F07ASF (CGETRS/ZGETRS) – 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

F07ASF (CGETRS/ZGETRS) solves a complex system of linear equations with multiple right-hand sides, $AX = B$, $A^TX = B$ or $A^HX = B$, where A has been factorized by F07ARF (CGETRF/ZGETRF).

2 Specification

```
SUBROUTINE F07ASF(TRANS, N, NRHS, A, LDA, IPIV, B, LDB, INFO)
ENTRY      cgetrs(TRANS, N, NRHS, A, LDA, IPIV, B, LDB, INFO)
INTEGER            N, NRHS, LDA, IPIV(*), LDB, INFO
complex            A(LDA,*), B(LDB,*)
CHARACTER*1        TRANS
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

To solve a complex system of linear equations $AX = B$, $A^TX = B$ or $A^HX = B$, this routine must be preceded by a call to F07ARF (CGETRF/ZGETRF) which computes the LU factorization of A as $A = PLU$. The solution is computed by forward and backward substitution.

If TRANS = 'N', the solution is computed by solving $PLY = B$ and then $UX = Y$.

If TRANS = 'T', the solution is computed by solving $U^TY = B$ and then $L^TP^TX = Y$.

If TRANS = 'C', the solution is computed by solving $U^HY = B$ and then $L^HP^TX = Y$.

4 References

[1] Golub G H and van Loan C F (1996) *Matrix Computations* Johns Hopkins University Press (3rd Edition), Baltimore

5 Parameters

1: TRANS — CHARACTER*1 *Input*

On entry: indicates the form of the equations as follows:

if TRANS = 'N', then $AX = B$ is solved for X;
if TRANS = 'T', then $A^TX = B$ is solved for X;
if TRANS = 'C', then $A^HX = B$ is solved for X.

Constraint: TRANS = 'N', 'T' or 'C'.

2: N — INTEGER *Input*

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

Constraint: N ≥ 0.

3: NRHS — INTEGER *Input*

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

Constraint: NRHS ≥ 0.

4: A(LDA,∗) — ***complex*** array *Input*

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

On entry: the *LU* factorization of A, as returned by F07ARF (CGETRF/ZGETRF).

5: LDA — INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07ASF (CGETRS/ZGETRS) is called.

Constraint: LDA $\geq$ max(1,N).

6: IPIV(∗) — INTEGER array *Input*

Note: the dimension of the array IPIV must be at least max(1,N).

On entry: the pivot indices, as returned by F07ARF (CGETRF/ZGETRF).

7: B(LDB,∗) — ***complex*** array *Input/Output*

Note: the second dimension of the array B must be at least max(1,NRHS).

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

On exit: the n by r solution matrix X.

8: LDB — INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F07ASF (CGETRS/ZGETRS) is called.

Constraint: LDB $\geq$ max(1,N).

9: INFO — INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

If INFO $= -i$, the ith parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

For each right-hand side vector b, the computed solution x is the exact solution of a perturbed system of equations $(A + E)x = b$, where

$$|E| \leq c(n)\epsilon P|L||U|,$$

$c(n)$ is a modest linear function of n, and ϵ is the ***machine precision***.

If $\hat{x}$ is the true solution, then the computed solution x satisfies a forward error bound of the form

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq c(n)\text{cond}(A, x)\epsilon$$

where $\text{cond}(A, x) = \| \, |A^{-1}||A||x| \, \|_\infty / \|x\|_\infty \leq \text{cond}(A) = \| \, |A^{-1}||A| \, \|_\infty \leq \kappa_\infty(A)$. Note that $\text{cond}(A, x)$ can be much smaller than $\text{cond}(A)$, and $\text{cond}(A^H)$ (which is the same as $\text{cond}(A^T)$) can be much larger (or smaller) than $\text{cond}(A)$.

Forward and backward error bounds can be computed by calling F07AVF (CGERFS/ZGERFS), and an estimate for $\kappa_\infty(A)$ can be obtained by calling F07AUF (CGECON/ZGECON) with NORM = 'I'.

8 Further Comments

The total number of real floating-point operations is approximately $8n^2r$.

This routine may be followed by a call to F07AVF (CGERFS/ZGERFS) to refine the solution and return an error estimate.

The real analogue of this routine is F07AEF (SGETRS/DGETRS).

9 Example

To solve the system of equations $AX = B$, where

$$A = \begin{pmatrix} -1.34 + 2.55i & 0.28 + 3.17i & -6.39 - 2.20i & 0.72 - 0.92i \\ -0.17 - 1.41i & 3.31 - 0.15i & -0.15 + 1.34i & 1.29 + 1.38i \\ -3.29 - 2.39i & -1.91 + 4.42i & -0.14 - 1.35i & 1.72 + 1.35i \\ 2.41 + 0.39i & -0.56 + 1.47i & -0.83 - 0.69i & -1.96 + 0.67i \end{pmatrix}$$

and

$$B = \begin{pmatrix} 26.26{+}51.78i & 31.32{-}\ 6.70i \\ 6.43{-}\ 8.68i & 15.86{-}\ 1.42i \\ -5.75{+}25.31i & -2.15{+}30.19i \\ 1.16{+}\ 2.57i & -2.56{+}\ 7.55i \end{pmatrix}.$$

Here A is nonsymmetric and must first be factorized by F07ARF (CGETRF/ZGETRF).

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.

