

**Manuel d'Utilisation**  
**Fascicule U4.9- : Impression/Test**  
**Document : U4.91.03**

## Procédure IMPR\_TABLE

---

### 1 But

---

Imprimer sur un fichier de type "listing" le contenu d'une table.

La commande permet d'imprimer un sous-ensemble de la table sous différents formats. Le format par défaut (EXCEL) est pratique pour un dépouillement par ce logiciel, mais d'autres formats permettent une visualisation commode avec AGRAF (par exemple).

Si la table est "grosse" (beaucoup de paramètres), l'impression peut se faire sur de larges pages (jusqu'à 2000 caractères par lignes). Au delà, les lignes sont tronquées.

## 2 Syntaxe

```

IMPR_TABLE (
  ♦ TABLE = matable, [tabl_*]
  ◇ TITRE_TABLE = titre [l_Kn]
  ◇ SENSIBILITE = ( ... voir [U4.50.02])
  ◇ FICHER = / 'RESULTAT', [DEFAULT]
              / nomfic, [Kn]

  # 1. choix des lignes à imprimer :
  ◇ FILTRE =(_F( ♦ NOM_PARA = para, [Kn]
                  / ◇ CRIT_COMP = / 'EQ', [défaut]
                              / 'NE',
                              / 'LT',
                              / 'GT',
                              / 'LE',
                              / 'GE',

                  ♦ / VALE_I = ival, [I]
                    / VALE_K = kval, [Kn]

                    / / VALE = rval, [R]
                      / VALE_C = cval, [C]
                    ◇ | PRECISION = / prec, [R8]
                              / 1.0D-3, [DEFAULT]
                      | CRITERE = / 'RELATIF', [DEFAULT]
                              / 'ABSOLU',

                    / CRIT_COMP = / 'MAXI',
                              / 'ABS_MAXI',
                              / 'MINI',
                              / 'ABS_MINI',
                              / 'VIDE',
                              / 'NON_VIDE',

                  ),),

  # 2. choix des colonnes à imprimer :
  ◇ / TOUT_PARA = 'OUI', [DEFAULT]
    / NOM_PARA = lpara, [l_Kn]

  # 3. choix de l'ordre des lignes à imprimer :
  ◇ TRI =(_F( ♦ NOM_PARA = lpara, [l_Kn]
              ◇ ORDRE = | 'CROISSANT', [l_Kn]
                      | 'DECROISSANT',

              ◇ | PRECISION = / prec, [R8]
                      / 1.0D-3, [DEFAULT]
              | CRITERE = / 'RELATIF', [DEFAULT]
                      / 'ABSOLU',

              ),),

  # 4. choix des formats d'impression :
  ◇ FORMAT = / 'EXCEL', [DEFAULT]
            / 'ASTER',
            / 'MOT_CLE',
            / 'AGRAF',
            / 'TABLEAU',

```

Titre : Procédure IMPR\_TABLE  
Auteur(s) : J. PELLET

Date : 08/12/03  
Clé : U4.91.03-E1 Page : 3/10

```

    ◇ PAGINATION      =      lpagi,                      [l_Kn]
    ◇ FORMAT_R        = / '1PE12.5',                    [DEFAULT]
                        / form,
    ◇ FORMAT_C        = / 'MODULE_PHASE',                [DEFAULT]
                        / 'REEL_IMAG',
    ◇ IMPR_FONCTION    = / 'NON',                        [DEFAULT]
                        / 'OUI',
    ◇ INFO      =      / 1,                               [DEFAULT]
                        / 2,

```

)

## 3 Opérandes

### 3.1 Qu'est-ce qu'une table ?

Une `table` est une structure de données de caractère informatique permettant de stocker un ensemble de valeurs entières, réelles, complexes ou chaînes de caractères.

Une `table` est comparable à une base de données EXCEL (version 5), c'est-à-dire qu'on peut la voir comme une liste de **colonnes** en vis-à-vis. Chaque colonne a un nom de champ, que nous appelons **paramètre**, et contient des données similaires de type : `I`, `R`, `C`, `K8`, `K16`, `K24` ou `K32`.

