

MORTRAN USER MANUAL

MORTRAN a été initialement développé par James COOK au SLAC. La version que nous utilisons a été développée par Rolf NIERHAUS au CERN, puis modifiée par nous-mêmes.

MORTRAN est un interpréteur de langage (actuellement à une seule passe) qui procède essentiellement par remplacement.

- des “macros”, écrites par le “programmeur”, permettent de définir le langage souhaité.
- un texte “en entrée”, donné par “l'utilisateur”, est ensuite lu et on y applique les macros précédentes : lorsqu'une instruction de ce langage est reconnue (“*a pattern is matched*”), elle est remplacée par une suite d'instructions et éventuellement de macros qui sont à leur tour réexplorées et peuvent être remplacées par une suite d'instructions et de macros qui sont à leur tour ...
- On obtient en sortie un texte (en FORTRAN dans notre cas, mais éventuellement dans un autre langage) qui doit ensuite être soumis à un compilateur.

MORTRAN est donc un “moteur” qui a ses propres règles, indépendamment des langages définis par le programmeur. Nous présentons ici ces principales règles. Les langages développés (MTN, ZDL, ZBM, PDL ...) sont présentés par ailleurs.

1) – TEXTE NON RECONNU PAR LES MACROS

Toute partie de texte non reconnue par les macros est transmise au fichier de sortie ; c'est la façon habituelle pour MORTRAN d'arrêter d'analyser une (ou une partie) instruction. En conséquence :

- Une instruction FORTRAN classique ne sera pas interprétée par les langages (si ceux-ci sont correctement définis) ; en plus des langages définis par le programmeur, on peut donc utiliser toutes les possibilités de FORTRAN.
- Une erreur de syntaxe du langage utilisé n'est pas toujours directement diagnosticable ; elle se traduit en général par la non interprétation du texte, qui est ainsi transmis directement au fichier de sortie ; une erreur sera générée lors de la compilation de ce texte. Dans d'autres cas, une erreur de syntaxe amorcera une “bombe” qui se déclenchera plus ou moins rapidement.

2) – OBJETS ET STRUCTURES DE BASE DE MORTRAN

LES DIFFÉRENTS TYPES DE CARACTÈRES	<ul style="list-style-type: none"> - caractère <u>alphanumérique</u> : c'est l'ensemble des caractères alphanumériques, ainsi que le caractère souligné (_) - caractère <u>non alphanumérique</u> : tous les autres. - caractères <u>spéciaux</u> : ' , ; , " , & , # , < , > .
LE MOT	C'est une suite composée uniquement de caractères alphanumériques et délimitée par des caractères non alphanumériques
L'INSTRUCTION	Elle est délimitée par des ;
LE BLOC	Il est délimité par le caractère de début de bloc (<) et de fin de bloc (>). Les blocs MORTRAN s'emboîtent, mais ne peuvent pas se chevaucher.

3) TRAITEMENT PARTICULIER DES CARACTÈRES ESPACES

Lors de la lecture du texte, MORTRAN effectue les traitements suivants :

- Tous les caractères espaces situés entre un caractère non alphanumérique et un caractère alphanumérique ou situés entre deux caractères non alphanumériques sont supprimés.
- Tous les caractères espaces situés entre deux caractères alphanumériques sont remplacés par un seul caractère espace.

Lors de l'écriture par l'utilisateur d'un texte correspondant à un langage donné :

- Peu importe si l'on met des espaces entre deux caractères non alphanumériques ou entre un caractère alphanumérique et un caractère non alphanumérique.
- Si le langage exige un espace entre deux caractères alphanumériques, il faut obligatoirement en mettre au moins un (contrairement au FORTRAN classique).

4) COMMENTAIRES

On distingue deux types de commentaires :

- une ligne commentaire : le premier caractère de la ligne est un !
- un commentaire quelconque, sur une partie de ligne : il est délimité par des doubles quotes ("). Le commentaire doit obligatoirement être fermé à la fin de la ligne, sinon diagnostic d'erreur :

```
*** MTN ERROR ***  
UNCLOSED COMMENT STRING
```

5) DIVERS

- Longueur des lignes actuellement limitée à **80 caractères**.
- Caractères autorisés : ASCII non étendu (pas de é ou autres).
- Les chaînes de caractères (c'est-à-dire tout texte compris entre quotes) ne sont pas traitées et modifiées par MORTRAN. Toute chaîne de caractères doit se terminer avant la fin de la ligne, sinon message d'erreur :

```
*** MTN ERROR ***  
UNCLOSED QUOTED STRING
```