```
*     F07ASF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, LDA, NRHMAX, LDB
      PARAMETER        (NMAX=8,LDA=NMAX,NRHMAX=NMAX,LDB=NMAX)
      CHARACTER        TRANS
      PARAMETER        (TRANS='N')
*     .. Local Scalars ..
      INTEGER          I, IFAIL, INFO, J, N, NRHS
*     .. Local Arrays ..
      complex          A(LDA,NMAX), B(LDB,NRHMAX)
      INTEGER          IPIV(NMAX)
      CHARACTER        CLABS(1), RLABS(1)
*     .. External Subroutines ..
      EXTERNAL         cgetrf, cgetrs, X04DBF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07ASF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, NRHS
      IF (N.LE.NMAX .AND. NRHS.LE.NRHMAX) THEN
*
*        Read A and B from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
         READ (NIN,*) ((B(I,J),J=1,NRHS),I=1,N)
*
```

```
*        Factorize A
*
         CALL cgetrf(N,N,A,LDA,IPIV,INFO)
*
         WRITE (NOUT,*)
         IF (INFO.EQ.0) THEN
*
*           Compute solution
*
            CALL cgetrs(TRANS,N,NRHS,A,LDA,IPIV,B,LDB,INFO)
*
*           Print solution
*
            IFAIL = 0
            CALL X04DBF('General',' ',N,NRHS,B,LDB,'Bracketed','F7.4',
     +                  'Solution(s)','Integer',RLABS,'Integer',CLABS,
     +                  80,0,IFAIL)
         ELSE
            WRITE (NOUT,*) 'The factor U is singular'
         END IF
      END IF
      STOP
*
      END
```

9.2 Program Data

```
F07ASF Example Program Data
  4  2                                                    :Values of N and NRHS
 (-1.34, 2.55) ( 0.28, 3.17) (-6.39,-2.20) ( 0.72,-0.92)
 (-0.17,-1.41) ( 3.31,-0.15) (-0.15, 1.34) ( 1.29, 1.38)
 (-3.29,-2.39) (-1.91, 4.42) (-0.14,-1.35) ( 1.72, 1.35)
 ( 2.41, 0.39) (-0.56, 1.47) (-0.83,-0.69) (-1.96, 0.67)  :End of matrix A
 (26.26, 51.78) (31.32, -6.70)
 ( 6.43, -8.68) (15.86, -1.42)
 (-5.75, 25.31) (-2.15, 30.19)
 ( 1.16,  2.57) (-2.56,  7.55)                            :End of matrix B
```

9.3 Program Results

```
F07ASF Example Program Results

Solution(s)
                    1                 2
1  ( 1.0000, 1.0000) (-1.0000,-2.0000)
2  ( 2.0000,-3.0000) ( 5.0000, 1.0000)
3  (-4.0000,-5.0000) (-3.0000, 4.0000)
4  ( 0.0000, 6.0000) ( 2.0000,-3.0000)
```