**Exemple :**

NUME_ORDRE	INST	NOEUD	G_LOCAL
1	10.	N1	5.
1	10.	N2	6.
1	10.	N3	7.
1	10.	N4	8.
2	20.	N1	9.
2	20.	N2	9.
2	20.	N3	8.
2	20.	N4	8.
3	30.	N1	7.
3	30.	N2	6.
3	30.	N3	5.

On peut aussi voir la `table` comme une suite d'enregistrements (que nous appellerons **lignes**). Toutes les lignes n'ont pas forcément la même structure (c'est-à-dire les mêmes paramètres). Par exemple :

ACTION	NUME_ORDRE	INST	NOEUD	DX	DY	MAILLE	SIXX
INTITULE 1	1	10.	N1	3.	5.		
INTITULE 1	1	10.	N2	6.	7.		
INTITULE 1	1	10.	N3	8.	9.		
INTITULE 1	2	20.	N1	11.	12.		
INTITULE 1	2	20.	N2	15.	13.		
INTITULE 1	2	20.	N3	19.	18.		
INTITULE 2	2	20.				MA1	-12.
INTITULE 2	2	20.				MA2	-14.

Pour imprimer le contenu d'une `table`, l'utilisateur pourra :

- sélectionner les colonnes et les lignes qu'il veut imprimer [§3.2]
- choisir un critère d'ordre d'impression des lignes [§3.3]
- choisir le "format" d'impression : en lignes, en tableaux, ... [§3.4]

### 3.2 Opérande TABLE

- ♦ `TABLE` = `matable`  
Nom de la `table` que l'on veut imprimer

### 3.3 Opérande TITRE\_TABLE

- ♦ `TITRE_TABLE` = `titre`  
Chaîne de caractère qui sera imprimée avant la `table` (et avant le titre éventuellement attaché à la `table`). Cette chaîne peut permettre de mieux distinguer des impressions de `tables` mises bout à bout dans un même fichier.

## 3.4 Opérande SENSIBILITE

- ◇ SENSIBILITE = par\_sensi  
Nom du paramètre sensible associé (voir [U4.50.01])

## 3.5 Sélectionner ce que l'on veut imprimer : les mots clés FILTRE et NOM\_PARA, TOUT\_PARA

L'utilisateur choisit les colonnes à imprimer grâce aux mots clés NOM\_PARA et TOUT\_PARA. Il choisit les lignes à imprimer grâce au mot clé facteur (répétable à volonté) FILTRE.

### 3.5.1 Mot clé FILTRE

Le mot clé facteur FILTRE permet de ne retenir dans la table que les lignes vérifiant certains critères imposés par l'utilisateur. Les **occurrences** du mot clé s'ajoutent les unes aux autres comme des **filtres successifs**. Pour chaque occurrence de ce mot clé, on précise le nom du paramètre pour lequel on impose une condition, le type de condition (égalité, non-égalité, plus petit, ...) ainsi que la valeur associée à la condition.

#### 3.5.1.1 Opérande NOM\_PARA

- ◇ NOM\_PARA = para  
para est le nom du paramètre sur lequel porte la contrainte de filtrage.

#### 3.5.1.2 Opérande CRIT\_COMP

- ◇ CRIT\_COMP = crit  
crit est le type de la contrainte de filtrage.

EQ	"égalité" pour les entiers, les textes, les réels ou les complexes. Pour les nombres flottants (réels ou complexes), cette égalité est évaluée avec une certaine tolérance donnée par les mots clés PRECISION et CRITERE.
NE	"non-égalité" (Cf. EQ)
LT	"plus petit que" Relations d'ordre : - naturelle pour les entiers et les réels - alphabétique pour les textes - <b>invalide</b> pour les <b>complexes</b>
GT	"plus grand que" (Cf. LT)
LE	"plus petit ou égal à" (Cf. LT)
GE	"plus grand ou égal à" (Cf. LT)
VIDE	cellule vide
NON_VIDE	cellule non vide
MAXI	la ligne retenue sera celle qui aura la valeur maximale
ABS_MAXI	la ligne retenue sera celle qui aura la valeur absolue maximale
MINI	la ligne retenue sera celle qui aura la valeur minimale
ABS_MINI	la ligne retenue sera celle qui aura la valeur absolue minimale

#### 3.5.1.3 Opérandes VALE / VALE\_I / VALE\_C / VALE\_K

Ces différents mots clés sont utilisés en fonction du type de la colonne associée au paramètre sur lequel porte la contrainte : entier, réel, complexe ou texte.

