1011	20
MD80	15
MD81	15
MD82	5
MD83	5
MD87	5

(max)

(min=1 et non 0)

Les mêmes variables, mais abrégées (cf. p. 3 in fine)

C50
C215
C600
C601
DHC5
DHC6
DASH
DSH8
TWIN
V410
V610
Y7
Y12
D228
C212
C235
A300
A310
A320
FA10
FA20
FA50

CRV
MERC
N262
CONC
ATR4
ATR7
S600
F27
F28
F50
F100
R111
R141
AN12
AN24
AN26
AN30
V400
IL18
IL62
IL76
IL86
T134
T154
YA40
YA42
S340
B111
B125
B146
B748
JS31
SHOR
TRID
VISC
BE99
BE19
H748
BNTR

B707
B720
B727
B737
B747
B757
B767
C580
C990
DC3
DC8
DC9
DC10
L100
L382
L410
3STA
MD80
MD81
MD82
MD83
MD87

S -1 : Ligne n DU REPERTOIRE, MANQUANT, SUPPRESSION, COMMENT,
UNIT

C C = C
1 MIN
53 MAX
C\$ COMPAGNIE
1 AIR ALGERIE
1 AH
2 EGYPTAIR
2 HS

...

50 BEYROUTH
50 BEY
51 DJEDDAH
51 JED
52 KOWEIT
52 KWI
53 DAMAS
53 DAM
C E = EMPLOYES
100 MIN
36000 MAX
C% = CAPITAL % ETAT
0 MIN
100 MAX
R\$ RESEAU
2 EN AF
5 EN EU
10 AF,EU
12 AF,PO
14 AM,EU
21 EU,PO
24 AF,AM,EU
27 AF,AS,EU
31 AF,EU,PO
37 AS,EU,PO
47 AF,AM,EU,PO
50 AF,AS,EU,PO
54 AM,AS,EU,PO
58 AF,AM,AS,EU,PA
59 AF,AM,AS,EU,PO
63 AM,AS,EU,PA,PO
64 AF,AM,AS,EU,PA,PO
64 MONDIAL
P = PASSAGERS EN 10**3
10 MIN
15000 MAX
P% = OCCUPATION DES SIEGES P%
30 MIN
30 MAX
PKM = PASSAGERS KM PAYANTS EN 10**6
10 MIN
30000 MAX

JBUNIT

F F = F
1 MIN
2 MAX
EMB = EMB120
1 MIN
5 MAX
C50 = CL50
1 MIN
5 MAX
C215 = CL215
1 MIN
10 MAX
C600 = CL600
1 MIN
5 MAX

...

C=1. AH. ALG. AF. EU. PO. P=40
81. F=1. A310=2. F27=6. B727
=11. B737=16. L100=3.

C=2. MS. CAI. E=9116. AF. AS.
EU. PO. P=2786. P%=59. PKM=4
386. F=1. A300=8. B707=6. B7
37=6. B747=1. B767=3.

C=3. LN. TIP. E=3798. AF. EU.
PO. P=1584. P%=60. PKM=1607
. F=1. F27=17. F28=3. B707=4
. B727=10. IL76=6.

C=4. AT. CAS. E=4531. CX=93.
AF. AM. EU. PO. P=1307. P%=65
. PKM=2573. F=1. F27=2. B707
=2. B727=8. B737=6. B747=2.

C=5. TU. TUN. E=4000. AF. EU.
PO. P=1260. P%=66. PKM=1589
. F=1. A300=1. B727=9. B737=5.

C=6. GI. CKY. E=400. AF. EU. F
=1. A300=1. B727=1. B737=1.
IL18=2. YAK40=1. AN12=1. AN
24=3.

C=7. TM. MPN. E=2000. AF. EU.
P=323. P%=64. PKM=593. F=1.
DC10=1. B737=3. TU134=1. IL
62=2.

C=8. TC. DAR. E=1549. CX=100
. AF. PO. P=499. P%=72. PKM=2
62. F=1. B737=2. TWIN OTTER
=3. F27=3.

C=9. QC. FIH. AF. EU. P=119. P
%=47. PKM=295. F=1. DC8=2. D
C10=2. B737=3. F27=3.

C=10. BP. GABORONE. P=55. P%
=59. PKM=19. EN AF. F=1. L38
2=2. F27=2. D0228=1.

C=11. MX. HARARE. E=1500. P=441.
P%=62. PKM=593. AF. EU.
F=1. VISCOUNT=10. B707=5. D
C8=2. B720=2.

C=12. MD. THR. E=1276. CX=81
. AF. EU. P=361. P%=60. PKM=4
07. F=1. B747=1. B737=2. HS7
48=2. TWIN OTTER=5.

C=13. MK. MRL. E=530. CX=51.
P=157. P%=71. PKM=513. AF. A
S. EU. F=1. B707=2. B747=1. B
737=1. BAE748=1. TWIN OTTE
R=2. F=2. ATR42=1.

C=14. ET. ADD. E=3229. CX=10
0. AF. EU. PO. P=429. P%=49. P
KM=950. F=1. B707=2. B720=5
. B767=2. DC3=9. B727=3. DHC
5=2. TWIN OTTER=5.

C=15. UV. DLA. E=1900. AF. EU
. F=1. B707=1. B747=1. B737=1.
BAE748=2.

C=16. OM. BLZ. E=742. EN AF.
P=128. P%=55. PKM=78. F=1. B
AC1-11=2. BAE748=2. SHORTS
=3.

C=17. GH. ACC. E=2000. CX=10
0. AF. EU. P=280. PKM=410. F=1.
DC10=1. DC9=1. F28=2.

C=18. MT. LOS. E=9000. CX=10
0. AF. AM. EU. PO. F=1. A310=4
. DC10=2. B707=3. B727=2. B7
37=11.

C=19. ZC. MTS. E=150. EN AF.
P=34. P%=54. PKM=23. F=1. F2
8=1.

C=20. LJ. FNA. CX=60. AF. EU.
P=30. P%=32. PKM=122. F=1. B
707=1. BN TRISLANDER=1.

C=21. OZ. LUN. E=1683. CX=10
0. AF. AS. EU. P=246. P%=64. P
KM=604. F=1. B707=4. DC10=1
. B737=1. BAE748=2.

C=22. DT. LAD. E=1600. CX=10
0. AF. AM. EU. P=690. P%=56. P
KM=917. F=1. B707=6. L382=1
. B737=6. F27=5. YAK40=3. CA
9A212=8.

C=23. KO. NBO. E=2690. CX=10
0. AF. AS. EU. PO. P=503. P%=6
0. PKM=1021. F=1. B707=3. B7
20=1. DC9=1. F27=2.

C=24. RK. ABJ. E=4280. AF. EU
. P=674. P%=57. PKM=2095. F=1.
A300=3. DC8=4. DC10=2. B7
27=1.

C=25. UU. ABJ. EN AF. F=1. F2
7=3. F28=2.

C=26. UT. PAP. E=6569. CX=0.
AF. AM. AS. EU. PA. PO. P=852.
P%=66. PKM=4958. F=1. DC10=6.
B747=3. DC8=1. F=2. B74
7=2.

C=27. AF. PAP. E=35232. CX=9
9. AF. AM. AS. EU. PA. PO. P=12
050. P%=68. PKM=28041. F=1.
CONCORDE=7. B747=31. A300=18.
A310=5. B727=29. B737=12.
F=2. A320=25. A310=1. B74
7=1.

C=28. MM. PAR. E=210. CX=0. A
F. AM. EU. P=300. F=1. DC8=4.
CRJ=2.

C=29. IR. MAD. E=24232. CX=9
9. AF. AM. EU. PO. P=12483. P%
=65. PKM=16547. F=1. A30
0=6. DC10=8. B747=6. DC9=28
. B727=36.

C=30. AO. MAD. E=1691. CX=99
. AF. EU. PO. P=4305. P%=79. P
KM=5246. F=1. DC9=20. F27=9

C=31. BX. MAD. E=815. AM. EU.
P=1117. P%=76. PKM=2153. F=1.
CU990=4. DC10=1. DC8=3. B
737=4.

C=32. KM. MLA. E=1120. CX=9
6. AF. EU. PO. P=378. CX=78. P
KM=561. F=1. B720=4. B737=3

C=33. AZ. ROM. E=18144. CX=99.
MONDIAL. P=7826. P%=64. PKM=13658.
F=1. A300=8. B747=9. DC10=3. DC9=17. B7
27=5. MD80=11. ATR42=6. F=2
. MD90=8. B747=1. ATR42=4

C=34. OA. ATH. E=10313. CX=1
00. AF. AM. AS. EU. PO. P=6
878. P%=67. PKM=6312. F=2.
D0228=2. F=1. A300=8. B707=6.
B727=6. B737=11. B747=3.
SHORTS=8. D0228=5.

C=35. CY. NIC. E=1288. CX=76
. EU. PO. P=601. P%=69. PKM=1
266. F2=2. A320=4. F=1. A
310=3. B707=3. BAC1-11=3.

C=36. TK. IST. AF. AS. EU. PO.
E=6714. CX=99. P=2687. P%=6
6. PKM=3004. F=2. A310=7.
F=1. DC10=2. DC9=9. A310=4.
F28=2. B727=9. B707=2. DA
SH7=3.

C=37. JU. BEG. E=6943. CX=10
0. P=3314. P%=66. PKM=3892.
AM. AS. EU. PA. PO. F=1. DC
10=3. B707=2. DC9=12. B727=7.
B737=2. F=2. B737=2.
C=38. JP. LJJ. EU. PO. E=606.
P=1123. PKM=1066. F=2. A320
=5. F=1. DC9=5. MD81=1. M
D82=3. DASH7=2. ROMBAC1-11
=1.

C=39. JJ. BEG. E=330. EN. EU.
P=397. P%=75. PKM=589. F=1.
B727=3. TU134=4.
C=40. LZ. SOF. E=2060. AF. AS
. EU. PO. F=1. TU134=16. T
U154=17. YAK40=12. AN12=3.
AN24=8. IL18=7.
C=41. TP. LIS. E=10000. C%=1
00. AF. AM. EU. P=2125. P%=69
. PKM=4274. F=1. L1011=5
. B707=7. B727=8. B737=7.

C=42. RD. BUH. EU. PO. F=1. IL
62=3. TU154=11. AN24=35. AN
26=6. IL18=12. BAC1-11=10.
ROMBAC1-11=2. F=2
. ROMBAC1-11=4.
C=43. MA. BUD. E=3909. C%=10
0. AF. EU. PO. P=1073. P%=57.
PKM=1239. F=1. TU134=8. T
U154=10. IL18=4.
C=44. OK. PRG. E=5499. C%=10
0. AF. AM. AS. EU. PO. P=1139.
P%=66. PKM=1950. F=1. IL6
2=10. IL18=2. TU134=13. YAK
40=16.

C=45. OS. UIE. E=2939. C%=99
. P=1467. P%=49. PKM=1398. A
F. EU. PO. F=1. DC9=4. MD81=1
1. MD82=1. F=2. A310=2. MD87
=4. MD80=2. F50=2.
C=46. RJ. AMM. E=4125. C%=10
0. AF. AM. EU. PO. P=1328. P%=
56. PKM=3610. F=1. L1011=8.
B747=3. B727=5. B707=3.

C=47. GF. BAH. E=4400. C%=10
0. AS. EU. PO. P=2697. P%=56.
PKM=4423. F=1. L1011=11. B7
37=8.
C=48. IA. BGM. E=5000. AF. AM
. AS. EU. PO. F=1. B747=4. B70
7=2. B727=6. B737=3. AN12=4
. AN24=3. IL76=19.
C=49. IR. THR. E=10200. C%=1
00. P=3884. P%=79. PKM=3874
. AS. EU. PO. F=1. A300=6. B74
7=9. B707=4. B727=9. B737=9

C=50. MD. BEV. E=5000. P=411
. P%=53. PKM=831. AS. EU. PO.
F=1. B747=3. B707=8. B720=8

C=52. FU. FMI. E=5804. C%=10
0. P=1490. P%=56. PKM=3749.
AF. AM. AS. EU. PO. F=1. A310=
8. B747=4. B707=2. B727=4.
F=2. B767=3.
C=51. EU. JED. E=25594. C%=1
00. P=11600. PKM=15845. AM,
AS. EU. PO. F=1. A300=11. L10
11=17. B747=15. DC8=4. B707
=8. B737=20. F28=2. F=2. B74
7=10.

C=53. RB. DAM. E=3300. AF. AS
. EU. PO. P=464. P%=57. PKM=9
23. F=1. B747=2. B727=3. CPU
=2. TU134=4. YAK40=7. AN26=
1. IL76=4. F=2. TU154=3.

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

1 ANNOTATION(S) ?
ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

1 ANNOTATION(S) ?
ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

1 ANNOTATION(S) ?
ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE
ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE

ECRITURE SUR LE FICHIER SORTIE
ECRITURE SUR LE FICHIER SORTIE

1 ANNOTATION(S) ?
ECRITURE SUR LE FICHIER SORTIE

GLOBAL.

C=1.AH.ALG.AF,EU,P0.P=4061.F=1.A310=2.F27=6.B727=11.B737=16.L100=3.

C=2.MS.CAI.E=9116.AF,AS,EU,P0.P=2786.PX=59.PKM=4386.
F=1.A300=8.B707=6.B737=6.B747=1.B767=3.

C=3.LN.TIP.E=3798.AF,EU,P0.P=1584.PX=60.PKM=1607.F=1.F27=17.F28=3.B707=4.B727=10.
IL76=6.

C=4.AT.CAS.E=4531.CX=93.AF,AM,EU,P0.P=1307.PX=65.PKM=2573.F=1.F27=2.B707=2.B727=8.
B737=6.B747=2.

C=5.TU.TUN.E=4000.AF,EU,P0.P=1260.P5=667PKM=1589.F=1.A300=1.B727=9.B737=5.

C=6.GI.CKY.E=400.AF,EU.F=1.A300=1.B727=1.B737=1.IL18=2.YAK40=1.AN12=1.AN24=3.

C=7.TM.MPM.E=2000.AF,EU.P=323.PX=64.PKM=593.F=1.DC10=1.B737=3.TU134=1.IL62=2.

C=8.TC.DAR.E=1549.CX=100.AF,P0.P=499.PX=72.PKM=262.F=1.B737=2.TWIN OTTER=3.F27=3.

C=9.QC.FIH.AF,EU.P=119.PX=47.PKM=295.F=1.DC8=2.DC10=2.B737=3.F27=3.

C=10.BP.GABORONE.P=55.PX597PKM=19.EN AF.F=1.L382=2.F27=2.D0228=1.

C=11.WX.HARARE.E=1500.P=441.PX=62.PKM=593.AF,EU.F=1.VISCOUNT=10.B707=5.DC8=2.B72

0=2.C=12.MD.TNR.E=1276.CX=81.AF,EU.P=361.PX=60.PKM=407.F=1.B747=1.B737=2.HS748=2

.TWIN OTTER=5.C=13.MK.MRL?E=530.CX=51.P=157.PX=71.PKM=513.AF,AS,EU.F=1.B707=2.B7

47=1.B737=1.BAE748=1.TWIN OTTER=2.F=2.ATR42=1.C=14.ET.ADD.E=3229.CX=100.AF,EU,P0.
P=429.PX=49.PKM=950.F=1.B707=2.B720=5.B767=2.DC3=9.B727=3.DHC5=2.TWIN OTTER=5.C

=15.UY.DLA.E=1900.AF,EU.F=1.B707=1.B747=1.B737=1.BAE748=2.C=16.QM.BLZ.E=742.EN A

F.P=128.PX=55.PKM=78.F=1.BAC1-11=2.BAE748=2.SHORTS=3.C=17.GH.ACC.E=2000.CX=100.A

F,EU.P=280.PKM=410.F=1.DC10=1.DC9=1.F28=2.C=18.WT.LOS.E=9000.CX=100.AF,AM,EU,P0.

F=1.A310=4.DC10=2.B707=3.B727=2.B737=11.C=19.ZC.MTS.E=150.EN AF.P=34.PX=54.PKM=2

3.F=1.F28=1.C=20.LJ.FNA.CX=60.AF,EU.P=30.PX=32.PKM=122.F=1.B707=1.BN TRISLANDER=

1.C=21.QZ.LUN.E=1683.CX=100.AF,AS,EU.P=246.PX=64.PKM=604.F=1.B707=4.DC10=1.B737=

1.BAE748=2.C=22.DT.LAD.E=1600.CX=100.AF,AM,EU.P=690.PX=56.PKM=917.F=1.B707=6.L38

2=1.B737=6.F27=5.YAK40=3.CASA212=8.C=23.KQ.NBO.E=2690.CX=100.AF,AS,EU,P0.P=503.P

X=60.PKM=1021.F=1.B707=3.B720=1.DC9=1.F27=2.C=24.RK.ABJ.E=4280.AF,EU.P=674.PX=57

.PKM=2095.F=1.A300=3.DC8=4.DC10=2.B727=1.C=25.VU.ABJ.EN AF.F=1.F27=3.F28=2.C=26.
UT.PAR.E=6569.CX=0.AF,AM,AS,EU,PA,P0.P=852.PX=66.PKM=4958.F=1.DC10=67B747=3.DC6=

1. F=2.B747=2.C=27.AF.PAR.E=35232.CX=99.AF,AM,AS,EU,PA,P0.P=12050.PX=68.PKM=280
41.F=1.CONCORDE=7.B747=31.A300=187A310=5.B727=29.B737=12.F=2.A320=257A310=1.B747

2 ANNOTATION(S) ?
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE

=1.C=28.MM.PAR.E=210.CX=0.AF,AM,EU.P=300.F=1.DC8=4.CRV=2.C=29.ID.MAD.E=24232.CX=99.AF,AM,EU,P0.P=12483.PX=65.PKM=16547. F=1.A300=6.DC10=8?8747=6.DC9=28.8727=3

1 ANNOTATION(S) ?
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE

6.C=30.A0.MAD.E=1691.CX=99.AF,EU,P0.P=4305.PX=79.PKM=5246.F=1.DC9=20.F27=9.C=31.
 BX.MAD.E=815.AM,EU.P=1117.PX=76.PKM=2153.F=1.CV990=4.DC10=1.DC8=3.8737=4.C=32.KH

ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE

.MLA.E=1120.CX=96.AF,EU,P0.P=378.CX=78.PKM=561.F=1.8720=4.8737=3.C=33.AZ.ROM.E=18144.CX=99. MONDIAL.P=7826.PX=64.PKM=13658. F=1.A300=8.8747=9.DC10=3.DC9=17.

ECRITURE SUR LE FICHER SORTIE

.8727=5.MD80=11.ATR42=6.F=2.MD80=8.8747=1. ATR42=4.C=34.OA.ATH.E=10313.CX=100.AF,AM,AS,EU,P0. P=6878.PX=67.PKM=6312. F=2.D0228=2.F=1.A300=8.8707=6.8727=6.8737

ECRITURE SUR LE FICHER SORTIE

=11.8747=3.SHORTS=8.D0228=5.C=35.CY.NIC.E=1288.CX=76.EU,P0.P=601.PX=69.PKM=1266.

