

**SOUS-PROGRAMME DE LECTURE DE PARAMÈTRES SAISIS
SANS CONTRAINTE DE CADRAGE SUR UN ENREGISTREMENT**

Hélène BIGOT
Centre de Calcul du CNRS
23 rue du Loess
67200 STRASBOURG
EARN : <C01301àFRCCSC21.EARN>

Résumé :

Le sous-programme LECPAR fait partie de la boîte à outils MODULAD proposé à tout programmeur FORTRAN. Il constitue une solution au problème de la programmation de la lecture de paramètres saisis sans contrainte de cadrage sur un enregistrement. Un programme pour lequel ce problème est réglé, est d'autant plus convivial.

Mots-clés :

outil de programmation FORTRAN 77, lecture de paramètres

1. Intérêt d'un tel sous-programme

Lorsqu'une nouvelle méthode (ou un nouvel algorithme) est programmée, l'auteur concentre en général d'abord son attention sur le noyau du programme, à savoir l'endroit où l'originalité calculatoire de la méthode est mise en oeuvre. Ceci est bien entendu primordial pour assurer la fiabilité du résultat. Cette phase est souvent délicate et longue, de sorte que lorsqu'elle est terminée, les problèmes d'entrée des paramètres de la méthode, et de sortie des résultats apparaissent comme étant mineurs, voire annexes.

De plus, lorsque le langage de programmation se trouve être le FORTRAN 77, celui-ci est tel (distinction entre nombre entier et nombre réel, peu de fonctions de manipulation de caractères, ...), que beaucoup de bonnes volontés se trouvent vite découragées par la complexité algorithmique d'un sous-programme permettant une lecture de paramètres saisis sans contrainte de cadrage.

Cet ensemble de circonstances fait que la plupart des programmes FORTRAN existant impose que les paramètres soient saisis avec des contraintes de cadrage strictes. Mais ceci rend ces programmes impopulaires auprès de leurs utilisateurs, d'autant que ces derniers n'ont en général aucune notion de la programmation en FORTRAN, mais au contraire une certaine pratique de logiciels intégrés considérés comme conviviaux.

Pour que l'intérêt d'une méthode programmée en FORTRAN ne soit pas diminué par la rigidité de son mode d'utilisation, il est donc nécessaire d'en soigner l'apparence pour les utilisateurs potentiels. C'est dans cette optique qu'a été développé le sous-programme LECPAR de lecture de paramètres saisis sans contrainte de cadrage sur un enregistrement.

Ce sous-programme a été conçu de façon suffisamment générale pour qu'il puisse être utilisé dans la majorité des cas. Et afin que tout un chacun n'ait pas à réinventer la poudre, il a semblé naturel qu'il fasse partie de la boîte à outils MODULAD proposée à tout programmeur FORTRAN.

2. Cadre d'utilisation

Le sous-programme LECPAR n'accepte pas n'importe quoi en entrée, il suppose certaines conditions, notamment sur la forme des paramètres :

- L'enregistrement à lire est de longueur inférieure ou égale à 80 caractères.
- Sur un enregistrement, il y a au maximum 40 paramètres à lire.
- Les paramètres sont séparés par au moins un caractère blanc.
- Un paramètre auquel on ne veut pas donner une valeur, doit être codé par un point isolé (..).

- Les paramètres peuvent être :
 - des entiers en format In, pour lesquels tout caractère zéro (0) est forcément codé,
 - des réels en format Fn.m, pour lesquels tout caractère zéro (0) est forcément codé,
 - des réels en format En.m, pour lesquels le caractère E est forcément codé et sans caractère blanc,
 - des chaînes de caractères sans caractère blanc.

