X04EAF – NAG Fortran Library Routine Document

Note. Before using this routine, please read the Users' Note for your implementation to check the interpretation of bold italicised terms and other implementation-dependent details.

1 Purpose

X04EAF is an easy-to-use routine to print an integer matrix stored in a two-dimensional array.

2 Specification

```
SUBROUTINE X04EAF(MATRIX, DIAG, M, N, A, LDA, TITLE, IFAIL)
INTEGER           M, N, A(LDA,*), LDA, IFAIL
CHARACTER*1       MATRIX, DIAG
CHARACTER*(*)     TITLE
```

3 Description

X04EAF prints an integer matrix. It is an easy-to-use driver for X04EBF. The routine uses default values for the format in which numbers are printed, for labelling the rows and columns, and for output record length.

X04EAF will choose a format code such that numbers will be printed with the smallest I edit descriptor that is large enough to hold all the numbers to be printed.

The matrix is printed with integer row and column labels, and with a maximum record length of 80.

The matrix is output to the unit defined by X04ABF.

4 References

None.

5 Parameters

1: MATRIX — CHARACTER*1 *Input*

On entry: indicates the part of the matrix to be printed, as follows:

MATRIX = 'G' (General), the whole of the rectangular matrix.

MATRIX = 'L' (Lower), the lower triangle of the matrix, or the lower trapezium if the matrix has more rows than columns.

MATRIX = 'U' (Upper), the upper triangle of the matrix, or the upper trapezium if the matrix has more columns than rows.

Constraint: MATRIX must be one of 'G', 'L' or 'U'.

2: DIAG — CHARACTER*1 *Input*

On entry: unless MATRIX = 'G', DIAG must specify whether the diagonal elements of the matrix are to be printed, as follows:

DIAG = 'B' (Blank), the diagonal elements of the matrix are not referenced and not printed.

DIAG = 'U' (Unit diagonal), the diagonal elements of the matrix are not referenced, but are assumed all to be unity, and are printed as such.

DIAG = 'N' (Non-unit diagonal), the diagonal elements of the matrix are referenced and printed.

If MATRIX = 'G', then DIAG need not be set.

Constraint: If MATRIX $\neq$ 'G', then DIAG must be one of 'B', 'U' or 'N'.

3: M — INTEGER *Input*

4: N — INTEGER *Input*

On entry: the number of rows and columns of the matrix, respectively, to be printed.

If either of M or N is less than 1, X04EAF will exit immediately after printing TITLE; no row or column labels are printed.

5: A(LDA,$*$) — INTEGER array *Input*

The second dimension of the array A must be at least max(1,N).

On entry: the matrix to be printed. Only the elements that will be referred to, as specified by parameters MATRIX and DIAG, need be set.

6: LDA — INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which X04EAF is called.

Constraint: LDA $\geq$ M.

7: TITLE — CHARACTER*(*) *Input*

On entry: a title to be printed above the matrix. If TITLE = ' ', no title (and no blank line) will be printed.

If TITLE contains more than 80 characters, the contents of TITLE will be wrapped onto more than one line, with the break after 80 characters.

Any trailing blank characters in TITLE are ignored.

8: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors detected by the routine:

IFAIL = 1

On entry, MATRIX $\neq$ 'G', 'L' or 'U'.

IFAIL = 2

On entry, MATRIX = 'L' or 'U', but DIAG $\neq$ 'N', 'U' or 'B'.

IFAIL = 3

On entry, LDA $<$ M.

7 Accuracy

Not applicable.

8 Further Comments

A call to X04EAF is equivalent to a call to X04EBF with the following argument values:

```
NCOLS = 80
INDENT = 0
LABROW = 'I'
LABCOL = 'I'
FORMAT = ' '
```

9 Example

This example program calls X04EAF twice, first to print a 3 by 5 rectangular matrix, and then to print a 5 by 5 triangular matrix.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     X04EAF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      INTEGER          NMAX, LDA
      PARAMETER        (NMAX=5,LDA=NMAX)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J
*     .. Local Arrays ..
      INTEGER          A(LDA,NMAX)
*     .. External Subroutines ..
      EXTERNAL         X04EAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'X04EAF Example Program Results'
      WRITE (NOUT,*)
*     Generate an array of data
      DO 40 J = 1, NMAX
         DO 20 I = 1, LDA
            A(I,J) = 10*I + J
   20    CONTINUE
   40 CONTINUE
*
      IFAIL = 0
*
*     Print 3 by 5 rectangular matrix
      CALL X04EAF('General',' ',3,5,A,LDA,'Example 1:',IFAIL)
*
      WRITE (NOUT,*)
*
*     Print 5 by 5 lower triangular matrix
      CALL X04EAF('Lower','Non-unit',5,5,A,LDA,'Example 2:',IFAIL)
*
      STOP
      END
```

9.2 Program Data

None.

9.3 Program Results

```
X04EAF Example Program Results

Example 1:
    1  2  3  4  5
1  11 12 13 14 15
2  21 22 23 24 25
3  31 32 33 34 35

Example 2:
    1  2  3  4  5
1  11
2  21 22
3  31 32 33
4  41 42 43 44
5  51 52 53 54 55
```