2 ANNOTATION(S) ?
 ECRITURE SUR LE FICHER SORTIE

F2=2?A320=4? F=1.A310=3.8707=3.BAC1-11=3.C=36.TK.IST.AF,AS,EU,P0.E=6714.CX=99.P=2687.PX=66.PKM=3004. F=2.A310=7. F=1.DC10=2.DC9=9.A310=4.F28=2.8727=9.8707=2

ECRITURE SUR LE FICHER SORTIE

.DASH7=3.C=37.JU.BEG.E=6943.CX=100.P=3314.PX=66.PKM=3892.AM,AS,EU,PA,P0. F=1.D

ECRITURE SUR LE FICHER SORTIE

C10=3.8707=2.DC9=12.8727=7.8737=2. F=2.8737=2.C=38.JP.LJU.EU,P0.E=606.P=1123.P

ECRITURE SUR LE FICHER SORTIE

KH=1066.F=2.A320=5. F=1.DC9=5.MD81=1.MD82=3.DASH7=2.ROMBAC1-11=1.C=39.JJ.BEG.E

ECRITURE SUR LE FICHER SORTIE

=330.EN EU.P=397.PX=75.PKM=589.F=1.8727=3.TU134=4.C=40.LZ.SOF.E=2060.AF,AS,EU,P0

ECRITURE SUR LE FICHER SORTIE

. F=1.TU134=16.TU154=17.YAK40=12.AN12=3.AN24=8.IL18=7.C=41.TP.LIS.E=10000.CX=1

ECRITURE SUR LE FICHER SORTIE

00.AF,AM,EU.P=2125.PX=69.PKM=4274. F=1.L1011=5.8707=7.8727=8.8737=7.C=42.RO.BU
 H.EU,P0.F=1.IL62=3.TU154=11.AN24=35.AN26=6.IL18=12.BAC1-11=10. ROMBAC1-11=2.

ECRITURE SUR LE FICHER SORTIE

F=2.ROMBAC1-11=4.C=43.HA.BUD.E=3909.CX=100.AF,EU,P0.P=1073.PX=57.PKM=1239. F

ECRITURE SUR LE FICHER SORTIE

=1.TU134=8.TU154=10.IL18=4.C=44.OK.FRG.E=5499.CX=100.AF,AM,AS,EU,P0.P=1139.PX=66

ECRITURE SUR LE FICHER SORTIE

.PKM=1950. F=1.IL62=10.IL18=2.TU134=13.YAK40=16.C=45.OS.VIE.E=2939.CX=99.P=1467
 .PX=49.PKM=1398.AF,EU,P0.F=1.DC9=4.MD81=11.MD82=1.F=2.A310=2.MD87=4.MD80=2.F50=2

ECRITURE SUR LE FICHER SORTIE

.C=46.RJ.AMH.E=4125.CX=100.AF,AM,EU,P0.P=1328.PX=56.PKM=3610.F=1.L1011=8.8747=3.

ECRITURE SUR LE FICHER SORTIE

8727=5.8707=3.C=47.GF.DAH.E=4400.CX=100.AS,EU,P0.P=2697.PX=56.PKM=4423.F=1.L1011

ECRITURE SUR LE FICHER SORTIE

=11.8737=8.C=48.IA.BGW.E=5000.AF,AM,AS,EU,P0.F=1.8747=4.8707=2.8727=6.8737=3.AN1

ECRITURE SUR LE FICHER SORTIE

2=4.AN24=3.IL76=19.C=49.IR.THR.E=10200.CX=100.P=3884.PX=79.PKM=3874.AS,EU,P0.F=1

ECRITURE SUR LE FICHER SORTIE

.A300=6.8747=9.8707=4.8727=9.8737=9.C=50.HE.BEY.E=5000.P=411.PX=53.PKM=831.AS,EU

ECRITURE SUR LE FICHER SORTIE

.P0.F=1.8747=3.8707=8.8720=8.C=52.KU.KWI.E=5804.CX=100.P=1498.PX=56.PKM=3749.AF,

ECRITURE SUR LE FICHER SORTIE

AM,AS,EU,P0.F=1.A310=8.8747=4.8707=2.8727=4. F=2.8767=3.C=51.SV.JED.E=25594.CX=1
 00.P=11600.PKM=15845.AM,AS,EU,P0.F=1.A300=11.L1011=17.8747=15.DC8=4.8707=8.8737=

ECRITURE SUR LE FICHER SORTIE

20.F28=2.F=2.8747=10.C=53.RB.DAM.E=3300.AF,AS,EU,P0.P=464.PX=57.PKM=923.F=1.8747
 =2.8727=3.CRV=2.TU134=4.YAK40=7.AN26=1.IL76=4.F=2.TU154=3.

ECRITURE SUR LE FICHER SORTIE

STOP.

C=1.
AIR ALGERIE.ALGER.AF,EU,PO.PASSAGERS EN 10**3=4061.

F=1.A310=2.F27=4.B727=11.B737=16.L100=3.

C=2.
EGYPTAIR.LE CAIRE.
EMPLOYES=9116.AF,AS,EU,PO.PASSAGERS EN 10**3=2786.OCCUPATION DES SIEGES PX=59.PASSAGERS KM PAYANTS EN 10**6=4386.

F=1.A300=6.B707=6.B737=6.B747=1.B767=3.

C=3.
LIBYAN ARAB AIRLINES.TRIPOLI.
EMPLOYES=3798.AF,EU,PO.PASSAGERS EN 10**3=1584.OCCUPATION DES SIEGES PX=60.PASSAGERS KM PAYANTS EN 10**6=1607.

F=1.F27=17.F28=3.IL76=6.B707=4.B727=10.

C=4.
ROYAL AIR MAROC.CASABLANCA.
EMPLOYES=4531.CAPITAL % ETAT=93.AF,AM,EU,PO.PASSAGERS EN 10**3=1307.OCCUPATION DES SIEGES PX=65.PASSAGERS KM PAYANTS EN 10**6=2573.

F=1.F27=2.B707=2.B727=8.B737=6.B747=2.

C=5.
TUNIS AIR.TUNIS.
EMPLOYES=4000.AF,EU,PO.PASSAGERS EN 10**3=1260.PASSAGERS KM PAYANTS EN 10**6=1589.

F=1.A300=1.B727=9.B737=5.

C=6.
AIR GUINEE.CONAKRY.
EMPLOYES=400.AF,EU.

F=1.A300=1.AN12=1.AN24=3.IL18=2.YAK40=1.B727=1.B737=1.

C=7.
LAM.MAPUTO.
EMPLOYES=2000.AF,EU.PASSAGERS EN 10**3=323.OCCUPATION DES SIEGES PX=64.PASSAGERS KM PAYANTS EN 10**6=593.

F=1.IL62=2.TU134=1.B737=3.DC10=1.

C=8.
AIR TANZANIE.DAR ES SALAM.
EMPLOYES=1549.CAPITAL % ETAT=100.AF,PO.PASSAGERS EN 10**3=499.OCCUPATION DES SIEGES PX=72.PASSAGERS KM PAYANTS EN 10**6=262.

F=1.TWIN OTTER=3.F27=3.B737=2.

C=9.
AIR ZAIRE.KINSHASA.AF,EU.PASSAGERS EN 10**3=119.OCCUPATION DES SIEGES PX=47.PASSAGERS KM PAYANTS EN 10**6=295.

F=1.F27=3.B737=3.DC8=2.DC10=2.

C=10.
AIR BOTSWANA.GABORONE.EN AF.PASSAGERS EN 10**3=55.PASSAGERS KM PAYANTS EN 10**6=19.

F=1.D0228=1.F27=2.L382=2.

C=11.
AIR ZIMBABWE.HARARE.
EMPLOYES=1500.AF,EU.PASSAGERS EN 10**3=441.OCCUPATION DES SIEGES PX=62.PASSAGERS KM PAYANTS EN 10**6=593.

F=1.VISCOUNT=10.B707=5.B720=2.DC8=2.

C=12.
AIR MADAGASCAR.MITAHUARIUO.
EMPLOYES=1276.CAPITAL % ETAT=81.AF,EU.PASSAGERS EN 10**3=361.OCCUPATION DES SIEGES P%=60.PASSAGERS KM PAYANTS EN 10**6=407.

F=1.TWIN OTTER=5.HS748=2.B737=2.B747=1.

C=13.
AIR MAURITIUS.
EMPLOYES=530.CAPITAL % ETAT=51.AF,AS,EU.PASSAGERS EN 10**3=157.OCCUPATION DES SIEGES P%=71.PASSAGERS KM PAYANTS EN 10**6=513.

F=1.TWIN OTTER=2.BAE748=1.B707=2.B737=1.B747=1.
F=2.ATR42=1.

C=14.
ETHIOPIAN AIRLINES.ADDIS ABABA.
EMPLOYES=3229.CAPITAL % ETAT=100.AF,EU,PO.PASSAGERS EN 10**3=429.OCCUPATION DES SIEGES P%=49.PASSAGERS KM PAYANTS EN 10**6=950.

F=1.DHC5=2.TWIN OTTER=5.B707=2.B720=5.B727=3.B767=2.DC3=9.

C=15.
CAMEROON AIRLINES.DOUALA.
EMPLOYES=1900.AF,EU.

F=1.BAE748=2.B707=1.B737=1.B747=1.

C=16.
AIR MALAWI.BLANTYRE.
EMPLOYES=742.EN AF.PASSAGERS EN 10**3=128.OCCUPATION DES SIEGES P%=55.PASSAGERS KM PAYANTS EN 10**6=78.

F=1.BAC1-11=2.BAE748=2.SHORTS=3.

C=17.
GHANA AIRWAYS.ACCRA.
EMPLOYES=2000.CAPITAL % ETAT=100.AF,EU.PASSAGERS EN 10**3=280.PASSAGERS KM PAYANTS EN 10**6=410.

F=1.F28=2.DC9=1.DC10=1.

C=18.
NIGERIA AIRWAYS.LAGOS.
EMPLOYES=9000.CAPITAL % ETAT=100.AF,AM,EU,PO.

F=1.A310=4.B707=3.B727=2.B737=11.DC10=2.

C=19.
ROYAL SWAZI NATIONAL AIRWAYS.MANZINI.
EMPLOYES=150.EN AF.PASSAGERS EN 10**3=34.OCCUPATION DES SIEGES P%=54.PASSAGERS KM PAYANTS EN 10**6=23.

F=1.F28=1.

C=20.
SIERRA LEONE AIRLINES.FREETOWN.CAPITAL % ETAT=60.AF,EU.PASSAGERS EN 10**3=30.OCCUPATION DES SIEGES P%=32.PASSAGERS KM PAYANTS EN 10**6=122.

F=1.BN TRISLANDER=1.B707=1.

C=21.
ZAMBIA AIRWAYS.LUSAKA.
EMPLOYES=1683.CAPITAL % ETAT=100.AF,AS,EU.PASSAGERS EN 10**3=246.OCCUPATION DES SIEGES P%=64.PASSAGERS KM PAYANTS EN 10**6=604.

F=1.BAE748=2.B707=4.B737=1.DC10=1.

C=22.
TAAG.LUANDA.
EMPLOYES=1600.CAPITAL % ETAT=100.AF,AM,EU.PASSAGERS EN 10**3=690.OCCUPATION DES SIEGES P%=56.PASSAGERS KM PAYANTS EN 10**6=917.

F=1.CASA212=8.F27=5.YAK40=3.B707=6.B737=6.L382=1.

C=23.
KENYA AIRWAYS.NAIROBI.
EMPLOYES=2690.CAPITAL % ETAT=100.AF,AS,EU,PO.PASSAGERS EN 10**3=503.OCCUPATION DES SIEGES PZ=60.PASSAGERS KM PAYANTS EN 10**6=1021.
F=1.F27=2.B707=3.B720=1.DC9=1.

C=24.
AIR AFRIQUE.ABIDJAN.
EMPLOYES=4280.AF,EU.PASSAGERS EN 10**3=674.OCCUPATION DES SIEGES PZ=57.PASSAGERS KM PAYANTS EN 10**6=2095.
F=1.A300=3.B727=1.DC8=4.DC10=2.

C=25.
AIR IVOIRE.ABIDJAN.EN AF.
F=1.F27=3.F28=2.

C=26.
UTA.PARIS.
EMPLOYES=6569.CAPITAL % ETAT=0.AF,AM,AS,EU,PA,PO.PASSAGERS EN 10**3=852.OCCUPATION DES SIEGES PZ=66.PASSAGERS KM PAYANTS EN 10**6=4958.
F=1.B747=3.DC8=1.
F=2.B747=2.

C=27.
AIR FRANCE.PARIS.
EMPLOYES=35232.CAPITAL % ETAT=99.AF,AM,AS,EU,PA,PO.PASSAGERS EN 10**3=12050.OCCUPATION DES SIEGES PZ=66.PASSAGERS KM PAYANTS EN 10**6=28041.
F=1.A310=5.CONCORDE=7.B727=29.B737=12.B747=31.
F=2.A310=1.B747=1.

C=28.
MINERVE.PARIS.
EMPLOYES=210.CAPITAL % ETAT=0.AF,AM,EU.PASSAGERS EN 10**3=300.
F=1.CARAVELLE=2.DC8=4.

C=29.
IBERIA.MADRID.
EMPLOYES=24232.CAPITAL % ETAT=99.AF,AM,EU,PO.PASSAGERS EN 10**3=12483.OCCUPATION DES SIEGES PZ=65.PASSAGERS KM PAYANTS EN 10**6=16547.
F=1.A300=6.B727=36.B747=6.DC9=28.

C=30.
AVIACO.MADRID.
EMPLOYES=1691.CAPITAL % ETAT=99.AF,EU,PO.PASSAGERS EN 10**3=4305.OCCUPATION DES SIEGES PZ=79.PASSAGERS KM PAYANTS EN 10**6=5246.
F=1.F27=9.DC9=20.

C=31.
SPANTAX.MADRID.
EMPLOYES=815.AM,EU.PASSAGERS EN 10**3=1117.OCCUPATION DES SIEGES PZ=76.PASSAGERS KM PAYANTS EN 10**6=2153.
F=1.B737=4.CV990=4.DC8=3.DC10=1.

C=32.
AIR MALTA.MALTE.
EMPLOYES=1120.CAPITAL % ETAT=78.AF,EU,PO.PASSAGERS EN 10**3=378.PASSAGERS KM PAYANTS EN 10**6=561.
F=1.B720=4.B737=3.

C=33.

ALITAL.A.ROMF.
EMPLOYES=18144.CAPITAL % ETAT=99.AF,AM,AS,EU,PA,PO.PASSAGERS EN 10**3=7626.OCCUPATION DES SIEGES P2=64.PASSAGERS KM PAYANTS EN 10**6=13656.

F=1.A300=8.ATR42=6.B727=5.B747=9.DC9=17.DC10=3.MD80=11.
F=2.ATR42=4.B747=1.MD80=8.

C=34.
OLYMPIC.ATHENES.
EMPLOYES=10313.CAPITAL % ETAT=100.AF,AM,AS,EU,PO.PASSAGERS EN 10**3=6878.OCCUPATION DES SIEGES PX=67.PASSAGERS KM PAYANTS EN 10**6=6312.

F=1.D0228=5.A300=8.SHORTS=8.B707=6.B727=6.B737=11.B747=3.
F=2.D0228=2.

C=35.
CYPRUS AIRWAYS.NICOSIE.
EMPLOYES=1288.CAPITAL % ETAT=76.EU,PO.PASSAGERS EN 10**3=601.OCCUPATION DES SIEGES PX=69.PASSAGERS KM PAYANTS EN 10**6=1266.

F=1.A310=3.BAC1-11=3.B707=3.

C=36.
THY.ISTANBUL.
EMPLOYES=6714.CAPITAL % ETAT=99.AF,AS,EU,PO.PASSAGERS EN 10**3=2687.OCCUPATION DES SIEGES PX=66.PASSAGERS KM PAYANTS EN 10**6=3004.

F=1.DASH7=3.A310=4.F28=2.B707=2.B727=9.DC9=9.DC10=2.
F=2.A310=7.

C=37.
JAT.BELGRADE.
EMPLOYES=6943.CAPITAL % ETAT=100.AM,AS,EU,PA,PO.PASSAGERS EN 10**3=3314.OCCUPATION DES SIEGES PX=66.PASSAGERS KM PAYANTS EN 10**6=3892.

F=1.B707=2.B727=7.B737=2.DC9=12.DC10=3.
F=2.B737=2.

C=38.
INEX.LJUBLJANA.
EMPLOYES=606.EU,PO.PASSAGERS EN 10**3=1123.PASSAGERS KM PAYANTS EN 10**6=1066.

F=1.DASH7=2.ROMBAC1-11=1.DC9=5.MD81=1.MD82=3.
F=2.A320=5.

C=39.
AVIOGENEX.BELGRADE.
EMPLOYES=330.EN EU.PASSAGERS EN 10**3=397.OCCUPATION DES SIEGES PX=75.PASSAGERS KM PAYANTS EN 10**6=589.

F=1.TU134=4.B727=3.

C=40.
BALKAN.SOFIA.
EMPLOYES=2060.AF,AS,EU,PO.

F=1.AN12=3.AN24=8.IL18=7.TU134=16.TU154=17.YAK40=12.

C=41.
AIR PORTUGAL.LISBONNE.
EMPLOYES=10000.CAPITAL % ETAT=100.AF,AM,EU.PASSAGERS EN 10**3=2125.OCCUPATION DES SIEGES PX=69.PASSAGERS KM PAYANTS EN 10**6=4274.

F=1.B707=7.B727=8.B737=7.L1011=5.

C=42.
TAROM.BUCAREST.EU,PO.

F=1.ROMBAC1-11=2.AN24=35.AN26=6.IL18=12.IL62=3.TU154=11.BAC1-11=10.
F=2.ROMBAC1-11=4

C=43.
MALEV.BUDAPEST.
EMPLOYES=3909.CAPITAL % ETAT=100.AF,EU,PO.PASSAGERS EN 10**3=1073.OCCUPATION DES SIEGES PX=57.PASSAGERS KM PAYANTS EN 10**6=1239.
F=1.IL18=4.TU134=8.TU154=10.

C=44.
CSA.FRAGUE.
EMPLOYES=5499.CAPITAL % ETAT=100.AF,AM,AS,EU,PO.PASSAGERS EN 10**3=1139.OCCUPATION DES SIEGES PX=66.PASSAGERS KM PAYANTS EN 10**6=1950.
F=1.IL18=2.IL62=10.TU134=13.YAK40=16.

C=45.
AUSTRIAN AIRWAYS.VIENNE.
EMPLOYES=2939.CAPITAL % ETAT=99.AF,EU,PO.PASSAGERS EN 10**3=1467.OCCUPATION DES SIEGES PX=49.PASSAGERS KM PAYANTS EN 10**6=1398.
F=1.DC9=4.MD81=11.MD82=1.
F=2.A310=2.F50=2.MD80=2.MD87=4.

C=46.
ALIA.AHMAM.
EMPLOYES=4125.CAPITAL % ETAT=100.AF,AM,EU,PO.PASSAGERS EN 10**3=1328.OCCUPATION DES SIEGES PX=56.PASSAGERS KM PAYANTS EN 10**6=3610
F=1.B707=3.B727=5.B747=3.L1011=8.