3. Principes du décodage

Pour chaque enregistrement traité par le sous-programme, le nombre de paramètres décodés est indiqué. Et pour chacun de ces paramètres, trois valeurs sont restituées : la valeur entière, la valeur réelle (simple précision) et la valeur caractère. Elles sont valorisées selon les règles suivantes :

- Si un paramètre n'est constitué que de chiffres et d'éventuellement un signe + ou - au début, alors le paramètre est considéré comme entier. Sa valeur entière correspond à l'entier effectivement codé. Sa valeur réelle est la conversion en réel de sa valeur entière. Sa valeur caractère est la chaîne correspondant au codage sur l'enregistrement.
- Si un paramètre n'est constitué que de chiffres, d'au plus un caractère E, d'au plus un caractère point (.) situé avant le caractère E et d'au plus deux signes + ou - (placés en début ou immédiatement après le caractère E), alors le paramètre est considéré comme réel. Sa valeur entière prend une valeur conventionnelle (i.e. le plus grand entier positif $INTMAX = 2^{**31} - 1$). Sa valeur réelle est le nombre réel saisi. Sa valeur caractère est la chaîne correspondant au codage sur l'enregistrement.
- Si un paramètre est codé par un point isolé, sa valeur entière prend une valeur conventionnelle (i.e. le plus grand entier négatif $-INTMAX = -2^{**31} + 1$), sa valeur réelle prend une valeur conventionnelle (i.e. le plus grand réel négatif $-REAMAX = -1E38$), et sa valeur caractère est un point isolé.
- Si un paramètre n'est ni entier, ni réel aux sens indiqués ci-dessus, ni un point isolé, alors le paramètre est considéré comme caractère. Sa valeur entière prend une valeur conventionnelle (i.e. le plus grand entier positif $INTMAX = 2^{**31} - 1$). Sa valeur réelle prend une valeur conventionnelle (i.e. le plus grand réel positif $REAMAX = 1E38$). Sa valeur caractère est la chaîne correspondant au codage sur l'enregistrement.
- Si sur un enregistrement, il y a plus de paramètres saisis que de paramètres à prendre en compte, les paramètres excédentaires en fin d'enregistrement sont ignorés.

- Si un paramètre en fin d'enregistrement n'est pas codé du tout (fin de l'enregistrement laissée vierge), sa valeur entière prend la valeur entière zéro, sa valeur réelle prend la valeur réelle zéro, et sa valeur caractère est la chaîne de caractère 'NON CODE'.
- S'il n'y a pas (ou plus) d'enregistrement à lire, le nombre de paramètres décodés prend la valeur entière négative -1, et le sous-programme se termine sans erreur de programmation. Cela permet, dans la suite du programme, de traiter ce cas d'erreur d'utilisation.

4. Principes d'utilisation

Lors de l'appel du sous-programme LECPAR, il faut indiquer :

- l'unité logique du fichier à lire,
- la longueur de l'enregistrement à lire,
- le nombre de paramètres à lire.

En retour, le sous-programme LECPAR restitue :

- le nombre de paramètres effectivement saisis,
- un vecteur monodimensionné de type entier contenant les décodages de type entier des paramètres,
- un vecteur monodimensionné de type réel contenant les décodages de type réel des paramètres,
- un vecteur monodimensionné de type caractère contenant les décodages de type caractère des paramètres.

Après un appel à LECPAR, le programmeur doit choisir pour chaque paramètre la valeur correspondant au bon type. De plus, la comparaison des trois valeurs restituées par LECPAR pour un paramètre donné permet de faire les contrôles de validité de la saisie.

Le sous-programme LECPAR permet aussi de lire une chaîne de caractères avec des blancs (par exemple le titre d'une analyse). Mais cela suppose après l'appel un traitement spécial par le programmeur. La chaîne restituée ne correspondra pas exactement à la chaîne lue : en effet, toute succession de caractères blancs, quelle qu'en soit la longueur, sera réduite à un seul caractère blanc. Cette restriction n'est pas toujours admissible, mais dans la majorité des cas (notamment pour traiter un titre), ce procédé est suffisant.

5. Conclusion

Le sous-programme LECPAR, malgré ses limitations, permet d'aborder plus facilement la lecture de paramètres saisis sans contrainte de cadrage sur un enregistrement, lorsque l'on programme en FORTRAN 77. Certaines améliorations de ce sous-programme pourraient être envisagées, notamment la prise en compte comme séparateur de paramètres d'un autre caractère que le caractère blanc (par exemple la virgule). Toute bonne volonté voulant contribuer au perfectionnement de LECPAR sera bien sûr la bienvenue.