On donne en argument la valeur associée à la contrainte ; par exemple 12 si on ne s'intéresse qu'au numéro d'ordre 12.

Cet opérande est inutile quand on utilise les types de contrainte : 'VIDE', 'NON\_VIDE', 'MAXI', 'MINI', ...

### 3.5.1.4 Opérandes CRITERE / PRECISION

Lorsque le paramètre contraint est de type flottant (réel ou complexe) et que le type de contrainte est l'égalité (ou la non-égalité), cette égalité est évaluée avec une certaine tolérance.

```
PRECISION = eps,          eps est la tolérance
CRITERE=   / 'RELATIF' ,   |x-xref| < eps*|xref|
           / 'ABSOLU'  ,   |x-xref| < eps
```

### 3.5.1.5 Exemple d'utilisation du mot clé FILTRE

Si l'on précise :

```
FILTRE= ( _F(NOM_PARA= 'NOEUD' , VALE_K = 'N7' , ) ,
          _F(NOM_PARA= 'INST' , CRIT_COMP= 'GT' , VALE= 3.0 , ) ,
          _F(NOM_PARA= 'INST' , CRIT_COMP= 'LT' , VALE= 13.0 , ) ,
        ) ,
```

On sélectionne ainsi les lignes de la table telles que le paramètre NOEUD vaut 'N7' et telles que le paramètre INST soit compris entre 3. et 13.

### 3.5.2 Opérandes TOUT\_PARA / NOM\_PARA

Le mot clé TOUT\_PARA = 'OUI' indique que l'on imprimera **toutes** les colonnes de la table.

Si l'utilisateur utilise le mot clé NOM\_PARA = lpara, on n'imprimera que les paramètres spécifiés dans la liste lpara et dans l'ordre de la liste.

## 3.6 Ordonner les lignes d'une table : le mot clé facteur TRI

Une seule occurrence est acceptée pour le mot clé facteur TRI.

### 3.6.1 Opérande NOM\_PARA = lpara

Ce mot clé sert à préciser la liste des paramètres qui serviront à ordonner les lignes de la table (il peut y avoir besoin de plusieurs paramètres).

#### Remarques :

*On peut utiliser pour le tri des paramètres que l'on n'imprime pas.  
Aujourd'hui, on n'accepte que 2 paramètres pour le tri (au plus).*

### 3.6.2 Opérande ORDRE

Ce mot clé sert à préciser pour chacun des paramètres de lpara, si on doit utiliser un ordre croissant ou décroissant.

Les relations d'ordre utilisées sont :

- l'ordre naturel pour les entiers et les réels,
- l'ordre alphabétique pour les textes et les noms de concepts.

#### Remarques :

*On ne peut pas se servir d'un paramètre complexe pour classer les lignes d'une table.  
Si on n'utilise que l'ordre croissant, le mot clé ORDRE est inutile (c'est la valeur par défaut).  
Pour les paramètres de type NOEUD (ou MAILLE), l'ordre est alphabétique car ces paramètres contiennent le nom des nœuds (ou des mailles).*

### 3.6.3 Opérands CRITERE / PRECISION

Lorsque le tri est fait sur deux paramètres, on peut fixer une précision pour le test d'égalité du premier paramètre, puis ensuite trier en fonction du deuxième paramètre. Cela permet par exemple de trier de façon logique les points d'une grille qui ne sont pas exactement placés sur des lignes parallèles à OY (erreur d'arrondi sur le 1<sup>er</sup> paramètre).

```
PRECISION = eps,          eps est la tolérance
CRITERE=   / 'RELATIF' ,   |x-xref| < eps*|xref|
           / 'ABSOLU'  ,   |x-xref| < eps
```

### 3.6.4 Exemple d'utilisation du mot clé TRI

Si l'on précise :

```
TRI=_F (NOM_PARA= ('NOEUD', 'INST'), ORDRE= ('CROISSANT', 'DECROISSANT')),
```

On imprimera les lignes de la table dans l'ordre alphabétique des nœuds. S'il existe plusieurs lignes correspondant à un nœud donné, le **second** critère de tri (INST) sera utilisé pour classer ces lignes. Dans cet exemple, on classera les instants par valeurs décroissantes.