C=47.
GULF AIR.BAHREIN.
EMPLOYES=4400.CAPITAL % ETAT=100.AS,EU,PO.PASSAGERS EN 10**3=2697.OCCUPATION DES SIEGES PX=56.PASSAGERS KM PAYANTS EN 10**6=4423.
F=1.B737=8.L1011=11.

C=48.
IRAQI AIRWAYS.BAGDAD.
EMPLOYES=5000.AF,AM,AS,EU,PO.
F=1.AN12=4.AN24=3.IL76=19.B707=2.B727=6.B737=3.B747=4.

C=49.
IRAN AIR.TEHERAN.
EMPLOYES=10200.CAPITAL % ETAT=100.AS,EU,PO.PASSAGERS EN 10**3=3884.OCCUPATION DES SIEGES PX=79.PASSAGERS KM PAYANTS EN 10**6=3874.
F=1.A300=6.B707=4.B727=9.B737=9.B747=9.

C=50.
MIDDLE EAST AIRLINES.BEYROUTH.
EMPLOYES=5000.AS,EU,PO.PASSAGERS EN 10**3=411.OCCUPATION DES SIEGES PX=53.PASSAGERS KM PAYANTS EN 10**6=831.
F=1.B707=8.B720=8.B747=3.

C=51.
SAUDIA.DJEDDAH.
EMPLOYES=25594.CAPITAL % ETAT=100.AM,AS,EU,PO.PASSAGERS EN 10**3=11600.PASSAGERS KM PAYANTS EN 10**6=15845.
F=1.A300=11.F28=2.B707=8.B737=20.B747=15.DC8=4.L1011=17.
F=2.B747=10.

C=52.
KUWAIT AIRWAYS.KOWEIT.
EMPLOYES=5804.CAPITAL % ETAT=100.AF,AM,AS,EU,PO.PASSAGERS EN 10**3=1498.OCCUPATION DES SIEGES PX=56.PASSAGERS KM PAYANTS EN 10**6=3749.
F=1.A310=8.B707=2.B727=4.B747=4.
F=2.B767=3.

C=53.

SYRIAN ARAB AIRLINES.DAHMS.

EMPLOYES=3300.AF,AS.EU.PO.PASSAGERS EN 10**3=464.OCCUPATION DES SIEGES PX=57.PASSAGERS KM PAYANTS EN 10**6=923.

F=1.CARAVELLE=2.AN26=1.IL76=4.TU134=4.YAK40=7.B727=3.B747=2.

F=2.TU154=3.

LOCAL.
 C=5.P%=66.
 C=10.P%=59.
 C=13.MRU.
 C=26.F=1.DC10=6.
 C=27.F=1.A300=18. F=2.A320=25.
 C=29. F=1. DC10=8.
 C=35.F=2. A320 = 4.
 STOP.

ENTRÉE

5

ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE
 ECRITURE SUR LE FICHER SORTIE

SORTIE IMPRIMÉE

LOCAL.
 C=5.P%=66.
 C=10.P%=59.
 C=13.MRU.
 C=26.F=1.DC10=6.
 C=27.F=1.A300=18. F=2.A320=25.
 C=29. F=1. DC10=8.
 C=35.F=2. A320 = 4.
 STOP.

5 0 *

10, 13, 26, 27, 29, 35.

=5.

UNIS AIR.TUNIS.

MPLOYES=4000.AF.EU.PO.PASSAGERS EN 10**3=1260.OCCUPATION DES SIEGES PX=66.PASSAGERS KM PAYANTS EN 10**6=1589.

=1.A300=1.8727=9.8737=5.

=10.

IR BOTSWANA.GABORONE.EN AF.PASSAGERS EN 10**3=55.OCCUPATION DES SIEGES PX=59.PASSAGERS KM PAYANTS EN 10**6=19.

=1.D0228=1.F27=2.L382=2.

=13.

IR MAURITIUS.PORT LOUIS.

MPLOYES=530.CAPITAL % ETAT=51.AF,AS,EU.PASSAGERS EN 10**3=157.OCCUPATION DES SIEGES PX=71.PASSAGERS KM PAYANTS EN 10**6=513.

=1.TWIN OTTER=2.BAE748=1.8707=2.8737=1.8747=1.

=2.ATR42=1.

=26.

TA.PARIS.

MPLOYES=6569.CAPITAL % ETAT=0.AF,AM,AS,EU,PA,PO.PASSAGERS EN 10**3=852.OCCUPATION DES SIEGES PX=66.PASSAGERS KM PAYANTS EN 10**6=4958.

=1.8747=3.DC8=1.DC10=6.

=2.8747=2.

=27.

IR FRANCE.PARIS.

MPLOYES=35232.CAPITAL % ETAT=99.AF,AM,AS,EU,PA,PO.PASSAGERS EN 10**3=12050.OCCUPATION DES SIEGES PX=68.PASSAGERS KM PAYANTS EN 10**6=28041.

=1.A300=18.A310=5.CONCORDE=7.8727=29.8737=12.8747=31.

=2.A310=1.A320=25.8747=1.

=29.

BERIA.MADRID.

MPLOYES=24232.CAPITAL % ETAT=99.AF,AM,EU,PO.PASSAGERS EN 10**3=12483.OCCUPATION DES SIEGES PX=65.PASSAGERS KM PAYANTS EN 10**6=16547.

=1.A300=6.8727=36.8747=6.DC9=28.DC10=8.

=35.

CYPRUS AIRWAYS.NICOSIE.

MPLOYES=1288.CAPITAL % ETAT=76.EU,PO.PASSAGERS EN 10**3=601.OCCUPATION DES SIEGES PX=69.PASSAGERS KM PAYANTS EN 10**6=1266.

=1.A310=3.BAC1-11=3.8707=3.

=2.A320=4.

5 0

SIEGE=MADRID.

29 30 31

53 ENREGISTREMENTS ONT ETE LUS
3 ENREGISTREMENTS ONT ETE EXTRAITS

5 0

A320>0.

27 35 38

53 ENREGISTREMENTS ONT ETE LUS
3 ENREGISTREMENTS ONT ETE EXTRAITS

5 0

A300 > 0 ! A310 > 0 ! A320>0

1 2 5 6 18 24 27 29 33 34 35 36 38 45 49 51 52

53 ENREGISTREMENTS ONT ETE LUS
17 ENREGISTREMENTS ONT ETE EXTRAITS

5 0

(A300 > 0 ! A310 > 0 ! A320>0) & (8707>0 ! 8727>0 ! 8737>0 ! 8747>0).

1 2 5 6 18 24 27 29 33 34 35 36 49 51 52

53 ENREGISTREMENTS ONT ETE LUS
15 ENREGISTREMENTS ONT ETE EXTRAITS

5 0

A320(2) > 0 & 8747(2)>0.

27

53 ENREGISTREMENTS ONT ETE LUS
1 ENREGISTREMENTS ONT ETE EXTRAITS

5 0

CAPITAL % ETAT (30 & CX = -1 &
EMPLOYES > 5000.

26

53 ENREGISTREMENTS ONT ETE LUS
1 ENREGISTREMENTS ONT ETE EXTRAITS

A320>0

C=27.

AIR FRANCE.PARIS.

EMPLOYES=35232.CAPITAL % ETAT=99.AF.AM.AS.EU.PA.PO.PASSAGERS EN 10**3=12050.OCCUPATION DES SIEGES PX=68.PASSAGERS KM PAYANTS EN 10**6=28041.

F=1.A300=18.A310=5.CONCORDE=7.B727=29.B737=12.B747=31.

F=2.A310=1.A320=25.B747=1.

C=35.

CYPRUS AIRWAYS.NICOSIE.

EMPLOYES=1288.CAPITAL % ETAT=76.EU.PO.PASSAGERS EN 10**3=601.OCCUPATION DES SIEGES PX=69.PASSAGERS KM PAYANTS EN 10**6=1266.

F=1.A310=3.BAC1-11=3.B707=3.

F=2.A320=4.

C=38.

INEX.LJUBLJANA.

EMPLOYES=606.EU.PO.PASSAGERS EN 10**3=1123.PASSAGERS KM PAYANTS EN 10**6=1066.

F=1.DASH7=2.ROMBAC1-11=1.DC9=5.MDB1=1.MDB2=3.

F=2.A320=5.

RAPPEL DES REGLES D'ALGEBRE DE BOOLE

Sur des opérandes booléens A, B, C (A, B, C ont la valeur "vrai" = V ou "faux" = F) peuvent porter les OPERATEURS suivants:

- 1) unaires : "^" (non) ou nul (vide)
- 2) binaires: "&" (et) ou "!" (ou)

TABLES DE VERITE DES OPERATEURS

NON		ET			OU		
A	$\neg A$	A	B	$A \& B$	A	B	$A ! B$
V	F	V	V	V	V	V	V
F	V	V	F	F	V	F	V
		F	V	F	F	V	V
		F	F	F	F	F	F

SYMETRIE

$$\begin{aligned} A ! B &= B ! A \\ A \& B &= B \& A \end{aligned}$$

ASSOCIATIVITE

$$\begin{aligned} (A ! B) ! C &= A ! (B ! C) \\ (A \& B) \& C &= A \& (B \& C) \end{aligned}$$

DOUBLE DISTRIBUTIVITE

$$\begin{aligned} A ! (B \& C) &= (A ! B) \& (A ! C) \\ A \& (B ! C) &= (A \& B) ! (A \& C) \end{aligned}$$

LOIS DE MORGAN

$$\begin{aligned} \neg(A ! B) &= (\neg A) \& (\neg B) \\ \neg(A \& B) &= (\neg A) ! (\neg B) \end{aligned}$$

$$\neg(\neg A) = A$$

PRIORITE DES OPERATEURS

$\neg$	(1)	ordre décroissant
$\&$	(2)	
$!$	(3)	

Quand il y a des parenthèses, c'est ce qui est au niveau le plus interne qui est d'abord exécuté avec les règles courantes.

↓

	0	1	2	3	4	5	6	7	8	9	0
	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890
1 C I.....											
2 C+ I.....											
3 S+ I.....											
4 E I.....											
5 C% I.....											
6 R\$ I.....											
7 P I.....											
8 PX I.....											
9 PKH I.....											
I F 11111111111211111111112211112222211121121111222											
2 EMB I.....											
3 CSO I.....											
4 C215 I.....											
5 C600 I.....											
6 C601 I.....											
7 DHC5 I.....											
8 DHC6 I.....											
9 DASH I.....											
10 DSH8 I.....											
11 TWIN I.....											
12 V410 I.....											
13 V610 I.....											
14 Y7 I.....											
15 Y12 I.....											
16 D228 I.....											
17 C212 I.....											
18 C235 I.....											
19 A300 I.....											
20 A310 I.....											
21 A320 I.....											
22 FA10 I.....											
23 FA20 I.....											
24 FA50 I.....											
25 CRV I.....											
26 MERC I.....											
27 N262 I.....											
28 CONC I.....											
29 ATR4 I.....											
30 ATR7 I.....											
31 S600 I.....											
32 F27 I.....											
33 F28 I.....											
34 F50 I.....											
35 F100 I.....											
36 R111 I.....											
37 R141 I.....											
38 AN12 I.....											
39 AN24 I.....											
40 AN26 I.....											
41 AN30 I.....											
42 V400 I.....											
43 IL18 I.....											
44 IL62 I.....											
45 IL76 I.....											
46 IL86 I.....											
47 I134 I.....											

I C	53	53
I C↓	53	53
I S↓	53	53
I E →	47	47
I C%	31	31
I R↓	53	53
I F	46	46
I P%	40	40
I Pkm	44	44
I F	67	67
I EmB	0	0
I C50	0	0
I C215	0	0
I C600	0	0
I C601	0	0
I DHC5	1	1
I DHC6	0	0
I DASH	2	2
I DSHB	0	0
I TWIN	4	4
I V410	0	0
I V610	0	0
I Y7	0	0
I Y12	0	0
I D228	3	3
I C212	1	1
I C235	0	0
I A300	10	10
I A310	9	9
I A320	3	3
I FA10	0	0
I FA20	0	0
I FA50	0	0
I CRJ	2	2
I MERC	0	0
I N262	0	0
I CONC	1	1
I ATR4	3	3
I ATR7	0	0
I S600	0	0
I F27	10	10
I F28	6	6
I F50	1	1
I F100	0	0
I R111	3	3
I R141	0	0
I AN12	3	3
I AN24	4	4
I AN26	2	2
I AN30	0	0
I V400	0	0
I IL18	5	5
I IL62	3	3
I IL76	3	3
I IL86	0	0
I IL96	0	0

736

I 1134	4	4
I 1A40	5	5
I YA42	0	0
I S340	0	0
I B111	3	3
I B125	0	0
I B146	0	0
I B748	4	4
I JS31	0	0
I SH0R	2	2
I TR1D	0	0
I VISC	1	1
I BE99	0	0
I BE19	0	0
I H748	1	1
I BNTR	1	1
I SWM	0	0
I B707	23	23
I B720	5	5
I B727	21	21
I B737	25	25
I B747	21	21
I B757	0	0
I B767	3	3
I C580	0	0
I C990	1	1
I DC3	1	1
I DC8	7	7
I DC9	9	9
I DC10	12	12
I L100	1	1
I L382	2	2
I L410	0	0
I 3STA	4	4
I MD80	3	3
I MD81	2	2
I MD82	2	2
I MD83	0	0
I MD87	1	1

↑
 $L=14$

[illegible]

STT= 0 = 736#

EXTRAIT DE SYNOPSIS2.

page 23

52 KWI 1
 53 DAMAS 1
 53 DAM 1
 -----TOTAL = 53

EMPLOYES
 150 = MIN
 35232 = MAX
 -----TOTAL = 47

CAPITAL % ETAT
 0 = MIN
 100 = MAX
 -----TOTAL = 31

ESEAU
 2 EN AF 4
 5 EN EU 1
 10 AF,EU 9
 12 AF,PO 1
 14 AM,EU 1
 21 EU,PO 3
 24 AF,AM,EU 3
 27 AF,AS,EU 2
 31 AF,EU,PO 8
 37 AS,EU,PO 3
 47 AF,AM,EU,PO 4
 50 AF,AS,EU,PO 5
 54 AM,AS,EU,PO 1
 59 AF,AM,AS,EU,PO 4
 63 AM,AS,EU,PA,PO 1
 64 AF,AM,AS,EU,PA,PO 3
 64 MONDIAL 3

-----TOTAL = 53
 ASSAGERS EN 10**3
 30 = MIN
 12483 = MAX
 -----TOTAL = 46

OCCUPATION DES SIEGES PX
 32 = MIN
 79 = MAX
 -----TOTAL = 40

ASSAGERS KM PAYANTS EN 10**6
 19 = MIN
 28041 = MAX
 -----TOTAL = 44

1 = MIN
 2 = MAX
 ----- = 67

MB120
 L50
 L215
 L600
 L601
 HC5

2 = MIN
 2 = MAX
 ----- = 1

PASSAGERS EN 10**3	-----TOTAL =	53
30 = MIN		
12483 = MAX		
OCCUPATION DES SIEGES PZ	-----TOTAL =	46
32 = MIN		
79 = MAX		
PASSAGERS KM PAYANTS EN 10**6	-----TOTAL =	40
19 = MIN		
28041 = MAX		
F	-----TOTAL =	44
1 = MIN		
2 = MAX		
EMB120	-----	67
CL50		
CL215		
CL600		
CL601		
DHC5		
2 = MIN		
2 = MAX		
DHC6	-----	1
DASH7		
2 = MIN		
3 = MAX		
DASH8	-----	2
TWIN OTTER		
2 = MIN		
5 = MAX		
LET410	-----	4
LET610		
Y7		
Y12		
D0228		
1 = MIN		
5 = MAX		
CASA212	-----	3
8 = MIN		
8 = MAX		
CN235	-----	1
A300		
1 = MIN		
18 = MAX		
A310	-----	10
1 = MIN		
8 = MAX		
A320	-----	9
4 = MIN		
25 = MAX		

ANNEXE

MISES A JOUR GLOBALES ET LOCALES: RESUME IMAGE

1. MODES DE SAISIE

Tout d'abord il y a deux modes de saisie en vue de stocker et/ou mettre à jour les informations enregistrées: le mode "LOCAL" et le mode "GLOBAL".

Le mode LOCAL permet de MODIFIER une ou plusieurs données déjà enregistrées et seulement celles-ci;

le mode GLOBAL EFFACE D'ABORD toutes les données éventuellement enregistrées.

Attention, ces deux modes sont donc de conception totalement différentes. Comme ils peuvent être utilisés alternativement et à volonté au cours d'un seul traitement, en passant d'un mode à l'autre il faut prendre des précautions.

2. CLEFS

Quel que soit le mode de saisie utilisé, il faut indiquer au système sur quelles données on va travailler, donc donner les clefs d'identification

- d'abord de l'unité (niveau unité ou "UNIT" du répertoire)
- puis de la sous-unité à l'intérieur de l'unité quand c'est le cas (niveau sous-unité ou "SUBUNIT" du répertoire).

Afin de mieux comprendre, nous prendrons comme exemple l'image d'un APPARTEMENT FERME A CLEF.

Pour ouvrir la porte principale, on a besoin d'une clef - que l'on appelle "clef d'accès" - . On la désigne par son numéro, ce qui donne l'autorisation d'accéder à l'information.

La porte principale ayant été ouverte avec sa clef, on se trouve dans un VESTIBULE sur lequel donne des PIECES numérotées elles aussi FERMEES A CLEF. L'appartement est L'UNITE d'enregistrement, avec sa clef d'accès numérotée. Les pièces sont les SOUS-UNITES, avec leur clef d'accès numérotée.

Supposons que l'on désire MEUBLER cet appartement:

- En mode GLOBAL, le simple fait d'ouvrir avec la clef commence par vider entièrement l'appartement de tous les meubles qui pouvaient s'y trouver (données enregistrées). Les meubles qu'on apporte vont le garnir, ou le regarnir s'il était meublé auparavant.

- En mode LOCAL, l'ouverture avec la clef ne touche à rien. On pourra ajouter des meubles (données), en remplacer ou en supprimer.

Rappelons qu'un changement de mode fait ressortir de l'appartement (unité). Donc on doit fournir immédiatement la clef de l'appartement où on veut accéder dans ce nouveau mode et ce peut être le même appartement.

PAR DEFAULT, ON RESTE DANS LE MODE QUI A ETE INDIQUE.

3. PROPRIETES

Pour la saisie en vue du stockage après validation, on peut continuer avec l'image de l'appartement à meubler.

Les meubles sont destinés à avoir chacun une place déterminée (valeurs ou données destinés aux variables). Si, par exemple, les variables sont les types de meubles ayant des

fonctions différentes, les données sont les modèles choisis dans les listes d'un catalogue (répertoire).

Quel que soit le mode de saisie, une fois qu'on a OUVERT la porte de l'appartement, on doit apporter D'ABORD les meubles du VESTIBULE, mais L'ORDRE DANS LEQUEL ON LES APPORTE N'A PAS D'IMPORTANCE, car le SYSTEME SE CHARGE de les mettre à la place à laquelle ils sont destinés. C'est la VALIDATION.

S'ils ne sont PAS CONFORMES à la commande (données mal orthographiées, etc.) ou d'un modèle non répertorié (pas dans les listes), le système les éjecte (rejette). (Les données en question ne sont pas enregistrées).