REMERCIEMENTS

Par ses multiples suggestions, Mr Gérard THAURONT de l'INRETS a beaucoup contribué à la mise au point du sous-programme LECPAR. Qu'il en soit remercié ici

NOTA : Pour obtenir la boîte à outils MODULAD, dont fait partie le sous-programme LECPAR, s'adresser à Martine CORNELIS à l'INRIA à Rocquencourt 78153 Le Chesnay Cedex

ANNEXE 1 : exemple d'appel à LECPAR

```
=====
C
C PROGRAMME PRINCIPAL PERMETTANT DE TESTER LE SOUS-PROGRAMME
C LECPAR DE LECTURE DE PARAMETRES EN FORMAT LIBRE
C *****
C
C H. BIGOT
C UNITE STATISTIQUE - CENTRE DE CALCUL
C CNRS - STRASBOURG
C (OCTOBRE 1988)
C
C MODULAD
C VERSION X.X
C (DATE MODULAD)
C=====
C
C JEU-TEST :
C -----
C LE JEU-TEST UTILISE EST LE SUIVANT, OU IL SUFFIT D'ENLEVER LE C
C EN COLONNE 1 SUR LES 6 ENREGISTREMENTS SITUES ENTRE LES DEUX
C LIGNES D'ASTERISQUES :
C*****
C 1789 1831 -5.22 0.14 5E3 -3.2E7 8E-2
C 4- 1 9 3 1415 3,14 5+7 4E2.2 E=MC2 ETC ...
C
C PREMIER TITRE
C
C CET ENCORE CELUI-LA ... (TITRE 3)
C*****
C=====
C
C REAL PARR(40)
C INTEGER PARI(40)
C CHARACTER*80 PARC(40),TITRE
C COMMON /LIMIT/ EPS,REMAX,INTMAX
C INP = 5
C IOU = 8
C EPS = 1E-8
C REAMAX = 1E38
C INTMAX = 2**30 - 1 + 2**30
C OPEN (INP)
C OPEN (IOU)
C
C TEST SUR 3 ENREGISTREMENTS
C
C LRECL = 80
C NPAR = 9
C WRITE(IOU,'(1X,26HTEST SUR 3 ENREGISTREMENTS)')
C WRITE(IOU,'(1X,26H-----)')
C DO 10 J=1,3
C WRITE(IOU,'(//1X,29HNUMERO DE L ENREGISTREMENT : ,I3)') J
C
C CALL LECPAR(INP,LRECL,NPAR, NCOD,PARI,PARR,PARC)
C
C WRITE(IOU,'(1X,30HNOMBRE DE PARAMETRES SAISIS = ,I3)') NCOD
C WRITE(IOU,'(1X,22HVALORISATIONS FAITES :)')
C WRITE (IOU,1000) (I,PARI(I),I,PARR(I),I,PARC(I),I=1,9)
1000 FORMAT(9(1X,'INT(',I1,')=',I11,
1 3X,'REAL(',I1,')=',E10.4,
2 3X,'CHAR(',I1,')=',A25/))
10 CONTINUE
```

```
C
C      UTILISATION POUR LA LECTURE D'UN TITRE
C
C      TOUS LES NCOD PARAMETRES CARACTERES FORMENT LE TITRE :
C
      LRECL = 80
      NPAR = 40
      WRITE(IOU,'(1H1,1X,17HTEST SUR 3 TITRES)')
      WRITE(IOU,'(1X,17H-----)')
      DO 50 K= 1,3
C
      CALL LECPAR(INP,LRECL,NPAR, NCOD,PARI,PARR,PARC)
C
      IF (NCOD.EQ.0) THEN
          TITRE = 'TITRE NON CODE'
      ELSE
          DO 20 L = 1,80
              TITRE(L:L) = ' '
          CONTINUE
          KK = 0
          DO 40 I = 1,NCOD
              DO 30 J = 1,80
                  KK = KK + 1
                  IF (PARC(I)(J:J).EQ.' ') GO TO 40
                  TITRE(KK:KK) = PARC(I)(J:J)
              CONTINUE
          CONTINUE
          CONTINUE
          ENDIF
          WRITE(IOU,'(//,1X,21HNUMERO DU TITRE LU = ,I2)') K
          WRITE(IOU,'(1X,A79)') TITRE
          CONTINUE
          STOP
          END
      20
      30
      40
      50
```