## 3.7 Choix du format d'impression : mots clés FORMAT, PAGINATION, ...

Par défaut le format d'impression est le format 'EXCEL', c'est à dire présentation en colonnes des différents paramètres sélectionnés (comme pour les exemples de ce document). Le logiciel EXCEL offre un puissant ensemble d'outils permettant d'exploiter cette base de données : tri, filtrage, tableaux dynamiques,...

### 3.7.1 FORMAT = 'EXCEL' ou 'AGRAF'

Un exemple de table imprimée au format 'EXCEL' :

NUME_ORDRE	INST	NOEUD	DX	DY
1	4.	N7	3.4	3.8
1	4.	N4	2.4	2.8
1	4.	N2	1.4	1.8
4	8.	N7	3.4	3.8
4	8.	N4	2.4	2.8
4	8.	N2	1.4	1.8
7	20.	N7	3.4	3.8
7	20.	N4	2.4	2.8
7	20.	N2	1.4	1.8

La seule différence entre le format 'EXCEL' et le format 'AGRAF' est que pour ce dernier, les colonnes de "texte" (par exemple NOEUD ci-dessus) sont précédés d'un "\".

### 3.7.2 FORMAT = 'ASTER'

La différence entre le format 'ASTER' et le format 'EXCEL' est l'entête de la table et sa dernière ligne. Cette différence est justifiée par le fait qu'une table imprimée au format 'ASTER' doit pouvoir être relue par Code\_Aster (commande LIRE\_TABLE [U7.02.03]).

Exemple :

```
#DEBUT_TABLE
#TITRE ASTER 6.03.11 CONCEPT TAB_REAC CALCULE LE 12/07/2002
#TITRE TABL_POST_RELE
INTITULE   RESU   —   NOM_CHAM      NUME_ORD      INST      DY
K8         K8         K16         I         R         R
MESSAGE    RESU      REAC_NODA         1  2.50000E-01 -2.25146E+00
MESSAGE    RESU      REAC_NODA         2  5.00000E-01 -4.44089E+00
MESSAGE    RESU      REAC_NODA         3  7.50000E-01 -6.59515E+00
MESSAGE    RESU      REAC_NODA         4  1.00000E+00 -8.65972E+00
MESSAGE    RESU      REAC_NODA         5  1.25000E+00 -1.06742E+01
MESSAGE    RESU      REAC_NODA         6  1.50000E+00 -1.26438E+01
MESSAGE    RESU      REAC_NODA         7  1.75000E+00 -1.45569E+01
#FIN_TABLE
```

### 3.7.3 FORMAT = 'MOT\_CLE'

```
NUME_ORDRE: 1 INST: 4.  NOEUD: N7 DX: 3.4  DY: 3.8
NUME_ORDRE: 1 INST: 4.  NOEUD: N4 DX: 2.4  DY: 2.8
NUME_ORDRE: 1 INST: 4.  NOEUD: N2 DX: 1.4  DY: 1.8
NUME_ORDRE: 4 INST: 8.  NOEUD: N7 DX: 3.4  DY: 3.8
...
```

Ce format ressemble beaucoup au format EXCEL, on fait précéder chaque valeur de la table par le nom du paramètre associé. Il peut être pratique pour des recherches via l'utilitaire UNIX `grep`.

### 3.7.4 FORMAT = 'TABLEAU'

Le format 'TABLEAU' est réservé aux tables à double entrée. L'impression se fait sous la forme d'un tableau dont on a documenté les noms de colonnes et les noms de lignes. Cette impression concerne les tables ayant 3 paramètres. On représente les valeurs de l'un des 3 paramètres en fonction des 2 autres.

Exemple : DX en fonction de NOEUD et INST :

DX		INST		
		4.	8.	20.
NOEUD	N7	3.4	3.4	3.4
	N4	2.4	2.4	2.4
	N2	1.4	1.4	1.4

```
NOM_PARA = ( 'NOEUD', 'INST', 'DX' )
```

Le paramètre "remplissant" le tableau est le 3<sup>ème</sup> de la liste indiquée par le mot clé simple NOM\_PARA.