D'autre part,

- il peut MANQUER des meubles, cela n'a pas d'importance,
- ou, au contraire, certains peuvent être EN DOUBLE , ou triple, etc., soit parce qu'il y en a déjà un à l'endroit auquel il est destiné (on est forcément en mode local) (il y a une donnée enregistrée pour la variable), soit parce qu'on en apporte plus d'un, et ceci en nombre quelconque (on a plusieurs données valables pour la variable), dans ce cas, le système CONSERVE AUTOMATIQUEMENT LE DERNIER VENU et éjecte (rejette) les autres..

Lorsqu'on a terminé avec le vestibule, on ouvre la porte d'une pièce quelconque avec la clef qui lui est affectée, en indiquant son numéro. On peut toujours apporter les meubles dans n'importe quel ordre, le système s'en charge exactement de la même façon que pour le vestibule, avec les mêmes propriétés.

Puis on ouvre une autre pièce avec sa clef, on répète le même processus.

Attention:

En mode local, absolument tout ce qu'on apporte après avoir ouvert la porte principale doit être correct (c'est-à-dire il n'y a aucune annotation sur le listing de sortie de validation), sinon, par sécurité, le système rejette en bloc la totalité de la livraison pour cet appartement:

il n'y aura PAS de message "ECRITURE SUR LE FICHIER SORTIE", donc aucune modification proposée pour l'enregistrement n'est retenue. Il faut y prendre garde.

COMMENTAIRES

Si on garde l'image de l'aménagement d'un appartement, on peut comparer les commentaires à des meubles - ou des plantes ou des objets décoratifs - qui ne sont pas encore répertoriés, ou que l'observateur ne veut pas répertorier.

On peut les apporter avec les meubles et mélangés avec ceux-ci. Pour que le système les reconnaisse immédiatement (sans avoir à chercher s'ils sont répertoriés), on les marque d'un fanion (on les fait précéder d'un caractère spécial qu'on a fourni dans le répertoire, et on marque la fin par un ".").

Ceux qu'on veut mettre dans le vestibule doivent y être mis avant de le quitter pour ouvrir une des pièces.

Ceux qu'on veut mettre dans une pièce ne peuvent y être mis qu'après avoir ouvert la pièce avec sa propre clef.

On les entre en nombre limité (le nombre maximum choisi pour l'unité et par sous-unité en mettant en place l'application).

L'ouverture par la clef, en mode global, les expulse avec le reste.

D'autre part, si on est en mode local, l'apport d'un nouvel objet marqué (commentaire) l'ajoute à ceux qui étaient déjà sur place (concaténation), ce qui est propre à ce type d'objets comme on l'a vu..

4. RESUME

Lors d'un traitement, on saisit:

- a. Le MODE, sous forme d'un des deux mots-clefs "GLOBAL" ou "LOCAL", introduisant instantanément le mode de saisie ainsi désigné.
- b. La CLEF d'accès à l'(unité d')enregistrement, quantitative entière, identifiée de façon unique par son numéro. Deux enregistrements ne peuvent qu'avoir des numéros différents.
- c. Les INFORMATIONS concernant l'unité, qualitatives ou quantitatives, dans n'importe quel ordre, en nombre quelconque (s'il y en a plusieurs valides pour une variable, c'est la dernière qui sera enregistrée), parsemées de commentaires optionnels.
- d. La CLEF d'accès à une sous-unité identifiée de façon unique par son numéro relativement à l'unité.
- e. Les INFORMATIONS concernant cette sous-unité, avec les mêmes possibilités que pour c.

On répète d,e autant de fois qu'on a de sous-unités.

On répète b à e autant de fois qu'on a d'enregistrements à saisir ^{dans} de le mode désigné en a, et qui ne changera que par la donnée explicite du mot-clef "LOCAL" succédant à "GLOBAL" et vice versa.

La saisie des informations répertoriées doit être conforme à leur orthographe telle qu'on l'a précisée dans le répertoire où le système va chercher à les identifier. Toutefois, outre les présentations synonymes, on peut toujours insérer des blancs DEVANT une "donnée" (au sens du logiciel), et aussi, pour les données quantitatives, devant le signe = et devant l'entier.

On fait la saisie en continu, caractère par caractère. On sait que les blancs entre deux points et les points successifs sont ignorés: tout ceci est susceptible de faciliter la saisie. Les mots-clefs GLOBAL et LOCAL sont saisis comme des données.


```

DO WHILE (REP = 'UNIT') :/*LIRE ET IMPRIMER LES CARTES SUI-
GET COPY EDIT (REP, TXT) :/*VANTES JUSQU'A AVOIR TROUVE LA
(A(9),X(15),A(56)) :/*CARTE 'UNIT'
IC=IC+1; :/*
END :/*

DO WHILE (REP = 'SUBUNIT') :/*FAIRE JUSQU'A AVOIR TROUVE LA
:/*CARTE 'SUBUNIT'
GET COPY EDIT (CARTE) :/*LIRE ET IMPRIMER LA CARTE SUI-
(A(9),A(3),A(4),A(6),A(2),A(56)) :/*VANTE, DECODER DANS 'CARTE'.
IC=IC+1; :/*

IF REP = ' ' :/*SI 'REP' EST BLANC, C'EST
THEN :/*UNE CARTE VARIABLE OU DONNEE
DO :/*DONT LE CHAMP 'TXT' N'EST PAS
DO N = 56 TO 1 BY -1 WHILE :/*BLANC, CALCULER LE NOMBRE N DE
(SUBSTR(TXT,N,1) = ' ') :/*CARACTERES SIGNIFICATIFS DANS
END :/*'TXT'.

IF N = 0 :/*SI N EST NUL, 'TXT' EST BLANC
THEN CALL ERREUR(0) :/*C'EST UNE ERREUR,
ELSE :/*SINON CONTINUER
IF VAR = ' ' :/*SI 'VAR' N'EST PAS BLANC,
THEN :/*C'EST UNE CARTE VARIABLE
IF J+1 > UMAXV :/*SI TROP DE VARIABLES,
THEN CALL ERREUR (1) :/*C'EST UNE ERREUR
ELSE :/*SINON, CONTINUER
DO :/*TRAITER UNE VARIABLE DE PLUS
J = J+1 :/*POUR UNIT
U.V(J).CODE = VAR :/*RANGE LE CODE (OU NOM ABREGE)
DO NC=4 TO 1 BY -1
WHILE (SUBSTR(VAR,NC,1) = ' ');
END;
U.V(J).NCARC=NC;
U.V(J).CHAP = CHP=' ' :/*RANGE L'INDICATEUR DE CHAPITRE
U.V(J).TYPE = OTT=' ' :/*RANGE LE TYPE ET LE NOMBRE DE
U.V(J).NCAR = N :/*CARACTERES SIGNIFICATIFS DU NOM
U.V(J).INDC = L :/*RANGE LA POSITION DANS LA ZONE
IF L+N > UMAXC :/*CARACTERE DE 'UNIT', SI TROP
THEN CALL ERREUR (2) :/*DE CARACTERES, C'EST UNE ERREUR
ELSE :/*SINON,
DO :/*ON PEUT RANGER LE NOM DANS LA
SUBSTR(UC,L,N) = TXT :/*ZONE CARACTERE DE UNIT
L = L+N :/*AVANCE LA POSITION DANS LA ZONE
END :/*DE N CARACTERES
U.V(J).VAL1 = K+1 :/*RANGE L'INDEX DE LA PREMIERE
U.V(J).VAL2 = 0 :/*DONNEE, ET O POUR LA DERNIERE
IF U.V(J).TYPE :/*POUR LES VARIABLES DE TYPE
THEN :/*QUANTITATIF
DO :/*
U.V(J).VAL1 = UVAL1 :/*RANGE LE MINIMUM
U.V(J).VAL2 = UVAL2 :/*ET LE MAXIMUM IMPLICITES
END :/*FIN DU TRAITEMENT
ELSE :/*D'UNE VARIABLE
IF VAL = ' ' :/*SI 'VAL' N'EST PAS BLANC,
THEN :/*C'EST UNE CARTE DONNEE
IF U.V(J).TYPE :/*POUR UNE VARIABLE QUANTITATIVE,
THEN :/*ON DOIT TROUVER MIN,
IF TXT = 'MIN' :/*SI OUI, RANGER MIN
THEN U.V(J).VAL1 = VAL :/*
ELSE :/*OU MAX,
IF TXT = 'MAX' :/*SI OUI, RANGER MAX
THEN U.V(J).VAL2 = VAL :/*SI NON, ERREUR
ELSE CALL ERREUR(3) :/*POUR UNE VARIABLE QUALITATIVE
ELSE :/*SI TROP DE DONNEES,
IF K+1 > UMAXD :/*C'EST UNE ERREUR
THEN CALL ERREUR (4) :/*SINON,
DO :/*
K = K+1 :/*RANGER LA VALEUR
U.D(K).EQUV = VAL :/*ET LE TEXTE DE LA DONNEE DE LA
U.D(K).NCAR = N :/*MEME MANIERE QUE LE NOM D'UNE
U.D(K).INDC = L :/*VARIABLE
IF L+N > UMAXC :/*
THEN CALL ERREUR (5) :/*
ELSE :/*
DO :/*
SUBSTR(UC,L,N) = TXT :/*
L = L+N :/*
END :/*
U.V(J).VAL2 = K :/*RANGE L'INDEX DE LA DONNEE
END :/*FIN DU TRAITEMENT D'UNE DONNEE
ELSE CALL ERREUR (6) :/*NI VARIABLE, NI DONNEE: ERREUR
END :/*

END :/*

U.NVAR = J :/*RANGE LE NOMBRE DE VARIABLES
U.NEQU = K :/*ET LE NOMBRE D'EQUIVALENCES
IF J = 0 THEN CALL ERREUR (7) :/*IL FAUT AU MOINS UNE VARIABLE
DO I = 1 TO U.NVAR :/*
IF U.V(I).VAL1 > U.V(I).VAL2 :/*ET CHAQUE VARIABLE DOIT AVOIR
THEN :/*AU MOINS UNE DONNEE, OU UN
DO :/*
CALL ERREUR (8) :/*MINIMUM INFERIEUR OU EGAL AU
PUT EDIT(' VAR',I)(A,F(3)); :/*
END; :/*MAXIMUM.
END :/*
:/*
:/*
:/*VERIFIER L'UNICITE DE TOUS LES
:/*LIBELLES (NOMS, CODES ET EQUI-
:/*VALENCES) DANS 'UNIT'

DO I = 1 TO U.NVAR :
DO J = I+1 TO U.NVAR :
IF U.V(I).CODE = U.V(J).CODE | U.V(I).CODE = SUBSTR(UC,U.V(J).INDC
U.V(J).NCAR) | SUBSTR(UC,U.V(I).INDC,U.V(I).NCAR) = U.V(J).CODE
SUBSTR(UC,U.V(I).INDC,U.V(I).NCAR) = SUBSTR(UC,U.V(J).INDC
U.V(J).NCAR) THEN CALL DUPLI(U.V(I).CODE) ; END ;
DO K = 1 TO U.NEQU :
IF U.V(I).CODE = SUBSTR(UC,U.D(K).INDC,U.D(K).NCAR) | SUBSTR(UC
U.V(I).INDC,U.V(I).NCAR) = SUBSTR(UC,U.D(K).INDC,U.D(K).NCAR)
THEN CALL DUPLI(U.V(I).CODE) ; END ;
END ;
DO J = 1 TO U.NEQU ; DO K = J+1 TO U.NEQU :
IF SUBSTR(UC,U.D(J).INDC,U.D(J).NCAR) = SUBSTR(UC,U.D(K).INDC
U.D(K).NCAR) THEN CALL DUPLI(SUBSTR(UC,U.D(J).INDC,U.D(J).NCAR));
END ;
END ;
:/*
:/*FIN DU TRAITEMENT DE UNIT
:/*
:/*TRAITEMENT DE SUBUNIT
:/*
J = 0 :/*NUMERO DE VARIABLE

```

```

L = 1
/* POSITION DANS LA CHAÎNE DE
/* CARACTÈRES C
*/

DO WHILE (REP = 'STOP')
/* FAIRE JUSQU'À AVOIR TROUVÉ LA
/* CARTE 'STOP'
*/
GET COPY EDIT (CARTE)
/* LIRE ET IMPRIMER LA CARTE SUI-
/* VANTE. DECODER DANS 'CARTE'.
*/
(A(9).A(3).A(4).A(6).A(2).A(56))
IC=IC+1;
/* *****
/*

IF REP = ' '
/* SI 'REP' EST BLANC. C'EST
/* UNE CARTE VARIABLE OU DONNÉE
/* DONT LE CHAMP 'TXT' N'EST PAS
/* BLANC. CALCULER LE NOMBRE N DE
/* CARACTÈRES SIGNIFICATIFS DANS
/* 'TXT'.
*/
THEN
DO
DO N = 56 TO 1 BY -1 WHILE
(SUBSTR(TXT,N,1) = ' ')
END
/* SI N EST NUL. 'TXT' EST BLANC
/* C'EST UNE ERREUR.
/* SINON CONTINUER
/* SI 'VAR' N'EST PAS BLANC.
/* C'EST UNE CARTE VARIABLE
/* SI TROP DE VARIABLES.
/* C'EST UNE ERREUR
/* SINON, CONTINUER
/* TRAITER UNE VARIABLE DE PLUS
/* POUR SUBUNIT
/* RANGE LE CODE (OU NOM ABREGÉ)
*/
IF N = 0
THEN CALL ERREUR (9)
ELSE
IF VAR = ' '
THEN
IF J+1 > SMAXV
THEN CALL ERREUR (10)
ELSE
DO
J = J+1
S.V(J).CODE = VAR
DO NC=4 TO 1 BY -1
WHILE (SUBSTR(VAR,NC,1) = ' ');
END;
S.V(J).NCARC=NC;
S.V(J).CHAP = CHP=' '
/* RANGE L'INDICATEUR DE CHAPITRE
/* RANGE LE TYPE ET LE NOMBRE DE
/* CARACTÈRES SIGNIFICATIFS DU NOM
/* RANGE LA POSITION DANS LA ZONE
/* CARACTÈRE DE SUBUNIT. SI TROP
/* DE CARACTÈRES. C'EST UNE ERREUR
/* SINON,
/* ON PEUT RANGER LE NOM DANS LA
/* ZONE CARACTÈRE DE SUBUNIT
/* AVANCE LA POSITION DANS LA ZONE
/* DE N CARACTÈRES
/* RANGE L'INDEX DE LA PREMIÈRE
/* DONNÉE, ET 0 POUR LA DERNIÈRE
/* POUR LES VARIABLES DE TYPE
/* QUANTITATIF
/*
/* RANGE LE MINIMUM
/* ET LE MAXIMUM IMPLICITES
/* FIN DU TRAITEMENT
/* D'UNE VARIABLE
/*
/* SI 'VAL' N'EST PAS BLANC.
/* C'EST UNE CARTE DONNÉE
/* POUR UNE VARIABLE QUANTITATIVE.
/*
/* ON DOIT TROUVER MIN.
/* SI OUI. RANGER MIN
/*
/* OU MAX.
/* SI OUI. RANGER MAX
/* SI NON. ERREUR
/* POUR UNE VARIABLE QUALITATIVE
/* SI TROP DE DONNÉES.
/* C'EST UNE ERREUR
/* SINON.
/*
/* RANGER LA VALEUR
/* ET LE TEXTE DE LA DONNÉE DE LA
/* MÊME MANIÈRE QUE LE NOM D'UNE
/* VARIABLE
/*
/*
/*
/* RANGE L'INDEX DE LA DONNÉE
/* FIN DU TRAITEMENT D'UNE DONNÉE
/* NI VARIABLE. NI DONNÉE: ERREUR
*/
END
/* VAL = ' '
/*
/* SI S.V(J).TYPE
/*
/* IF TXT = 'MIN'
/* THEN S.V(J).VAL1 = VAL
/* ELSE
/* IF TXT = 'MAX'
/* THEN S.V(J).VAL2 = VAL
/* ELSE CALL ERREUR (12)
/*
/* IF K+1 > SMAXD
/* THEN CALL ERREUR (13)
/* ELSE
/* DO : K=K+1
/* S.D(K).EQUV = VAL
/* S.D(K).NCAR = N
/* S.D(K).INDC = L
/* IF L+N > SMAXC
/* THEN CALL ERREUR (14)
/* ELSE
/* DO
/* SUBSTR(SC,L,N) = TXT
/* L = L+N
/* END
/* S.V(J).VAL2 = K
/* END
/* ELSE CALL ERREUR (15)
/*
END

```

```

END
/*
/*
/* RANGE LE NOMBRE DE VARIABLES
/* ET LE NOMBRE D'EQUIVALENCES
/* IL FAUT AU MOINS UNE VARIABLE
/*
/* ET CHAQUE VARIABLE DOIT AVOIR
/* AU MOINS UNE DONNÉE, OU UN
/* MINIMUM INFÉRIEUR OU ÉGAL AU
/* MAXIMUM.
/*
/*
/* VÉRIFIER L'UNICITÉ DE TOUS LES
/* LIBELLES (NOMS, CODES ET EQUI-
/* LÉNCES) DANS 'SUBUNIT'
*/

DO I = 1 TO S.NVAR :
DO J = I+1 TO S.NVAR:
IF S.V(I).CODE = S.V(J).CODE | S.V(I).CODE = SUBSTR(SC,S.V(J).INDC,
S.V(J).NCAR) | SUBSTR(SC,S.V(I).INDC,S.V(I).NCAR) = S.V(J).CODE |
SUBSTR(SC,S.V(I).INDC,S.V(I).NCAR) = SUBSTR(SC,S.V(J).INDC,
S.V(J).NCAR) THEN CALL DUPLI(S.V(I).CODE) : END :
DO K = 1 TO S.NEQU :
IF S.V(I).CODE = SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR) | SUBSTR(SC,
S.V(I).INDC,S.V(I).NCAR) = SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR)
THEN CALL DUPLI(S.V(I).CODE) : END :
END :
DO J = 1 TO S.NEQU: DO K = J+1 TO S.NEQU :
IF SUBSTR(SC,S.D(J).INDC,S.D(J).NCAR) = SUBSTR(SC,S.D(K).INDC,
S.D(K).NCAR) THEN CALL DUPLI(SUBSTR(SC,S.D(J).INDC,S.D(J).NCAR)):
END:
END :
/*
/*
/* FIN DU TRAITEMENT DE SUBUNIT
/*
/*
/* VÉRIFIER L'UNICITÉ DES LIBELLES
/* ENTRE UNIT ET SUBUNIT
*/

DO I = 1 TO U.NVAR:
DO J = 1 TO S.NVAR:
IF U.V(I).CODE = S.V(J).CODE | U.V(I).CODE = SUBSTR(SC,S.V(J).INDC,
S.V(J).NCAR) THEN CALL DUPLI(U.V(I).CODE):
IF SUBSTR(UC,U.V(I).INDC,U.V(I).NCAR) = S.V(J).CODE | SUBSTR(UC,
U.V(I).INDC,U.V(I).NCAR) = SUBSTR(SC,S.V(J).INDC,S.V(J).NCAR)