ANNEXE 2 : listing résultant de l'exemple précédent

TEST SUR 3 ENREGISTREMENTS

NUMERO DE L ENREGISTREMENT : 1
NOMBRE DE PARAMETRES SAISIS = 8
VALORISATIONS FAITES :

INT(1)=	1789	REAL(1)=0.1789E+04	CHAR(1)=1789
INT(2)=	2147483647	REAL(2)=0.1831E+04	CHAR(2)=1831.
INT(3)=	2147483647	REAL(3)=-.5220E+01	CHAR(3)=-5.22
INT(4)=-	2147483647	REAL(4)=-.1000E+39	CHAR(4)=.
INT(5)=	2147483647	REAL(5)=0.1400E+00	CHAR(5)=0.14
INT(6)=	2147483647	REAL(6)=0.5000E+04	CHAR(6)=5E3
INT(7)=	2147483647	REAL(7)=-.3200E+08	CHAR(7)=-3.2E7
INT(8)=	2147483647	REAL(8)=0.8000E-01	CHAR(8)=8E-2
INT(9)=	0	REAL(9)=0.0000E+00	CHAR(9)=NON CODE

NUMERO DE L ENREGISTREMENT : 2
NOMBRE DE PARAMETRES SAISIS = 9
VALORISATIONS FAITES :

INT(1)=	2147483647	REAL(1)=0.1000E+39	CHAR(1)=4-
INT(2)=	1	REAL(2)=0.1000E+01	CHAR(2)=1
INT(3)=	2147483647	REAL(3)=0.1000E+39	CHAR(3)=.
INT(4)=	2147483647	REAL(4)=0.9000E+00	CHAR(4)=.9
INT(5)=	2147483647	REAL(5)=0.3141E+01	CHAR(5)=3.1415
INT(6)=	2147483647	REAL(6)=0.1000E+39	CHAR(6)=3,14
INT(7)=	2147483647	REAL(7)=0.1000E+39	CHAR(7)=5+7
INT(8)=	2147483647	REAL(8)=0.1000E+39	CHAR(8)=4E2.2
INT(9)=	2147483647	REAL(9)=0.1000E+39	CHAR(9)=E=MC2

NUMERO DE L ENREGISTREMENT : 3
NOMBRE DE PARAMETRES SAISIS = 0
VALORISATIONS FAITES :

INT(1)=	0	REAL(1)=0.0000E+00	CHAR(1)=NON CODE
INT(2)=	0	REAL(2)=0.0000E+00	CHAR(2)=NON CODE
INT(3)=	0	REAL(3)=0.0000E+00	CHAR(3)=NON CODE
INT(4)=	0	REAL(4)=0.0000E+00	CHAR(4)=NON CODE
INT(5)=	0	REAL(5)=0.0000E+00	CHAR(5)=NON CODE
INT(6)=	0	REAL(6)=0.0000E+00	CHAR(6)=NON CODE
INT(7)=	0	REAL(7)=0.0000E+00	CHAR(7)=NON CODE
INT(8)=	0	REAL(8)=0.0000E+00	CHAR(8)=NON CODE
INT(9)=	0	REAL(9)=0.0000E+00	CHAR(9)=NON CODE

TEST SUR 3 TITRES

NUMERO DU TITRE LU = 1
PREMIER TITRE
NUMERO DU TITRE LU = 2
TITRE NON CODE
NUMERO DU TITRE LU = 3
ET ENCORE CELUI-LA ... (TITRE 3)