Le paramètre "abscisse" est le 2<sup>ème</sup> de la liste indiquée par le mot clé simple NOM\_PARA.

Le paramètre "ordonnée" est le 1<sup>er</sup> de la liste indiquée par le mot clé simple NOM\_PARA.

#### Remarque :

Si la table contient plus de 3 paramètres, on peut utiliser le mot clé PAGINATION pour "égrenner" les valeurs prises par les autres paramètres (voir exemple 3).



### 3.7.5 Opérande PAGINATION

Ce mot clé sert à imprimer une table par morceaux (comme un ensemble de plus petites tables). On donne une liste de paramètres (`lpagi`) qui seront égrenés et imprimés comme titres des petites tables.

La liste des paramètres de pagination (`lpagi`) doit être incluse dans la liste complète des paramètres que l'on veut imprimer (`lpara`).

Si l'on veut utiliser la pagination et le format 'TABLEAU', il faut que la liste `lpara`, une fois qu'on lui a retiré les paramètres de `lpagi` soit formée de 3 paramètres résiduels. Ces 3 paramètres seront utilisés pour la présentation en tableau [§3.4.3].

Exemple : `FORMAT='EXCEL', PAGINATION = 'NOEUD'`

NOEUD: N7

NUME_ORDRE	INST	DX	DY
1	4.	3.4	3.8
4	8.	3.4	3.8
7	20.	3.4	3.8

NOEUD: N4

NUME_ORDRE	INST	DX	DY
1	4.	2.4	2.8
4	8.	2.4	2.8
7	20.	2.4	2.8

NOEUD: N2

NUME_ORDRE	INST	DX	DY
1	4.	1.4	1.8
4	8	1.4	1.8
7	20.	1.4	1.8

### 3.7.6 Opérande FORMAT\_C

Pour les complexes, deux formats d'impression sont disponibles (module/phase par défaut ou partie réelle/partie imaginaire).

### 3.7.7 Opérande FORMAT\_R

Ce mot clé permet de choisir le nombre de décimales imprimées pour chaque nombre flottant : (réel ou complexe).

On utilise pour cela la syntaxe du FORTRAN.

Par exemple, pour la valeur par défaut : '1PE12.5', on imprimera chaque nombre flottant sur 12 caractères, avec 5 décimales et en notation scientifique (avec un exposant).

### 3.7.8 Opérande IMPR\_FONCTION

Pour les tables contenant dans leurs cellules des noms de `fonction`, ce mot clé permet d'indiquer que l'on veut également imprimer le **contenu** des fonctions référencées dans la table.

On imprime d'abord la table (comme il a été vu ci-dessus) puis on "boucle" sur toutes les fonctions contenues dans la partie imprimée de la table et on imprime ces fonctions (comme le fait la commande `IMPR_COURBE`).

## 3.8 Opérande INFO

◇ `INFO = inf`

Imprime dans le fichier "message" des informations complémentaires si `inf=2`. Rien ne se passe si `inf=1`.

## 4 Exemples

Exemple 1 : pour découvrir la structure d'une table

```
IMPR_TABLE(TABLE = POST_REL)
```

Exemple 2 : imprimer quelques paramètres avec filtre et tri

```
IMPR_TABLE(TABLE = POST_REL,  
  FILTRE=_F(NOM_PARA='INST', VALE= 0., CRITERE = 'ABSOLU'),  
  TRI=_F(NOM_PARA=('ABS_CURV', 'COOR_X'), ORDRE=('DECROISSANT', 'CROISSANT')),  
  NOM_PARA = ('COOR_X', 'TEMP'),  
)
```

Exemple 3 : utiliser la pagination et le format TABLEAU

```
IMPR_TABLE(TABLE = POST_REL,  
  NOM_PARA = ('COOR_X', 'TEMP', 'ABS_CURV', 'INST', 'COOR_Y'),  
  PAGINATION= ('INST', 'COOR_X'),  
  FORMAT = 'TABLEAU',  
)
```

Exemple 4 : imprimer le contenu de la dérivée de la table G1 par rapport au paramètre sensible YOUNG :

```
IMPR_TABLE(TABLE = G1, SENSIBILITE = YOUNG)
```