```

```

END;
DO K = 1 TO S.NEQU;
IF U.V(I).CODE = SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR) | SUBSTR(UC,
U.V(I).INDC,U.V(I).NCAR) = SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR)
THEN CALL DUPLI(SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR));
END;
END;
DO J = 1 TO U.NEQU; DO K = 1 TO S.NEQU;
IF SUBSTR(UC,U.D(J).INDC,U.D(J).NCAR) = SUBSTR(SC,S.D(K).INDC,
S.D(K).NCAR) THEN CALL DUPLI(SUBSTR(SC,S.D(K).INDC,S.D(K).NCAR));
END; END;

```

```

PUT EDIT ('U.NEQU = ',U.NEQU) (SKIP,A,F(10));
PUT EDIT ('S.NEQU = ',S.NEQU) (SKIP,A,F(10));

```

```

IF U.V(U.NVAR).TYPE
THEN LU = U.V(U.NVAR).INDC + U.V(U.NVAR).NCAR - 1 ;
ELSE
LU = U.D(U.NEQU).INDC + U.D(U.NEQU).NCAR - 1 ;
PUT EDIT ('LONGUEUR DE UC = ',LU)(SKIP,A,F(6));
IF S.V(S.NVAR).TYPE
THEN LS = S.V(S.NVAR).INDC + S.V(S.NVAR).NCAR - 1 ;
ELSE
LS = S.D(S.NEQU).INDC + S.D(S.NEQU).NCAR - 1 ;
PUT EDIT ('LONGUEUR DE SC = ',LS)(SKIP,A,F(6));

```

```

IF NBERR=0
THEN DO:
OPEN FILE (REPRT) TITLE ('REPRT' || NUMERO);
WRITE FILE (REPRT) FROM (IDENT);
WRITE FILE (REPRT) FROM (U);
WRITE FILE (REPRT) FROM (UC);
WRITE FILE (REPRT) FROM (S);
WRITE FILE (REPRT) FROM (SC);
CLOSE FILE (REPRT);
END;
ELSE PUT EDIT ('NBERR = ',NBERR)(SKIP,A,F(6));
END;

```

```

DGV: PROC OPTIONS(MAIN);
/* SUR VSAM SEN
% DECLARE UMAXV CHARACTER /* NOMBRE DE VARIABLES
      UMAXD CHARACTER /* NOMBRE DE DONNEES OU EQUIVAL.
      UMAXC CHARACTER /* NOMBRE DE CARACTERES
      UMAXB CHARACTER /* NOMBRE DE BITS/QUANTITATIVES
% UMAXV = '300'
% UMAXD = '1500'
% UMAXC = '32752'
% UMAXB = '31'
% DECLARE SMAXV CHARACTER /* NOMBRE DE VARIABLES
      SMAXD CHARACTER /* NOMBRE DE DONNEES OU EQUIVAL.
      SMAXC CHARACTER /* NOMBRE DE CARACTERES
      SMAXB CHARACTER /* NOMBRE DE BITS/QUANTITATIVES
% SMAXV = '35'
% SMAXD = '1500'
% SMAXC = '32752'
% SMAXB = '15'
/* 2 OCTETS
% DCL SMAX CHARACTER /* NOMBRE DE SUBUNITS
% SMAX = '20'
% DCL CTUMAX CHARACTER /* MAX DE COMMENTAIRES DE U
% CTMAX CHARACTER /* MAX + 1
% CTSMAX CHARACTER /* MAX DE COMMENTAIRES DE S
% CTUMAX = '800'
% CTSMAX = '20'
% CTMAX = '801'
% DCL CL CHARACTER /*
% PICTURE CHARACTER /*
% CL = '9999' /* FORME DE LA CLEF NUMERIQUE
% PICTURE = '||CL||'
DCL
1 IDENT STATIC.
2 NUMERO CHAR(1)
2 MISSING BIN(UMAXB) FIXED.
2 CDEL CHAR(1)
2 COMMT CHAR(1);
DECLARE
1 U
2 NVAR BINARY FIXED /* UNIT
  NEOU BINARY FIXED /* NOMBRE DE VARIABLES
  V(UMAXV) /* NOMBRE DE DONNEES OU EQUIVAL.
  CODE CHARACTER(4) /* VARIABLES
  NCARC BIN(7) FIXED /* CODE OU NOM ABREGE
  CHAP BIT /* NOMBRE DE CARACTERES DU NOM AB.
  TYPE BIT /* INDICATEUR DE CHAPITRE
  NCAR BIN(7) FIXED /* TYPE DE LA VARIABLE
  INDC BINARY FIXED /* NOMBRE DE CARACTERES DU NOM
  VAL1 BIN(UMAXB) FIXED /* INDEX DANS LA CHAINE C DU NOM
  VAL2 BIN(UMAXB) FIXED /* INDEX PREMIERE DONNEE OU MINI
  /* INDEX DERNIERE DONNEE OU MAXI
2 D(UMAXD) /* DONNEES OU EQUIVALENCES :
  EQUV BINARY FIXED /* VALEUR NUMERIQUE
  NCAR BIN(7) FIXED /* NOMBRE DE CARACTERES DU TEXTE
  INDC BINARY FIXED /* INDEX DANS LA CHAINE C DU TEXTE
1 UC CHARACTER(UMAXC) /* CHAINE DE CARACTERES POUR LE
  /* STOCKAGE EN CONTINU DES TEXTES
DECLARE
1 S
2 NVAR BINARY FIXED /* SUBUNIT
  NEOU BINARY FIXED /* NOMBRE DE VARIABLES
  V(SMAXV) /* NOMBRE DE DONNEES OU EQUIVAL.
  CODE CHARACTER(4) /* VARIABLES
  NCARC BIN(7) FIXED /* CODE OU NOM ABREGE
  CHAP BIT /* NOMBRE DE CARACTERES DU NOM AB.
  TYPE BIT /* INDICATEUR DE CHAPITRE
  NCAR BIN(7) FIXED /* TYPE DE LA VARIABLE
  INDC BINARY FIXED /* NOMBRE DE CARACTERES DU NOM
  VAL1 BIN(SMAXB) FIXED /* INDEX DANS LA CHAINE C DU NOM
  VAL2 BIN(SMAXB) FIXED /* INDEX PREMIERE DONNEE OU MINI
  /* INDEX DERNIERE DONNEE OU MAXI
2 D(SMAXD) /* DONNEES OU EQUIVALENCES :
  EQUV BINARY FIXED /* VALEUR NUMERIQUE
  NCAR BIN(7) FIXED /* NOMBRE DE CARACTERES DU TEXTE
  INDC BINARY FIXED /* INDEX DANS LA CHAINE C DU TEXTE
1 SC CHARACTER(SMAXC) /* CHAINE DE CARACTERES POUR LE
  /* STOCKAGE EN CONTINU DES TEXTES
DCL MODE CHAR(6) VARYING INIT('IND') STATIC.
1 ENRT STATIC.
2 CLE PIC PICTURE.
2 DU(UMAXV) BIN(UMAXB) FIXED UNALIGNED.
  CTU CHAR(CTUMAX).
2 (SMAX)
2 DS(SMAXV) BIN(SMAXB) FIXED UNALIGNED.
  CTS CHAR(CTSMAX).
/* CT CHAR(CTUMAX) VARYING STATIC. /* CHAINE DE TRAVAIL COMMENTS
/* CT CHAR(CTMAX) VARYING STATIC. /* CHAINE DE TRAVAIL COMMENTS
/* L BIN FIXED STATIC.
/* NU BIN FIXED. /* INDICE DE POSITION DANS CTU
/* NS(SMAX) BIN FIXED. /* INDICE DE POSITION DANS CTS
NO UNIT BIN FIXED STATIC.
NO SUBUNIT BIN FIXED STATIC.
DEL BIT.
CONVER BIT STATIC. /*SI '1'B. IL Y A UNE CONVERSION
QIT BIT.
VAL CHAR(6) VAR.
NB ANNOT BIN(7) FIXED INIT(0) STATIC.
NO U BIN FIXED DEFINED NO UNIT.
NO S BIN FIXED DEFINED NO SUBUNIT.
CART CHAR(80) STATIC.
C CHAR(1).
NC BIN FIXED INIT(0) STATIC.
NO CHAR(1).
TEXT CHAR(56) VAR;

```

DCL REPORT FILE SEQUENTIAL INPUT;

DCL DONNEE FILE
KEYED ENVIRONMENT (VSAM);

DCL CLE PIC PICTURE STATIC;

OPEN FILE (SYSPRINT) PAGESIZE(66);
GET COPY EDIT (NO,TEXT) (X(5),A(1),X(18),A(56));

```

OPEN FILE (REPERT) FILE (REPERT) (1 NO);
READ FILE (REPERT) INTO (IDENT);
IF NO = NUMERO
THEN DO:
  PUT EDIT ('LE DSN NE CORRESPOND PAS A L IDENTIFICATION INTERNE')
  (SKIP,A);
  STOP;
END;
READ FILE (REPERT) INTO (U);
READ FILE (REPERT) INTO (UC);
READ FILE (REPERT) INTO (S);
READ FILE (REPERT) INTO (SC);
CLOSE FILE (REPERT);

OPEN FILE (DONNEE) DIRECT UPDATE;

IF S.V(1).VAL2>SMAX
THEN DO:
  PUT EDIT('MAX DE CLE2 NON CONCORDANT AVEC SMAX. NB DE SUBUNITES')
  (SKIP(2).A);
  STOP;
END;

LECT:PROC(C):
  DCL C CHAR(1);
  NC=NC+1;
  IF NC>80
  THEN DO:
    PUT EDIT (CART) (SKIP,COLUMN(41).A(80));
    GET EDIT (CART) (A(80));
    NC=1;
  END;
  C=SUBSTR(CART,NC,1);
  END;

ON ENDFILE(SYSIN) BEGIN;
  IF NC <= 80 THEN PUT SKIP EDIT(CART)(COL(41).A);
  PUT SKIP EDIT
  ('PAS DE STOP..DERNIER ENRT A CHARGER')(A);
  STOP;
END;

NO U=MISSING;
NO S=MISSING;

  GET EDIT (CART) (A(80));
DEBUT:
  TEXT='';
  DEL='O'B;
  CONVER='O'B;
  DO UNTIL (C~='.'&C~=' ');
  CALL LECT(C);
  END;
  IF C=DEL &MODE='LOCAL'
  THEN DO:
    DEL='1'B;
    DO UNTIL (C~=' ');
    CALL LECT (C);
    END;
    GOTO DECODE;
  ELSE
  IF C=COMMT
  THEN DO:
    DO UNTIL (C~=' ');
    CALL LECT(C);
    END;
    IF NO S=MISSING
    THEN
      IF NO U = MISSING
      THEN CALL COMMENT(CTU.NU.CTUMAX);
      ELSE CALL ANNOT;
      ELSE CALL COMMENT(S(NO_S).CTS.NS(NO_S).CTSMAX);
    END;
  ELSE
  DO:
    DECODE: DO UNTIL (C~='.' | C~=' ');
    TEXT=TEXT || C;
    CALL LECT (C);
    END;
    IF C='.'
    THEN DO:
      IF TEXT='STOP' /***** NON QUANTITATIVE *****/
      THEN DO:
        CALL ECRITURE ENRT;
        IF NC <= 80
        THEN PUT EDIT (CART) (SKIP,COLUMN(41).A(80));
        STOP;
        END;
      ELSE
      IF TEXT='GLOBAL'
      THEN DO:
        CALL ECRITURE ENRT;
        MODE='GLOBAL';
        END;
      ELSE
      IF TEXT='LOCAL'
      THEN DO:
        CALL ECRITURE ENRT;
        MODE='LOCAL';
        END;
      ELSE
      IF NO S=MISSING /* SOUS-UNITE DEFINIE */
      THEN BEGIN;
        DO K=1 TO S.NEQU
          WHILE (SUBSTR(SC.S.D.INDC(K).S.D.NCAR(K)) ~
          TEXT);
        END;
        IF K>S.NEQU
        THEN CALL ANNOT;
        ELSE DO:
          /*RECHERCHE DE L'INDICE DE VARIABLE J */
          DO J=1 TO S.NVAR;
            IF ~S.V(J).TYPE
            THEN
              IF K>S.V(J).VAL1 & K<=S.V(J).VAL2
              THEN LEAVE;
            END;
            IF DEL
            THEN
              IF ENRT.S(NO_S).DS(J)=S.D.EQUV(K)
              THEN ENRT.S(NO_S).DS(J)=MISSING;
              ELSE CALL ANNOT;
            END;
          END;
        END;
      END;
    END;
  END;

```

```

END:
ELSE IF NO U=MISSING /*UNITE NON DEFINIE*/
THEN DO:
IF NC<= 80 THEN PUT EDIT (CART)
(SKIP.COLUMN(41).A(80));
PUT EDIT ('UNITE NON DEFINIE ET SOUS UNITE NON DEFINIE')
(SKIP.A);
STOP:
END:
ELSE BEGIN: /*SOUS-UNITE NON DEFINIE.UNITE DEFINIE*/
DO K=1 TO U.NEQO
WHILE(SUBSTR(U.C.U.D.INDC(K).U.D.NCAR(K)) ~=
TEXT);
END:
IF K>U.NEQO
THEN CALL ANNOT:
ELSE DO:
/*RECHERCHE DE L'INDICE DE VARIABLE J */
DO J=1 TO U.NVAR:
IF ~U.V(J).TYPE
THEN
IF K>=U.V(J).VAL1 & K<=U.V(J).VAL2
THEN LEAVE:
END:
IF ~DEL
THEN
IF ENRT.U.DU(J)=U.D.EQUV(K)
THEN ENRT.U.DU(J)=MISSING:
ELSE CALL ANNOT:
ELSE ENRT.U.DU(J)=U.D.EQUV(K):
END:
END:
END: /* FIN A */
ELSE BEGIN: /* TRAITEMENT B ***** QUANTITATIVE ***** */
VAL='':
N=0:
DO UNTIL (C ~= ' ');
CALL LECT(C):
END:
DO WHILE (C ~= '.' & N<6):
VAL=VAL || C:
N=N+1:
CALL LECT(C):
END:
ON CONVERSION BEGIN:
CONVER='1'B:
CALL ANNOT:
GOTO B:
END:
/***** B *****/
B : OTT='1'B:
IF MODE='IND'
THEN DO:
IF NC <= 80 THEN PUT EDIT (CART)
(SKIP.COLUMN(41).A(80)):
PUT EDIT ('MODE INDEFINI')(SKIP.A):
STOP:
END:
ELSE IF (TEXT=SUBSTR(U.V(1).CODE,1,U.V(1).NCARC) !
TEXT=SUBSTR(UC.U.V(1).INDC,U.V(1).NCAR))
THEN DO:
IF CONVER THEN NB_ANNOT = NB_ANNOT - 1:
CALL ECRITURE_ENRT:
IF CONVER
THEN
DO:
IF NC <= 80
THEN PUT EDIT (CART) (SKIP.COL(41).A(80)):
PUT EDIT ('NO D UNITE INACCEPTABLE') (SKIP.A):
STOP:
END:
ELSE
IF VAL <= U.V(1).VAL2 & VAL >= U.V(1).VAL1
THEN
DO:
NO U = VAL:
IF ~DEL THEN
IF MODE='GLOBAL'
THEN DO:
ENRT.U.DU(1)=VAL:
CALL INIT:
END:
ELSE
DO:
CLE=NO.U:
READ FILE (DONNEE) INTO(ENRT) KEY(CLE):
ENRT.U.DU(1)=VAL:
DO N=CTUMAX TO 1 BY -1 WHILE(SUBSTR(CTU.N,1)=' ');
END:
IF N=0
THEN NU=1:
ELSE NU=N+2:
DO J=1 TO SMAX:
DO N=CTSMAX TO 1 BY -1
WHILE(SUBSTR(S(J).CTS,N,1)=' ');
END:
IF N=0
THEN NS(J)=1:
ELSE NS(J)=N+2:
END:
END:
ELSE DO: CALL INIT: DU(1)=MISSING: END:
END:
ELSE
DO:
CALL ANNOT:
IF NC <= 80 THEN PUT EDIT(CART) (SKIP.COL(41).A):
STOP:
END:
ELSE IF NO U=MISSING
THEN DO:
IF NC <= 80 THEN PUT EDIT (CART)
(SKIP.COLUMN(41).A(80)):
PUT EDIT('UNITE INDEFINIE')(SKIP.A):
STOP:
END:
ELSE IF (TEXT=SUBSTR(S.V(1).CODE,1,S.V(1).NCARC) !
TEXT=SUBSTR(SC.S.V(1).INDC,S.V(1).NCAR))
THEN
DO:
NO S = MISSING:

```

```

IF VAL>=S.V(1).VAL1 & VAL <=S.V(1).VAL2
THEN
  IF -DEL
  THEN DO: NO S=VAL; ENRT.S(V(1).DS(1)=VAL; END;
  ELSE DO I=1 TO SMAXV; S(V(1).DS(I)=MISSING; END;
  ELSE CALL ANNOT;
  ELSE CALL ANNOT;
END;
ELSE
  IF -CONVER
  THEN
    IF NO S=MISSING
    THEN BEGIN: /*RECHERCHE DU NOM VAR DS UNIT */
      /****** C *****/
      DO J=1 TO U.NVAR;
        IF U.V(J).TYPE
        THEN
          IF SUBSTR(UC.U.V(J).INDC,U.V(J).NCAR)
          = TEXT
          | SUBSTR(U.V(J).CODE,1,U.V(J).NCARC)
          = TEXT
          THEN LEAVE;
        END;
        IF J>U.NVAR
        THEN CALL ANNOT;
        ELSE /*ON A TROUVE LA VAR */
          IF VAL>=U.V(J).VAL1&VAL<=U.V(J).VAL2
          THEN
            IF DEL
            THEN
              IF ENRT.U.DU(J)=VAL
              THEN ENRT.U.DU(J)=MISSING;
              ELSE CALL ANNOT;
              ELSE ENRT.U.DU(J)=VAL;
              ELSE CALL ANNOT;
            END;
            /* END BEGIN C */
            /****** D *****/
            ELSE BEGIN: /* NO S DEFINI */
              DO J=1 TO S.NVAR;
                IF S.V(J).TYPE
                THEN
                  IF SUBSTR(SC.S.V(1).INDC(J),S.V(NCAR(J))
                  = TEXT
                  | SUBSTR(S.V(J).CODE,1,S.V(J).NCARC)
                  = TEXT
                  THEN LEAVE;
                END;
                IF J>S.NVAR
                THEN CALL ANNOT;
                ELSE VAL>=S.V(J).VAL1&VAL<=S.V(J).VAL2
                THEN
                  IF DEL
                  THEN
                    IF ENRT.S(NO S).DS(J)=VAL
                    THEN ENRT.S(NO S).DS(J)=MISSING;
                    ELSE CALL ANNOT;
                    ELSE ENRT.S(NO S).DS(J)=VAL;
                    ELSE CALL ANNOT;
                  END;
                  /* FIN DE D */
            END;
            /* FIN DE B */
          END;
        GOTO DEBUT;