6) DIRECTIVES MORTRAN

SYNTAXE	<ul style="list-style-type: none"> - Les directives &u, &v, &F, &M doivent être en début de ligne, et toutes seules. Elles ne doivent pas être terminées par un ; - Les autres directives doivent être terminées par un ; et peuvent être regroupées sur une même ligne. - La syntaxe générale est &X; pour activer la directive X et &X0. (0 : chiffre zéro) pour la désactiver
ANNOTATION	<ul style="list-style-type: none"> - &A; [&A0]; : active [ou désactive] le mode annotation - Valeur par défaut : &A0; <p><i>L'annotation permet d'avoir, dans le fichier de sortie FORTRAN, en commentaire l'instruction MORTRAN lue, puis les instructions FORTRAN générées.</i></p>
CONVERSION MINUSCULE-MAJUSCULE	<ul style="list-style-type: none"> - &C; [&C0]; : conversion en majuscule [pas de conversion] du texte lu. - Valeur par défaut : &C; <p><i>Quelle que soit l'option de conversion, l'interprétation des instructions du langage est "case insensitive"</i></p>
TEXTE FORTRAN OU MORTRAN	<ul style="list-style-type: none"> - &F : le texte qui suit est du FORTRAN, avec ses caractéristiques habituelles (1 instruction par ligne, premier caractère en colonne 7, ...). Ce texte n'est pas du tout interprété par MORTRAN jusqu'à la rencontre de la directive : - &M : le texte qui suit doit être interprété par MORTRAN
CONVERSION DES CHAÎNES DE CARACTÈRES EN CODE HOLLERITH	<ul style="list-style-type: none"> - &H; [&H0]; : les chaînes de caractères sont [ne sont pas] converties en code Hollerith - Valeur par défaut : &H
LISTING (LISTAGE en français dans le texte)	<ul style="list-style-type: none"> - &L; [&L0]; : active [ou désactive] la génération d'un listing. - Valeur par défaut : &L0; <p><i>En plus de l'impression du texte lu par MORTRAN, le listing comporte quelques indications dans les premières colonnes</i></p> <ul style="list-style-type: none"> - Lors de l'ouverture (<) ou de la fermeture (>) d'un bloc MORTRAN, un chiffre indique le niveau d'emboîtement de ce bloc. A la fin de chaque routine (instruction end;) tous les blocs doivent être fermés (le niveau d'emboîtement doit être 0)
FICHIER DE SORTIE	<ul style="list-style-type: none"> - &V nom-de-fichier
LECTURE D'UN FICHIER	<ul style="list-style-type: none"> - &U nom-de-fichier
AUTRES DIRECTIVES (DEBUGGING (C.À.D. DÉBOGAGE))	<ul style="list-style-type: none"> - &T1; [ou &T2; &T3; &T9]; : macro trace. Plus le chiffre est élevé, plus c'est bavard. - &T0; : fin du mode trace - &D; : dump (vidage) des macros - && : fin de fichier

7) LES ERREURS DONT VOUS NE VOUS LASSEREZ PAS

Un des inconvénients essentiels de MORTRAN est la difficulté de générer des diagnostics pour des erreurs de syntaxe, de frappe... Les erreurs les plus courantes sont les suivantes :

- Il manque un ; . Cette erreur se traduit en général par deux instructions à la suite dans le fichier de sortie FORTRAN. Parfois les langages exigent un ; en début ou en fin d'instruction. L'absence de ; entraînera la non reconnaissance de ces instructions.
- Une ligne trop longue a été tronquée.
- Le nombre d'ouvertures de bloc MORTRAN (<) ne correspond pas au nombre de fermetures (>). Si le nombre de fermetures de blocs est supérieur au nombre d'ouvertures, on a le diagnostic suivant :

```
*** MTN ERROR ***
Unable to close an unopen MTN block
```

Si, à la fin d'une subroutine, tous les blocs ne sont pas fermés, on a le diagnostic suivant :

```
*** MTN ERROR ***
UNCLOSED MTN BLOCK
STACK POINTERS (should be all zeroes)
```

La sortie "listing" de MORTRAN (ne pas oublier de mettre l'option **&L;**) facilite en général la détection des erreurs ci-dessus grâce à des indications portées dans les premières colonnes (cf. commentaire de l'option &L).