NAG Fortran Library Routine Document

F04AJF

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

F04AJF calculates the approximate solution of a set of real linear equations with multiple right-hand sides, $AX = B$, where A has been factorized by F03AFF.

2 Specification

```
SUBROUTINE F04AJF(N, IR, A, IA, P, B, IB)
INTEGER           N, IR, IA, IB
real              A(IA,N), P(N), B(IB,IR)
```

3 Description

To solve a set of real linear equations $AX = B$, the routine must be preceded by a call to F03AFF which computes an LU factorization of A with partial pivoting, $PA = LU$, where P is a permutation matrix, L is lower triangular and U is unit upper triangular. The columns x of the solution X are found by forward and backward substitution in $Ly = Pb$ and $Ux = y$, where b is a column of the right-hand sides.

4 References

Wilkinson J H and Reinsch C (1971) *Handbook for Automatic Computation II, Linear Algebra* Springer-Verlag

5 Parameters

1: N – INTEGER *Input*

On entry: n, the order of the matrix A.

2: IR – INTEGER *Input*

On entry: r, the number of right-hand sides.

3: A(IA,N) – ***real*** array *Input*

On entry: details of the LU factorization, as returned by F03AFF.

4: IA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F04AJF is called.

Constraint: IA $\geq$ N.

5: P(N) – ***real*** array *Input*

On entry: details of the row interchanges as returned by F03AFF.

6: B(IB,IR) – ***real*** array *Input/Output*

On entry: the n by r right-hand side matrix B.

On exit: B is overwritten by the solution matrix X.

7: IB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F04AJF is called.

Constraint: IB $\geq$ N.

6 Error Indicators and Warnings

None.

7 Accuracy

The accuracy of the computed solutions depends on the conditioning of the original matrix. For a detailed error analysis see page 106 of Wilkinson and Reinsch (1971).

8 Further Comments

The time taken by the routine is approximately proportional to $n^2 r$.

9 Example

To solve the set of linear equations $AX = B$ where

$$A = \begin{pmatrix} 33 & 16 & 72 \\ -24 & -10 & -57 \\ -8 & -4 & -17 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -359 \\ 281 \\ 85 \end{pmatrix}.$$

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.

```
*     F04AJF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NMAX, IR, IA, IB
      PARAMETER        (NMAX=8,IR=1,IA=NMAX,IB=NMAX)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             D1, EPS
      INTEGER          I, ID, IFAIL, J, N
*     .. Local Arrays ..
      real             A(IA,NMAX), B(IB,IR), P(NMAX)
*     .. External Functions ..
      real             X02AJF
      EXTERNAL         X02AJF
*     .. External Subroutines ..
      EXTERNAL         F03AFF, F04AJF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F04AJF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      WRITE (NOUT,*)
      IF (N.GT.0 .AND. N.LE.NMAX) THEN
         READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
         IFAIL = 1
         EPS = X02AJF()
*
*        Crout decomposition
         CALL F03AFF(N,EPS,A,IA,D1,ID,P,IFAIL)
*
```

```
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'Error in F03AFF. IFAIL =', IFAIL
         ELSE
            READ (NIN,*) ((B(I,J),J=1,IR),I=1,N)
*
*           Approximate solution of linear equations
            CALL F04AJF(N,IR,A,IA,P,B,IB)
*
            WRITE (NOUT,*) ' Solution'
            DO 20 I = 1, N
               WRITE (NOUT,99999) (B(I,J),J=1,IR)
   20       CONTINUE
         END IF
      ELSE
         WRITE (NOUT,99998) 'N is out of range: N = ', N
      END IF
      STOP
*
99999 FORMAT (1X,8F9.4)
99998 FORMAT (1X,A,I5)
      END
```

9.2 Program Data

```
F04AJF Example Program Data
  3
   33   16   72
  -24  -10  -57
   -8   -4  -17
 -359  281   85
```

9.3 Program Results

```
 F04AJF Example Program Results

  Solution
    1.0000
   -2.0000
   -5.0000
```