```

ECRITURE ENRT: PROCEDURE;

```

  IF NO U = MISSING
  THEN GOTO RAZ;
  IF NB_ANNOT > 0
  THEN
    DO:
      PUT EDIT(NB_ANNOT, ' ANNOTATION(S) ?')
      (SKIP,F(S),A);
      IF MODE='GLOBAL' THEN GOTO RAZ;
    END;
    CLE=NO U;
    REWRITE FILE (DONNEE) FROM (ENRT) KEY (CLE);
    PUT EDIT(' ECRITURE SUR LE FICHIER SORTIE')(SKIP,A);
  RAZ:
    NO U=MISSING;
    NO S=MISSING;
    NB_ANNOT=0;
    END;

```

INIT: PROCEDURE;

```

  DO I=2 TO UMAXV;
    DU(I)=MISSING;
  END;
  DO I=1 TO CTUMAX;
    SUBSTR(CTU.I,1)= ' ';
  END;
  NU=1;

  DO J=1 TO SMAX;
    DO I=1 TO SMAXV;
      S(J).DS(I)=MISSING;
    END;
    DO I=1 TO CTSMAX;
      SUBSTR(S(J).CTS.I,1)= ' ';
    END;
    NS(J)=1;
  END;

```

END;

ANNOT: PROC;

```

  NB_ANNOT=NB_ANNOT+1;
  SUBSTR(CART.NC,1)= '?';
  END;

```

COMMENT: PROC(CTUS,N,M);
 DCL CTUS CHAR(*).
 N BIN FIXED.
 M BIN FIXED;

DEB:

```

  IF N>M
  THEN DO:
    SUBSTR(CART.NC,1)= 'à';
    IF C = 'à'
    THEN DO:
      DO UNTIL (C = '.');
        CALL LECT(C);
      END;
      SUBSTR(CART.NC,1)= 'à';
    END;
  END;
  ELSE DO:
    IF C = 'à'
    THEN DO:
      SUBSTR(CTUS.N,1)=C;

```

```
      N=N+1;  
      CALL LECT(C);  
      GOTO DEB;  
    END;  
  ELSE N=N+1;  
  END;  
END;
```

```
END; /* FIN PROC MAIN */
```

```

EGV2 : PROC OPTIONS(MAIN);
% DECLARE UMAXV CHARACTER
      UMAXD CHARACTER
      UMAXC CHARACTER
      UMAXB CHARACTER
% UMAXV = ' 300'
% UMAXD = ' 1500'
% UMAXC = '32752'
% UMAXB = ' 31'
% DECLARE SMAXV CHARACTER
      SMAXD CHARACTER
      SMAXC CHARACTER
      SMAXB CHARACTER
% SMAXV = ' 35'
% SMAXD = ' 1500'
% SMAXC = '32752'
% SMAXB = ' 15'
% DCL SMAX CHARACTER
      SMAX = ' 20'
% DCL MAXB CHARACTER
      MAXB = '31'
% DCL CTUMAX CHARACTER
      CTSMAX CHARACTER
% CTUMAX = ' 800'
% CTSMAX = ' 20'
% DCL CL CHARACTER
      PICTCLE CHARACTER
% CL = '9999'
% PICTCLE = '|||CL|'
DCL
1 IDENT STATIC.
2 NUMERO CHAR(1).
3 MISSING BIN(MAXB) FIXED.
3 CDEL CHAR(1)
2 COMMT CHAR(1);

DECLARE
1 U
2 NVAR BINARY FIXED
  NEQU BINARY FIXED
2 V(UMAXV)
  CODE CHARACTER(4)
  NCARC BIN(7) FIXED
  CHAP BIT
  TYPE BIT
  NCAR BIN(7) FIXED
  INDC BINARY FIXED
  VAL1 BIN(UMAXB) FIXED
  VAL2 BIN(UMAXB) FIXED
2 D(UMAXD)
  EQUV BINARY FIXED
  NCAR BIN(7) FIXED
  INDC BINARY FIXED
1 UC CHARACTER(UMAXC)

DECLARE
1 S
2 NVAR BINARY FIXED
  NEQU BINARY FIXED
2 V(SMAXV)
  CODE CHARACTER(4)
  NCARC BIN(7) FIXED
  CHAP BIT
  TYPE BIT
  NCAR BIN(7) FIXED
  INDC BINARY FIXED
  VAL1 BIN(SMAXB) FIXED
  VAL2 BIN(SMAXB) FIXED
2 D(SMAXD)
  EQUV BINARY FIXED
  NCAR BIN(7) FIXED
  INDC BINARY FIXED
1 SC CHARACTER(SMAXC)

DCL
1 ENRT STATIC.
2 CLE PIC PICTCLE.
2 U.
3 DU(UMAXV) BIN(UMAXB) FIXED UNALIGNED.
3 CTU CHAR(CTUMAX).
2 S(SMAX).
3 DS(SMAXV) BIN(SMAXB) FIXED UNALIGNED.
3 CTS CHAR(CTSMAX) ; /****/

DCL (NO.NOE.NCLE) CHAR(1);
DCL REPERT FILE RECORD SEQUENTIAL INPUT
  DONNEE FILE KEYED ENVIRONMENT(VSAM) ;

DCL IL BIN FIXED INIT(1) STATIC.
  LIGNE CHAR(132) INIT((132)') STATIC;

DCL CART CHAR(80) STATIC.
  NC BIN FIXED STATIC INIT(0).
  C CHAR(1) STATIC.
  NUM(2500) BIN FIXED INIT((2500)0).
  I BIN FIXED.

  COD BIN FIXED STATIC.
  TXT CHAR(80) VAR STATIC;

LECT : PROC;
  NC=NC+1;
  IF NC>80
  THEN DO;
    PUT EDIT (CART) (SKIP,A);
    GET EDIT (CART) (A(80));
    NC=1;
  END;
  C=SUBSTR(CART,NC,1);
END LECT;

ERREUR : PROC;

```

```

IF NC <= 80
THEN PUT EDIT (CART) (SKIP,A);
ON ENDFILE(SYSIN) STOP;
FIN : GET COPY EDIT (CART) (A(80));
      GOTO FIN;
      END ERREUR;

```

CHAMP : PROC;

```

DCL (I,N) BIN FIXED;
TXT='';
COD=0;

IF C = '.'
THEN COD=1;
ELSE
IF C = 'A'
THEN COD=2;
ELSE
IF C = ' '
THEN COD=3;
ELSE DO:
DO WHILE (C ~= '.' & C ~= 'A' & C ~= ' ');
TXT=TXT || C;
CALL LECT;
END;
IF LENGTH(TXT) > 1
THEN DO:
DO N=LENGTH(TXT) TO 1 BY -1 WHILE (SUBSTR(TXT,N,1)=' ');
END;
TXT=SUBSTR(TXT,1,N);
END;
IF LENGTH(TXT) > 1 & LENGTH(TXT) < 6
THEN DO:
DO I=1 TO N WHILE (SUBSTR(TXT,I,1) >= '0' & SUBSTR(TXT,I,1) <= '9');
END;
IF I < N+1 THEN CALL ERREUR;
ELSE DO:
COD=4;
RETURN;
END;
END;
ELSE
IF LENGTH(TXT)=1
THEN
IF TXT >= '1' & TXT <= '9'
THEN DO:
COD=4;
RETURN;
END;
CALL ERREUR;
END;

IF COD == 3
THEN DO:
DO UNTIL (C ~= ' ');
CALL LECT;
END;
END;

END CHAMP;

EDIT_ENRT : PROC;

```

```

DCL Y BIN FIXED;
T CHAR(6) VAR;
DCL TEXTE CHAR(63) VAR INIT('');

```

```

TEXTE=SUBSTR(UC.U.V(1).INDC.U.V(1).NCAR);
TEXTE=TEXTE || '=';
Y=DU(1);
CALL RECODE( Y,T);
TEXTE=TEXTE || T || '.';

PUT EDIT (TEXTE) (SKIP,A);
DO JL=CTUMAX TO 1 BY -1 WHILE (SUBSTR(CTU,JL,1)=' '); END;
IF JL>0 THEN DO:
PUT EDIT (COMMT) (COLUMN(132),A);
IO = JL / 132;
IF IO>0 THEN
DO I=1 TO IO;
PUT EDIT (SUBSTR(CTU,1+(I-1)*132,132)) (SKIP,A);
END;
IF MOD(JL,132) > 0
THEN PUT EDIT (SUBSTR(CTU,IO*132+1,MOD(JL,132)) || '.' )
(SKIP,A);
ELSE; END;ELSE;

DO J=2 TO U.NVAR;
IF DU(J) ~= MISSING
THEN DO:
IF U.V(J).CHAP | IL=132
THEN CALL IMP;
TEXTE='';
IF U.V(J).TYPE
THEN DO:
TEXTE=SUBSTR(UC.U.V(J).INDC.U.V(J).NCAR) || '=';
Y=DU(J);
CALL RECODE(Y,T);
TEXTE=TEXTE || T || '.';
END;
ELSE DO:
DO K=U.V(J).VAL1 TO U.V(J).VAL2
WHILE (U.D.EQUV(K) ~= DU(J));
END;
TEXTE=SUBSTR(UC.U.D.INDC(K),U.D.NCAR(K)) || '.';
END;
LGT=LENGTH(TEXTE);
DO WHILE (IL+LGT>133);
DO I=133-IL TO 1 BY -1
WHILE (SUBSTR(TEXTE,I,1) ~= ' ' | IL+I>134);
END;
IF I>1
THEN DO:
SUBSTR(LIGNE,IL,I-1)=SUBSTR(TEXTE,1,I-1);
LGT=LGT-I;
TEXTE=SUBSTR(TEXTE,I+1,LGT);
END;
CALL IMP;
END;
SUBSTR(LIGNE,IL,LGT)=TEXTE;
IL=IL+LGT;

```

```

END;
CALL IMP;
PUT SKIP;

DO L=1 TO SMAX;
Y=S(L).DS(1);
IF Y ~= MISSING
THEN
DO;
DO J=1 TO S.NVAR;
IF S(L).DS(J) ~= MISSING
THEN DO;
IF IL=133
THEN CALL IMP;
TEXTE='';
IF S.V(J).TYPE
THEN DO;
TEXTE=SUBSTR(SC.S.V(J).INDC,S.V(J).NCAR) || '=';
Y=S(L).DS(J);
CALL RECODE(Y);
TEXTE=TEXTE || Y || '.';
END;
ELSE DO;
DO K=S.V(J).VAL1 TO S.V(J).VAL2
WHILE (S.D.EQUV(K) ~= S(L).DS(J));
END;
TEXTE=SUBSTR(SC.S.D.INDC(K),S.D.NCAR(K)) || '.';
END;
LGT=LENGTH(TEXTE);
DO WHILE (IL+LGT>133);
DO I=133-IL TO 1 BY -1
WHILE (SUBSTR(TEXTE,I,1) ~= ' ' | IL+I>134);
END;
IF I>1
THEN DO;
SUBSTR(LIGNE,IL,I-1)=SUBSTR(TEXTE,1,I-1);
LGT=LGT-1;
TEXTE=SUBSTR(TEXTE,I+1,LGT);
END;
CALL IMP;
END;
SUBSTR(LIGNE,IL,LGT)=TEXTE;
IL=IL+LGT;
END;
END;
CALL IMP;

DO I=1 TO CTSMAX WHILE (SUBSTR(S(L).CTS,I,1) ~= ' ');
END;
IF I < CTSMAX + 1
THEN
PUT EDIT (COMMT || S(L).CTS || '.'
(SKIP,A);
END;
END;
PUT SKIP EDIT ((132) '-' )(A);
END EDIT ENRT;

RECODE : PROC(X,T);
DCL T CHAR(6) VAR;
X BIN(MAXB) FIXED;
N5 PIC'S99999';

IF X=0 THEN T='O';
ELSE DO;
N5=X;
T=N5;
DO K=2 TO 6 WHILE(SUBSTR(T,K,1)='O');
END;
IF SUBSTR(T,1,1)='- '
THEN T='-' || SUBSTR(T,K,7-K);
ELSE
T=SUBSTR(T,K,7-K);
END;
END;

IMP : PROC;
PUT EDIT (LIGNE) (SKIP,A);
IL=1;
LIGNE=(132) ' ';
END;

OPEN FILE (SYSPRINT) PAGESIZE(66);

GET COPY EDIT (NO,NOE,NCLE,CDEL) (3(X(5),A(1)),X(61),A(1));
OPEN FILE (REPRT) TITLE ('REPRT' || NO);
READ FILE (REPRT) INTO (IDENT);
READ FILE (REPRT) INTO (U);
READ FILE (REPRT) INTO (UC);
READ FILE (REPRT) INTO (S);
READ FILE (REPRT) INTO (SC);
CLOSE FILE (REPRT);

IF NCLE ~= ' '
THEN DO;

I=0;
INCR=0;
GET EDIT (CART) (A(80));
DO UNTIL (C ~= ',');
CALL LECT;
END;

DEBUT : CALL CHAMP;
IF COD ~= 4 THEN CALL ERREUR;
ELSE
IF INCR=0
THEN DO;
I=I+1;
NUM(I)=TXT;
END;
ELSE DO;
DO K=1 TO TXT-NUM(I);
NUM(I+K)=NUM(I)+K;
END;
I=I+K-1;
INCR=0;
END;
CALL CHAMP;
IF COD=1
THEN GOTO DEBUT;
ELSE
IF COD=2
THEN DO;
INCR=1;
GOTO DEBUT;

```

```
ELSE  
  IF COD = 3  
    THEN CALL ERREUR;
```

```
SUITE : M=I:  
  IF NC <= 80 THEN PUT EDIT (CART) (SKIP,A(80));
```

```
OPEN FILE (DONNEE) TITLE ('DATA' || NOE) DIRECT INPUT;  
DO IC=1 TO M:  
  CLE=NUM(IC);  
  READ FILE (DONNEE) INTO (ENRT) KEY(CLE);  
  IF DU(1) = MISSING  
    THEN CALL EDIT ENRT;  
END;  
END;
```

```
ELSE DO:  
  OPEN FILE (DONNEE) TITLE ('DATA' || NOE) SEQUENTIAL INPUT;  
  ON ENDFILE(DONNEE) STOP;  
BOUCLE : READ FILE (DONNEE) INTO (ENRT);  
IF DU(1) = MISSING  
  THEN CALL EDIT ENRT;  
  GOTO BOUCLE;  
END;  
END;
```

```

SGV : PROC OPTIONS(MAIN);
/* SELECTION SUR VSAM SEN */
/*
% DECLARE UMAXV CHARACTER /*NOMBRE DE VARIABLES
UMAXD CHARACTER /*NOMBRE DE DONNEES OU EQUIVAL.
UMAXC CHARACTER /*NOMBRE DE CARACTERES
UMAXB CHARACTER /*NOMBRE DE BITS/QUANTITATIVES
/*
% UMAXV = ' 300'
UMAXD = ' 1500'
UMAXC = '32752'
UMAXB = ' 31'
/*BLKSIZE=32760.RECFM=VBST
/* 4 OCTETS
/*
% DECLARE SMAXV CHARACTER /*NOMBRE DE VARIABLES
SMAXD CHARACTER /*NOMBRE DE DONNEES OU EQUIVAL.
SMAXC CHARACTER /*NOMBRE DE CARACTERES
SMAXB CHARACTER /*NOMBRE DE BITS/QUANTITATIVES
/*
% SMAXV = ' 35'
SMAXD = ' 1500'
SMAXC = '32752'
SMAXB = ' 15'
/* 2 OCTETS
/*
% DECLARE SMAX CHARACTER /*NOMBRE DE SUBUNITS
SMAX = ' 20'
/*
% DECLARE MAXB CHARACTER /*
MAXB = '31' /* MAX(UMAXB,SMAXB)
/*
% DCL CTUMAX CHARACTER /*MAX DE COMMENTAIRES DE U
CTSMAX CHARACTER /*MAX DE COMMENTAIRES DE S
/*
% CTUMAX = ' 800'
CTSMAX = ' 20'
/*
% DCL CL CHARACTER /*
PICTCLE CHARACTER /*
CL = '9999' /* FORME DE LA CLEF NUMERIQUE
PICTCLE = '||CL||' /*
/*
% DCL NBOPR CHARACTER /*NOMBRE D'OPERANDES BOOLEENES N
MAX CHARACTER /*DIMENSION DE LA PILE
NBOPR = ' 200' /* NBOPR = N*SMAX
MAX = '1200' /*DIMENSION DE LA PILE NBOPR*6
/*
DCL REPERT FILE RECORD SEQUENTIAL INPUT;
DCL
1 IDENT STATIC.
NUMERO CHAR(1).
MISSING BIN(MAXB) FIXED.
CODEL CHAR(1)
COMMT CHAR(1);
/*
DECLARE
1 U /*UNIT
NVAR BINARY FIXED /*NOMBRE DE VARIABLES
NEOU BINARY FIXED /*NOMBRE DE DONNEES OU EQUIVAL.
V(UMAXV) /*VARIABLES
CODE CHARACTER(4) /*CODE OU NOM ABREGE
NCARC BIN(7) FIXED /*NOMBRE DE CARACTERES DU NOM AB.
NCAP BIT /*INDICATEUR DE CHAPITRE
TYPE /*TYPE DE LA VARIABLE
NCAR BIN(7) FIXED /*NOMBRE DE CARACTERES DU NOM
INDC BINARY FIXED /*INDEX DANS LA CHAINE C DU NOM
VAL1 BIN(UMAXB) FIXED /*INDEX PREMIERE DONNEE OU MINI
VAL2 BIN(UMAXB) FIXED /*INDEX DERNIERE DONNEE OU MAXI
2 D(UMAXD) /*DONNEES OU EQUIVALENCES
EQUV BINARY FIXED /*VALEUR NUMERIQUE
NCAR BIN(7) FIXED /*NOMBRE DE CARACTERES DU TEXTE
INDC BINARY FIXED /*INDEX DANS LA CHAINE C DU TEXTE
1 UC CHARACTER(UMAXC) /*CHAINE DE CARACTERES POUR LE
/*STOCKAGE EN CONTINU DES TEXTES
/*
DECLARE
1 U /*SUBUNIT
NVAR BINARY FIXED /*NOMBRE DE VARIABLES
NEOU BINARY FIXED /*NOMBRE DE DONNEES OU EQUIVAL.
V(SMAXV) /*VARIABLES
CODE CHARACTER(4) /*CODE OU NOM ABREGE
NCARC BIN(7) FIXED /*NOMBRE DE CARACTERES DU NOM AB.
NCAP BIT /*INDICATEUR DE CHAPITRE
TYPE /*TYPE DE LA VARIABLE
NCAR BIN(7) FIXED /*NOMBRE DE CARACTERES DU NOM
INDC BINARY FIXED /*INDEX DANS LA CHAINE C DU NOM
VAL1 BIN(SMAXB) FIXED /*INDEX PREMIERE DONNEE OU MINI
VAL2 BIN(SMAXB) FIXED /*INDEX DERNIERE DONNEE OU MAXI
2 D(SMAXD) /*DONNEES OU EQUIVALENCES
EQUV BINARY FIXED /*VALEUR NUMERIQUE
NCAR BIN(7) FIXED /*NOMBRE DE CARACTERES DU TEXTE
INDC BINARY FIXED /*INDEX DANS LA CHAINE C DU TEXTE
1 SC CHARACTER(SMAXC) /*CHAINE DE CARACTERES POUR LE
/*STOCKAGE EN CONTINU DES TEXTES
/*
DCL ENTREE FILE KEYED ENVIRONMENT (VSAM).
CLE PIC PICTCLE STATIC;
DCL
1 ENRT STATIC.
2 CLE PIC PICTCLE.
3 U.
4 DU(UMAXV) BIN(UMAXB) FIXED UNALIGNED.
5 CTU CHAR(CTUMAX).
2 (SMAX).
3 DS(SMAXV) BIN(SMAXB) FIXED UNALIGNED.
4 CTS CHAR(CTSMAX) : /****/
DCL NC BIN FIXED INIT(0) STATIC.
CART CHAR(80) STATIC.
CAR CHAR(1) STATIC.
(NO.NOE.NOS) CHAR(i).
1 CH STATIC.
CODE BIN(7) FIXED.
TEXT CHAR(56).
NUMV BIN FIXED.
IND1 BIN(7) FIXED.
IND2 BIN(MAXB) FIXED.
2 TYPE BIN(7) FIXED;
DCL CD LIKE CH STATIC.
CG LIKE CH STATIC;
DCL A(MAX) BIN(MAXB) FIXED STATIC.
ETAT BIN(7) FIXED.
B(NBOPR) BIT STATIC.
RESULTAT BIT STATIC;

```

DCL (I,M) BIN FIXED;

DCL SORTIE FILE KEYED ENVIRONMENT(VSAM):

```
LECT:PROC(C):
  DCL C CHAR(1);
  NC=NC+1;

  IF NC>80
  THEN DO:
    PUT EDIT (CART) (SKIP,A(80));
    GET EDIT (CART) (A(80));
    NC=1;
  END;
  C=SUBSTR(CART,NC,1);
END;
```

/***** DEBUT *****/

```
GET COPY EDIT (NO.NOE.NOS.CAR) (3(X(5),A(1)),X(61),A(1));
OPEN FILE (REPRT) TITLE ('REPRT' || NO);
OPEN FILE (ENTREE) SEQUENTIAL INPUT TITLE ('DATA' || NOE);
IF NOS = ' '
THEN OPEN FILE (SORTIE) SEQUENTIAL OUTPUT TITLE ('DATA' || NOS);

READ FILE (REPRT) INTO (IDENT);
READ FILE (REPRT) INTO (U);
READ FILE (REPRT) INTO (UC);
READ FILE (REPRT) INTO (S);
READ FILE (REPRT) INTO (SC);
CLOSE FILE (REPRT);
```

```
GET EDIT (CART) (A(80));
DO UNTIL (CAR=' ');
CALL LECT(CAR);
END;
CALL DECOD FORMULE;
IF NC <= 80 THEN PUT EDIT (CART) (SKIP,A(80));
```

```
M=0; MS=0;
ON ENDFILE (ENTREE) BEGIN:
  PUT SKIP;
  PUT EDIT (M,' ENREGISTREMENTS ONT ETE LUS'
            MS,' ENREGISTREMENTS ONT ETE EXTRAITS ');
  IF NOS = ' ' THEN CLOSE FILE (SORTIE);
  STOP;
END;

PUT SKIP;
BOUCLE : READ FILE (ENTREE) INTO (ENRT);
IF U.DU(1) = MISSING
THEN DO:
  M=M+1;
  CALL EXEC FORMULE;
  IF RESULTAT
  THEN DO:
    PUT EDIT (ENRT,U.DU(1)) (F(6));
    CLE=ENRT,U.DU(1);
    IF NOS = ' '
    THEN WRITE FILE (SORTIE) FROM (ENRT) KEYFROM(CLE);
    MS=MS+1;
  END;
END;
GOTO BOUCLE;
```

DECODAGE_CHAMP : PROCEDURE;

```
DCL C(9) CHAR(1) INIT(' ','<','>','&','|','.')'.','>');
TXT CHAR(160) VAR INIT(' ');
SUB CHAR(6) VAR INIT('');
```

```
CH.CODE=0;
CH.TEXT=(56) ' ';
CH.NUMV=0;
CH.IND1=0;
CH.IND2=0;
CH.TYPE=0;

DO L=1 TO 9;
IF CAR=C(L)
THEN LEAVE;
END;
IF L<=9
THEN DO:
  IF L<=8
  THEN CH.CODE=L;
  ELSE CH.CODE=11;
  IF CH.CODE = 7
  THEN
    DO UNTIL (CAR=' ');
    CALL LECT(CAR);
    END;
  END;
ELSE
DO:
  N=0;
  WHILE (CAR=C(1)&CAR=C(2)&CAR=C(3)&CAR=C(4)&
        CAR=C(5)&CAR=C(6)&CAR=C(7)&CAR=C(8)&CAR=C(9));
  TXT=TXT || CAR;
  N=N+1;
  CALL LECT(CAR);
  END;
  IF N = 1
  THEN
    IF TXT < '0' | TXT > '9'
    THEN GOTO ALPHA_N;
  ELSE;
  ELSE DO:
    DO K=N TO 1 BY -1
    WHILE (SUBSTR(TXT,K,1)=' ');
    END;
    TXT=SUBSTR(TXT,1,K);
    IF K = 1
    THEN
      IF TXT < '0' | TXT > '9'
      THEN GOTO ALPHA_N;
    ELSE;
    ELSE IF SUBSTR(TXT,1,1) = '-' &
      (SUBSTR(TXT,1,1) < '0' &
```



```

ELSE
  IF ETAT=0
  THEN ETAT=1;
ELSE
  IF ETAT=-1
  THEN DO:
    PUT EDIT('EXPRESSION MIXTE')
      (SKIP,A);
    CALL ERREUR;
  END;
END;
IF LGT <= 4
THEN DO:
  DO JV=1 TO S.NVAR
  WHILE (LGT ~= S.V(JV).NCARC |
    SUBSTR(S.V(JV).CODE,1,LGT) ~= TXT);
  END;
  IF JV<=S.NVAR
  THEN DO:
    CH.NUMV=JV;
    CH.IND2=JV;
    CH.TYPE=4;
    RETURN;
  END;
END;
DO JV=1 TO S.NVAR
WHILE
  (SUBSTR(SC,S.V(JV).INDC,S.V(JV).NCAR)~=TXT);
END;
IF JV<=S.NVAR
THEN DO:
  CH.NUMV=JV;
  CH.IND2=JV;
  CH.TYPE=4;
  END;
ELSE DO:
  PUT EDIT ('VARIABLE OU DONNEE INCONNUE')
    (SKIP,A);
  CALL ERREUR;
  END;
END;
END;
RETURN;
END;

```

```

ERREUR: PROC:
  SUBSTR(CART.NC,1)='7';
  IF NC<=80
  THEN PUT EDIT (CART) (SKIP,A(80));
  ON ENDFILE(SYSIN) STOP;
  FIN: GET COPY EDIT (CART) (A(80));
  GOTO FIN;
  END ERREUR;

```

```

ERR : PROC;
PUT EDIT ('ERREUR DE COMPARAISON') (SKIP,A);
CALL ERREUR;
END;

```

```

RANGT_CODE : PROC(I,J);
  DCL (I,J) BIN FIXED;
  J=J-1;
  IF J <= 1
  THEN DO:
    PUT EDIT ('FORMULE DE SELECTION TROP LONGUE') (SKIP,A);
    CALL ERREUR;
  END;
  ELSE A(J) = CH.CODE;
  END;

```

```

TRANSF_A : PROC(I,J);
  DCL (I,J) BIN FIXED;
  DO WHILE (CH.CODE >= A(J));
  I=I+1;
  A(I) = A(J);
  J=J+1;
  END;
  IF CH.CODE =6
  THEN
    IF A(J) ~= 8
    THEN DO:
      PUT EDIT ('ERREUR SUR NOMBRE DE PARENTHESES') (SKIP,A);
      CALL ERREUR;
    END;
  ELSE J=J+1;
  END;

```

```

RELATION_X_OP_Y : PROC(I,J);
DCL OP BIN FIXED;
DCL (I,J) BIN FIXED;
CG=CH;
CALL DECODAGE_CHAMP;
IF (CH.CODE ~= 1 & CH.CODE ~= 2 & CH.CODE ~= 11)
THEN DO:
  PUT EDIT('ERREUR D OPERATEUR') (SKIP,A);
  CALL ERREUR;
END;
ELSE DO:
  OP=CH.CODE;
  CALL DECODAGE_CHAMP;
  IF (CH.CODE ~= 12 & CH.CODE ~= 13)
  THEN DO:
    PUT EDIT ('ERREUR DE SYNTAXE') (SKIP,A);
    CALL ERREUR;
  END;
  ELSE DO:
    CG = CH;
    IF OP = 11
    THEN DO:
      OP=2;
      CG=CG;
      CG=CH;
    END;
    IF CG.TYPE =1
    THEN
      IF CG.TYPE ~= 3
      THEN CALL ERR;

```

```

ELSE
  IF CD.TYPE = 2
  THEN
    IF CG.TYPE = 4
    THEN CALL ERR;
  ELSE
    IF CG.TYPE = 1
    THEN
      IF CD.TYPE = 3
      THEN CALL ERR;
    ELSE
      IF CG.TYPE = 2
      THEN
        IF CD.TYPE = 4
        THEN CALL ERR;
      ELSE
        IF CD.TYPE = 0 & CD.IND1 = 126
        THEN
          IF CG.TYPE = 3 & CG.TYPE = 4
          THEN CALL ERR;
        ELSE
          IF CG.TYPE = 0 & CG.IND1 = 126
          THEN
            IF CD.TYPE = 3 & CD.TYPE = 4
            THEN CALL ERR;
          ELSE
            IF CG.TYPE=4 & CD.TYPE=4
            THEN
              IF CG.IND2=CD.IND2
              | CG.IND1=0 | CD.IND1=0 | CG.IND1>SMAX
              | CD.IND1>SMAX
              | CG.IND1=CD.IND1
              THEN CALL ERR;
            ELSE
              SUITE: CH.CODE = 8;
              CALL RANGT_CODE(I,J);
              IF I + 5 >= J
              THEN DO:
                PUT EDIT ('FORMULE DE SELECTION TROP LONGUE') (SKIP,A);
                CALL ERREUR;
              END;
              ELSE DO:
                A(I+1) = OP;
                A(I+2) = CG.IND1;
                A(I+3) = CG.IND2;
                A(I+4) = CD.IND1;
                A(I+5) = CD.IND2;
                I = I + 5;
                CH.CODE = 6;
                CALL TRANSF_A(I,J);
              END;
            END;
          END;
        END;
      DECOD FORMULE : PROC;

      I=0;
      J= MAX;
      A(J)=10;
      ETAT=0;
      DEBUT: CALL DECODAGE_CHAMP;
      IF CH.CODE = 8 | CH.CODE = 3
      THEN DO:
        CALL RANGT_CODE(I,J);
        GOTO DEBUT;
      END;
      ELSE
        IF CH.CODE = 12 | CH.CODE = 13
        THEN DO:
          CALL RELATION X OF Y(I,J);
          CALL DECODAGE_CHAMP;
          IF CH.CODE = 7
          THEN
            IF CH.CODE = 6
            THEN DO:
              CALL TRANSF_A(I,J);
              GOTO E3;
            END;
          ELSE
            IF CH.CODE = 4 | CH.CODE = 5
            THEN DO:
              CALL TRANSF_A(I,J);
              CALL RANGT_CODE(I,J);
              GOTO DEBUT;
            END;
          ELSE DO:
            PUT EDIT ('ERREUR DE SYNTAXE') (SKIP,A);
            CALL ERREUR;
          END;
        ELSE DO:
          CALL TRANSF_A(I,J);
          IF A(J) = 10
          THEN DO:
            CALL REPRO;
            A(I+1)=6;
          END;
          ELSE DO:
            PUT EDIT ('ERREUR DE SYNTAXE') (SKIP,A);
            CALL ERREUR;
          END;
        END;
      E4: END;
      ELSE
        IF CH.CODE = 7
        THEN GOTO E4;
      ELSE DO:
        PUT EDIT ('ERREUR LEXICALE') (SKIP,A);
        CALL ERREUR;
      END;
    END;
  END;
  REPRO : PROC;
  DCL X BIN(SMAXB) FIXED.
  (K,J) BIN FIXED;
  IF ETAT=2
  THEN DO:
    DO K=1 TO SMAX;
      DO J=1 TO I;
        X=A(J+(K-1)*I);

```

```

      IF A=127
      THEN A(J+(K-1)*I)=K;
      IF K = SMAX
      THEN A(J+K*I)=X;
      END;
    END;
    I=SMAX+1;
    DO J=1 TO SMAX-1;
      I=I+1;
      A(I)=5;
    END;
  END;
END;

EXEC FORMULE : PROC;
F: PROC(X,Y) RETURNS (BIN(MAXB) FIXED);
DCL X BINARY FIXED;
Y BIN(MAXB) FIXED;
IF X=0
THEN RETURN(ENRT.U.DU(Y));
ELSE
  IF X = 126
  THEN RETURN(Y);
  ELSE RETURN(ENRT.S(X).DS(Y));
END;
G: PROC(P,I,Q) RETURNS(BIT(1));
DCL P BIN(MAXB) FIXED;
I BINARY FIXED;
Q BIN(MAXB) FIXED;
IF A(I) = 1
THEN
  IF P = Q
  THEN RETURN('1'B);
  ELSE RETURN('0'B);
ELSE
  IF A(I) = 2
  THEN
    IF P < Q
    THEN RETURN('1'B);
    ELSE RETURN('0'B);
  END;
DCL I BINARY FIXED;
P BIN(MAXB) FIXED;
Q BIN(MAXB) FIXED;
J BINARY FIXED;
I=0;
J=0;
B(1)='1'B;
BOUCLE : I= I + 1;
          IF A(I)=1 | A(I)=2
          THEN DO;
            P = F(A(I+1),A(I+2));
            Q = F(A(I+3),A(I+4));
            J = J + 1;
            B(J) = G(P,I,Q);
            I = I + 4;
          END;
          ELSE
            IF A(I)=3
            THEN B(J)=-B(J);
            ELSE
              IF A(I)=4
              THEN DO: B(J-1) = B(J-1)&B(J); J = J-1; END;
              ELSE
                IF A(I)=5
                THEN DO: B(J-1) = B(J-1) | B(J); J = J-1; END;
                ELSE
                  IF A(I)=6
                  THEN GOTO FIN;
                END;
              GOTO BOUCLE;
            END;
          GOTO BOUCLE;
          RESULTAT=B(1);
        IN :
      END;

      END SGV;

```

```

PP : PROC OPTIONS(MAIN);

/* SENEGAL
/* NOMBRE DE VARIABLES
/* NOMBRE DE DONNEES OU EQUIVAL.
/* NOMBRE DE CARACTERES
/* NOMBRE DE BITS/QUANTITATIVES
% DECLARE UMAXV CHARACTER
% UMAXD CHARACTER
% UMAXC CHARACTER
% UMAXB CHARACTER
% UMAXV = '300'
% UMAXD = '1500'
% UMAXC = '32752'
% UMAXB = '31'
% DECLARE SMAXV CHARACTER
% SMAXD CHARACTER
% SMAXC CHARACTER
% SMAXB CHARACTER
% SMAXV = '35'
% SMAXD = '1500'
% SMAXC = '32752'
% SMAXB = '15'
% DCL SMAX CHARACTER
% SMAX = '20'
% DCL MAXB CHARACTER
% MAXB = '31'
% DCL CTUMAX CHARACTER
% CTSMAX CHARACTER
% CTUMAX = '800'
% CTSMAX = '20'
% DCL CL CHARACTER
% PICTCLE CHARACTER
% CL = '9999'
% PICTCLE = '||CL||'
/* NOMBRE DE SUBUNITS
/*
/* MAX(UMAXB,SMAXB)
/*
/* MAX DE COMMENTAIRES DE U
/* MAX DE COMMENTAIRES DE S
% DCL COD1(2) CHAR(1) INIT (' ',' ',' ',' ',' ',' ',' ',' ',' ')
% DCL COD2(36) CHAR(1) INIT ('A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z',' ');
DCL
1 IDENT STATIC.
2 NUMERO CHAR(1)
2 MISSING BIN(MAXB) FIXED.
2 CDEL CHAR(1)
2 COMMT CHAR(1);
DECLARE
1 U
2 NVAR BINARY FIXED
2 NEQU BINARY FIXED
2 V(UMAXV)
3 CODE CHARACTER(4)
3 NCARC BIN(7) FIXED
3 CHAP BIT
3 TYPE BIT
3 NCAR BIN(7) FIXED
3 INDC BINARY FIXED
3 VAL1 BIN(UMAXB) FIXED
3 VAL2 BIN(UMAXB) FIXED
2 D(UMAXD)
3 EQUIV BINARY FIXED
3 NCAR BIN(7) FIXED
3 INDC BINARY FIXED
1 UC CHARACTER(UMAXC)
/* UNIT
/* NOMBRE DE VARIABLES
/* NOMBRE DE DONNEES OU EQUIVAL.
/* VARIABLES :
/* CODE OU NOM ABREGE
/* NOMBRE DE CARACTERES DU NOM AB.
/* INDICATEUR DE CHAPITRE
/* TYPE DE LA VARIABLE
/* NOMBRE DE CARACTERES DU NOM
/* INDEX DANS LA CHAINE C DU NOM
/* INDEX PREMIERE DONNEE OU MINI
/* INDEX DERNIERE DONNEE OU MAXI
/* DONNEES OU EQUIVALENCES :
/* VALEUR NUMERIQUE
/* NOMBRE DE CARACTERES DU TEXTE
/* INDEX DANS LA CHAINE C DU TEXTE
/* CHAINE DE CARACTERES POUR LE
/* STOCKAGE EN CONTINU DES TEXTES
DECLARE
1 S
2 NVAR BINARY FIXED
2 NEQU BINARY FIXED
2 V(SMAXV)
3 CODE CHARACTER(4)
3 NCARC BIN(7) FIXED
3 CHAP BIT
3 TYPE BIT
3 NCAR BIN(7) FIXED
3 INDC BINARY FIXED
3 VAL1 BIN(SMAXB) FIXED
3 VAL2 BIN(SMAXB) FIXED
2 D(SMAXD)
3 EQUIV BINARY FIXED
3 NCAR BIN(7) FIXED
3 INDC BINARY FIXED
1 SC CHARACTER(SMAXC)
/* SUBUNIT
/* NOMBRE DE VARIABLES
/* NOMBRE DE DONNEES OU EQUIVAL.
/* VARIABLES :
/* CODE OU NOM ABREGE
/* NOMBRE DE CARACTERES DU NOM AB.
/* INDICATEUR DE CHAPITRE
/* TYPE DE LA VARIABLE
/* NOMBRE DE CARACTERES DU NOM
/* INDEX DANS LA CHAINE C DU NOM
/* INDEX PREMIERE DONNEE OU MINI
/* INDEX DERNIERE DONNEE OU MAXI
/* DONNEES OU EQUIVALENCES :
/* VALEUR NUMERIQUE
/* NOMBRE DE CARACTERES DU TEXTE
/* INDEX DANS LA CHAINE C DU TEXTE
/* CHAINE DE CARACTERES POUR LE
/* STOCKAGE EN CONTINU DES TEXTES
DCL
1 ENRT STATIC
2 CLE PIC PICTCLE.
2 U
3 DU(UMAXV) BIN(UMAXB) FIXED UNALIGNED.
3 CTU CHAR(CTUMAX).
2 S(SMAX)
3 DS(SMAXV) BIN(SMAXB) FIXED UNALIGNED.
3 CTS CHAR(CTSMAX) ; /****/
DCL UNOM(UMAXV) CHAR(4) DEFINED U.V.CODE.
DCL SNOM(SMAXV) CHAR(4) DEFINED S.V.CODE;
DCL DOWNE FILE KEYED ENVIRONMENT(VSAM);
DCL REPERT FILE RECORD SEQUENTIAL INPUT;
DCL (NOR.NOD) CHAR(1);
DCL (LMIN.LMAX) CHAR(3);
DCL CLE PIC PICTCLE;
DCL PU(100.UMAXV) BIN(7) FIXED UNALIGNED.
DCL PS(100.SMAXV) BIN(7) FIXED UNALIGNED.
DCL P BIN(7) FIXED;
DCL (SLI(100).SL) BIN FIXED.
DCL PLI(100) PIC ZZZ9'.
DCL CLI(100) CHAR(1).
DCL (ST.STT) BIN(31) FIXED;
DCL SCO BIN(31) FIXED;
DCL SCOL(UMAXV+SMAXV) BIN(31) FIXED ;
DCL C CHAR(100)
DCL CH(10) CHAR(1) INIT ('1','2','3','4','5','6','7','8','9','0').
DCL C1(500) CHAR(1).
DCL C2(500) CHAR(1).
DCL C5 CHAR(100) INIT ((100)'-').
DCL C4(10) CHAR(10);

```

```

R BIN FIXED:
OPEN FILE (SYSPRINT) PAGESIZE(66):
GET COPY EDIT (NOR,NOD,LMIN,LMAX) (2(X(5),A(1)),2(X(3),A(3))):

Q = (LMAX + 1)/10:
R = LMAX+1 -Q*10:

DO I = 1 TO 10:
  C2(I) = '':
  C4(I) = '1234567890':
END:
DO J = 1 TO Q+1:
  DO K = 1 TO 10:
    DO I = 1 TO 11:
      C2(I-1 + K*11 + (J-1)*110) = CH(K):
    END:
  END:
END:

DO I = 1 TO 109:
  C1(I) = '':
END:
DO K = 1 TO Q:
  DO M = 1 TO 110:
    C1(109+M+(K-1)*110) = CH(K):
  END:
END:

OPEN FILE (REPRT) TITLE ('REPRT'|(NOR):
OPEN FILE (DONNEE) TITLE ('DATA'|(NOD) SEQUENTIAL INPUT:

READ FILE (REPRT) INTO (IDENT):
READ FILE (REPRT) INTO (U):
READ FILE (REPRT) INTO (UC):
READ FILE (REPRT) INTO (S):
READ FILE (REPRT) INTO (SC):
CLOSE FILE (REPRT):

IF LMIN -> 0
THEN
DO:
  CLE = LMIN*100:
  READ FILE (DONNEE) INTO (ENRT) KEY (CLE):
  END:

STT = 0:

DO I = 1 TO U.NVAR + S.NVAR:
  SCOL(I)=0:
END:

IMP = LMIN*11 + 1:

DO LIM=LMIN+1 TO LMAX+1:
  C = (100) ' ':
  ST = 0:

  DO K=1 TO 100:
    DO I=1 TO U.NVAR:
      PU(K,I)=0:
    END:
    DO I=1 TO S.NVAR:
      PS(K,I)=0:
    END:
  END:

  DO K = 1 TO 100:
    SL=0:
    CLI(K)=' ':
    PLI(K)=0:

    ON ENDFILE(DONNEE) GOTO PR:

    READ FILE (DONNEE) INTO (ENRT):

    DO I = 1 TO U.NVAR:
      P = 0:
      IF DU(I) -> MISSING
      THEN P = 1:
      ELSE:
        PU(K,I) = P:
        SL = SL + P:
      END:

    DO I = 1 TO S.NVAR:
      P = 0:
      DO J = 1 TO SMAX:
        IF S(J).DS(I) -> MISSING
        THEN P = P+1:
        ELSE:
          PS(K,I) = P:
          SL = SL + P:
        END:
      END:
      SLI(K) = SL:
      ST = ST + SLI(K):
    END:

  PR: STT = STT + ST:

  /*****
  IMPRESSION DE 100 ENRTS */

  PUT PAGE:
  IMP1 = IMP:
  IMP0 = IMP + 10:
  PUT EDIT ((C1(N) DO N = IMP1 TO IMP0).
    (C2(N) DO N = IMP1 TO IMP0).
    CH(10). (CH(M) DO M = 1 TO 10).
    (C4(M) DO M = 1 TO 10).
    C5)
    (SKIP(3).
    3 (SKIP(1).COLUMN(11).A(1).X(8).9(A(1).X(9)).A(1)).
    SKIP(1).COLUMN(11).TO A(10).
    SKIP(1).COLUMN(11).A(100)):

  DO I = 1 TO U.NVAR:
    SCO = 0:
    DO K = 1 TO 100:
      L = PU(K,I):
      SCO = SCO + L:
      SUBSTR( C,K,1) = COD1(L+1):
    END:

    SCOL(I) = SCOL(I) + SCO:
    PUT EDIT (I.UNOM(I) 'I' C 'I' UNOM(I).SCO)
    (SKIP(1).F(3).X(1).A(4).X(1).A.A.A.X(1).A(4).F(5)):
    IF LIM = LMAX+1
    THEN PUT EDIT (SCOL(I)) (SKIP(0).COL(121).F(10)):

```

```

END;

DO I = 1 TO S.NVAR:                /*VARIABLES DE S*/
  SCO = 0;
  DO K = 1 TO 100:
    L = PS(K,I);
    SCO = SCO + L;
    SUBSTR( C,K,1) = COD2(L+1);
  END;

  J = I + U.NVAR;
  SCOL(J) = SCOL(J) + SCO;
  PUT EDIT (I,SNOM(I),I,C,I,SNOM(I),SCO)
    (SKIP(1),F(3),X(1),A(4),
    X(1),A,A(100),A,X(1),A(4),F(5));

  IF LIM = LMAX + 1
  THEN PUT EDIT (SCOL(J)) (SKIP(0),COL(121),F(10));
END;                                /*IMPRESSION CADRE INFERIEUR*/
PUT EDIT (C5
  (C1(N) DO N = IMP1 TO IMP0),
  (C2(N) DO N = IMP1 TO IMP0),
  CH(10), (CH(M) DO M = 1 TO 10),
  (C4(M) DO M = 1 TO 10))
  3 (SKIP(1),COLUMN(11),A(100),
  (SKIP(1),COLUMN(11),A(1),X(8),9(A(1),X(9)),A(1)),
  SKIP(1),COLUMN(11),10 A(10));

PUT SKIP EDIT (ST) (COL(121),F(10));
PUT SKIP;
DO K = 1 TO 100;
  PLI(K) = SLI(K);
END;
DO J=1 TO 4;
  DO K=1 TO 100;
    CLI(K)=SUBSTR(PLI(K),J,1);
  END;
  PUT EDIT((CLI(I) DO I=1 TO 100)) (SKIP,COL(11),100 A(1));
END;

PUT EDIT ('LEGENDE','O =') (SKIP(4),COLUMN(15),A,SKIP(2),COLUMN(16),
A);
IF SMAX > 9
THEN
DO;
  DO I = 11 TO SMAX + 1;
    PUT EDIT (I-1,'=' ,COD2(I))(SKIP(1),COLUMN(15),F(2),A,A);
  END;
END;
IMP = IMP + 11;
END;
PUT SKIP DATA (STT);                /* IMPRESSION DE LA JAUGE*/

PUT PAGE;
PUT EDIT('I',(CH(M) DO M=1 TO 10),'I')
  (SKIP(1),COL(10),10 (A(1),X(9)),A,A);
PUT EDIT('I',(C4(M) DO M=1 TO 10),'I')
  (SKIP(1),COL(10),A,10 A(10),A);
END PP;

```

```

SYNOPSIS2: PROC OPTIONS(MAIN); /* FREQUENCE DE CHAQUE DONNEE */
/* SUR FICHIER VSAM SEN */

% DECLARE UMAXV CHARACTER : /*NOMBRE DE VARIABLES */
UMAXD CHARACTER : /*NOMBRE DE DONNEES OU EQUIVAL. */
UMAXC CHARACTER : /*NOMBRE DE CARACTERES */
UMAXB CHARACTER : /*NOMBRE DE BITS/QUANTITATIVES */

% UMAXV = ' 300' : /*
% UMAXD = ' 1500' : /*
% UMAXC = '32752' : /*BLKSIZE=32752.RECFM=VBST
% UMAXB = ' 31' : /* 4 OCTETS
% DECLARE SMAXV CHARACTER : /*NOMBRE DE VARIABLES */
SMAXD CHARACTER : /*NOMBRE DE DONNEES OU EQUIVAL. */
SMAXC CHARACTER : /*NOMBRE DE CARACTERES */
SMAXB CHARACTER : /*NOMBRE DE BITS/QUANTITATIVES */

% SMAXV = ' 35' : /*
% SMAXD = ' 1500' : /*
% SMAXC = '32752' : /*
% SMAXB = ' 15' : /* 2 OCTETS

% DCL SMAX CHARACTER : /*NOMBRE DE SUBUNITS */
% SMAX = ' 20' : /*

% DCL MAXB CHARACTER : /*
% MAXB = '31' : /* MAX(UMAXB,SMAXB)

% DCL CTUMAX CHARACTER : /*MAX DE COMMENTAIRES DE U */
% CTSMAX CHARACTER : /*MAX DE COMMENTAIRES DE S */

% CTUMAX = ' 600' :
% CTSMAX = ' 20' :

% DCL CL CHARACTER : /*
% PICTCLE CHARACTER : /*
% CL = '9999' : /* FORME DE LA CLEF NUMERIQUE
% PICTCLE = '||CL||' : /*

% DCL MAXD CHARACTER : /*APRES CALCUL EVENTUEL
% MAXD = '170' : /*ETENDUE REELLE QUALITATIVES

DCL
1 IDENT STATIC.
2 NUMERO CHAR(1)
2 MISSING BIN(MAXB) FIXED.
2 CDEL CHAR(1)
2 COMMT CHAR(1);

DECLARE
1 U : /*
2 NVAR BINARY FIXED : /*UNIT
2 NEQU BINARY FIXED : /*NOMBRE DE VARIABLES
2 V(UMAXV) : /*NOMBRE DE DONNEES OU EQUIVAL.
3 CODE CHARACTER(4) : /*VARIABLES :
3 NCARC BIN(7) FIXED : /*CODE OU NOM ABREGE
3 CHAP BIT : /*NOMBRE DE CARACTERES DU NOM AB.
3 TYPE BIT : /*INDICATEUR DE CHAPITRE
3 NCAR BIN(7) FIXED : /*TYPE DE LA VARIABLE
3 INDC BINARY FIXED : /*NOMBRE DE CARACTERES DU NOM
3 VAL1 BIN(UMAXB) FIXED : /*INDEX DANS LA CHAINE C DU NOM
3 VAL2 BIN(UMAXB) FIXED : /*INDEX PREMIERE DONNEE OU MINI
2 D(UMAXD) : /*INDEX DERNIERE DONNEE OU MAXI
3 EQUV BINARY FIXED : /*DONNEES OU EQUIVALENCES :
3 NCAR BIN(7) FIXED : /*VALEUR NUMERIQUE
3 INDC BINARY FIXED : /*NOMBRE DE CARACTERES DU TEXTE
1 UC : /*INDEX DANS LA CHAINE C DU TEXTE
: /*CHAINE DE CARACTERES POUR LE
: /*STOCKAGE EN CONTINU DES TEXTES

DECLARE
1 S : /*
2 NVAR BINARY FIXED : /*SUBUNIT
2 NEQU BINARY FIXED : /*NOMBRE DE VARIABLES
2 V(SMAXV) : /*NOMBRE DE DONNEES OU EQUIVAL.
3 CODE CHARACTER(4) : /*VARIABLES :
3 NCARC BIN(7) FIXED : /*CODE OU NOM ABREGE
3 CHAP BIT : /*NOMBRE DE CARACTERES DU NOM AB.
3 TYPE BIT : /*INDICATEUR DE CHAPITRE
3 NCAR BIN(7) FIXED : /*TYPE DE LA VARIABLE
3 INDC BINARY FIXED : /*NOMBRE DE CARACTERES DU NOM
3 VAL1 BIN(SMAXB) FIXED : /*INDEX DANS LA CHAINE C DU NOM
3 VAL2 BIN(SMAXB) FIXED : /*INDEX PREMIERE DONNEE OU MINI
2 D(SMAXD) : /*INDEX DERNIERE DONNEE OU MAXI
3 EQUV BINARY FIXED : /*DONNEES OU EQUIVALENCES :
3 NCAR BIN(7) FIXED : /*VALEUR NUMERIQUE
3 INDC BINARY FIXED : /*NOMBRE DE CARACTERES DU TEXTE
1 SC : /*INDEX DANS LA CHAINE C DU TEXTE
: /*CHAINE DE CARACTERES POUR LE
: /*STOCKAGE EN CONTINU DES TEXTES

DCL
1 ENRT STATIC.
2 CLE PIC PICTCLE.
2 U :
3 DU(UMAXV) BIN(UMAXB) FIXED UNALIGNED.
3 CTU CHAR(CTUMAX).
2 S(SMAX).
3 DS(SMAXV) BIN(SMAXB) FIXED UNALIGNED.
3 CTS CHAR(CTSMAX) : /*****/

DCL REPERT FILE RECORD SEQUENTIAL INPUT;
DCL NOR CHAR(1);

DCL DONNEE FILE KEYED ENVIRONMENT(VSAM);

DCL FOCV(UMAXV+SMAXV,MAXD) BIN FIXED.
MINI(UMAXV+SMAXV) BIN(MAXB) FIXED INIT((UMAXV) 999999.(SMAXV) 32767).
MAXI(UMAXV+SMAXV) BIN(MAXB) FIXED INIT((UMAXV) -99999.
(SMAXV) -32768).
DATA BIN(MAXB) FIXED ;

DCL FOC(UMAXV+SMAXV) BIN FIXED INIT ((UMAXV+SMAXV) 0);

GET COPY EDIT (NOR) (X(5),A(1));

OPEN FILE (REPERT) TITLE ('REPERT'||NOR);
READ FILE (REPERT) INTO (IDENT);
READ FILE (REPERT) INTO (U);
READ FILE (REPERT) INTO (UC);
READ FILE (REPERT) INTO (S);
READ FILE (REPERT) INTO (SC);
CLOSE FILE (REPERT);

OPEN FILE (DONNEE) SEQUENTIAL INPUT;

DO I=1 TO UMAXV+SMAXV;
DO J=1 TO MAXD;
FOCV(I,J)=0;
END;
END;

ON ENDFILE(DONNEE) GOTO SUITE;

```

```

BOUCLE: READ FILE (DONNEE) INTO (ENRT);
DO I = 1 TO U.NVAR:                /*POUR CHAQUE DONNEE NON MANQUANTE DE */
/*ENRT, DANS U PUIS S. ETUDIEE DANS DATA*/
DATA=ENRT.U.DU(I);
IF DATA ~= MISSING
THEN
DO:
FOC(I) = FOC(I) + 1:                /*FREQUENCE DE LA VARIABLE INCREMENTEE */
IF U.V(I).TYPE
THEN                                /*QUANTITATIVE: RECHERCHE DU MIN ET MAX */
DO:                                /*REELS
IF DATA<MINI(I)
THEN MINI(I)=DATA;
ELSE:
IF DATA>MAXI(I)
THEN MAXI(I)=DATA;
ELSE:
END:
ELSE                                /*QUALITATIVE: FREQUENCE DE CHAQUE */
DO:                                /*DONNEE
DO L=U.V(I).VAL1 TO U.V(I).VAL2;
M=L-U.V(I).VAL1+1;
IF DATA=U.D.EQV(L)
THEN FOCV(I,M) = FOCV(I,M)+1;
END:
END:
END:
END:
/*****/
DO J=1 TO SMAX:
DO I=1 TO S.NVAR:
DATA=ENRT.S(J).DS(I);
IF DATA ~= MISSING
THEN
DO:
FOC(I+UMAXV) = FOC(I+UMAXV) + 1:
IF S.V(I).TYPE
THEN
DO:
IF DATA < MINI(UMAXV + I)
THEN MINI(UMAXV + I) = DATA;
ELSE:
IF DATA > MAXI(UMAXV+I)
THEN MAXI(UMAXV + I) = DATA;
ELSE:
END:
ELSE
DO:
DO L= S.V(I).VAL1 TO S.V(I).VAL2;
M=L-S.V(I).VAL1 + 1;
IF DATA=S.D.EQV(L)
THEN FOCV(I+UMAXV,M) = FOCV(I+UMAXV,M) + 1;
END:
END:
END:
END:
GOTO BOUCLE;
SUITE: CLOSE FILE (DONNEE);
/*****IMPRESSION*****/
DO I=1 TO U.NVAR:
PUT SKIP EDIT(SUBSTR(UC.U.V(I).INDC.U.V(I).NCAR)) (A);
IF U.V(I).TYPE
THEN
DO:
IF MINI(I)~=999999
THEN PUT SKIP EDIT(MINI(I). ' = MIN') (F(10),A);
ELSE:
IF MAXI(I) ~= -999999
THEN PUT SKIP EDIT(MAXI(I). ' = MAX') (F(10),A);
ELSE:
END:
ELSE
DO:
DO L=U.V(I).VAL1 TO U.V(I).VAL2;
M=L-U.V(I).VAL1 + 1;
IF FOCV(I,M) ~= 0
THEN PUT SKIP EDIT(U.D.EQV(L).SUBSTR(UC.U.D.INDC(L).U.D.NCAR(L)).
FOCV(I,M)) (F(10),X(3),A.F(6));
END:
END:
IF FOC(I) ~= 0
THEN PUT SKIP EDIT('-----TOTAL = ',FOC(I))(X(30),A.F(4));
END:
DO I=1 TO S.NVAR:
PUT SKIP EDIT(SUBSTR(SC.S.V(I).INDC.S.V(I).NCAR)) (A);
IF S.V(I).TYPE
THEN
DO:
IF MINI(I+UMAXV)~=32767
THEN PUT SKIP EDIT(MINI(I+UMAXV). ' = MIN') (F(10),A);
ELSE:
IF MAXI(I+UMAXV) ~= -32768
THEN PUT SKIP EDIT(MAXI(I+UMAXV ). ' = MAX') (F(10),A);
ELSE:
END:
ELSE
DO:
DO L=S.V(I).VAL1 TO S.V(I).VAL2;
M=L-S.V(I).VAL1 + 1;
IF FOCV(I+UMAXV,M) ~= 0
THEN PUT SKIP EDIT(S.D.EQV(L).SUBSTR(SC.S.D.INDC(L).S.D.NCAR(L)).
FOCV(I+UMAXV,M)) (F(10),X(3),A.F(6));
END:
END:
IF FOC(I+UMAXV)~=0
THEN PUT SKIP EDIT('----- = ',FOC(I+UMAXV))(X(30),A.F(4));
END:
END:

```