NAG Library Routine Document

F07AEF (DGETRS)

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

F07AEF (DGETRS) solves a real system of linear equations with multiple right-hand sides,

$$AX = B \qquad \text{or} \qquad A^{\mathrm{T}}X = B,$$

where A has been factorized by F07ADF (DGETRF).

2 Specification

```
SUBROUTINE F07AEF (TRANS, N, NRHS, A, LDA, IPIV, B, LDB, INFO)

INTEGER             N, NRHS, LDA, IPIV(*), LDB, INFO
REAL (KIND=nag_wp)  A(LDA,*), B(LDB,*)
CHARACTER(1)        TRANS
```

The routine may be called by its LAPACK name ***dgetrs***.

3 Description

F07AEF (DGETRS) is used to solve a real system of linear equations $AX = B$ or $A^{\mathrm{T}}X = B$, the routine must be preceded by a call to F07ADF (DGETRF) 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' or 'C', the solution is computed by solving $U^{\mathrm{T}}Y = B$ and then $L^{\mathrm{T}}P^{\mathrm{T}}X = Y$.

4 References

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

5 Parameters

1: TRANS – CHARACTER(1) *Input*

On entry: indicates the form of the equations.

TRANS = 'N'
$AX = B$ is solved for X.

TRANS = 'T' or 'C'
$A^{\mathrm{T}}X = B$ 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,$*$) – REAL (KIND=nag_wp) array *Input*

Note: the second dimension of the array A must be at least $\max(1, \mathrm{N})$.

On entry: the LU factorization of A, as returned by F07ADF (DGETRF).

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07AEF (DGETRS) is called.

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

6: IPIV($*$) – INTEGER array *Input*

Note: the dimension of the array IPIV must be at least $\max(1, \mathrm{N})$.

On entry: the pivot indices, as returned by F07ADF (DGETRF).

7: B(LDB,$*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array B must be at least $\max(1, \mathrm{NRHS})$.

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 F07AEF (DGETRS) is called.

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

9: INFO – INTEGER *Output*

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

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

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

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

where $\mathrm{cond}(A, x) = \left\| |A^{-1}||A||x| \right\|_\infty / \|x\|_\infty \leq \mathrm{cond}(A) = \left\| |A^{-1}||A| \right\|_\infty \leq \kappa_\infty(A)$.

Note that cond(A,x) can be much smaller than cond(A), and cond$\left(A^{\mathrm{T}}\right)$ can be much larger (or smaller) than cond(A).

Forward and backward error bounds can be computed by calling F07AHF (DGERFS), and an estimate for $\kappa_\infty(A)$ can be obtained by calling F07AGF (DGECON) with NORM = 'I'.

8 Further Comments

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

This routine may be followed by a call to F07AHF (DGERFS) to refine the solution and return an error estimate.

The complex analogue of this routine is F07ASF (ZGETRS).

9 Example

This example solves the system of equations $AX = B$, where

$$A = \begin{pmatrix} 1.80 & 2.88 & 2.05 & -0.89 \\ 5.25 & -2.95 & -0.95 & -3.80 \\ 1.58 & -2.69 & -2.90 & -1.04 \\ -1.11 & -0.66 & -0.59 & 0.80 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 9.52 & 18.47 \\ 24.35 & 2.25 \\ 0.77 & -13.28 \\ -6.22 & -6.21 \end{pmatrix}.$$

Here A is nonsymmetric and must first be factorized by F07ADF (DGETRF).

9.1 Program Text

```
    Program f07aefe

!     F07AEF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: dgetrf, dgetrs, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
      Character (1), Parameter         :: trans = 'N'
!     .. Local Scalars ..
      Integer                          :: i, ifail, info, lda, ldb, n, nrhs
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:,:), b(:,:)
      Integer, Allocatable             :: ipiv(:)
!     .. Executable Statements ..
      Write (nout,*) 'F07AEF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, nrhs
      lda = n
      ldb = n
      Allocate (a(lda,n),b(ldb,nrhs),ipiv(n))

!     Read A and B from data file

      Read (nin,*)(a(i,1:n),i=1,n)
      Read (nin,*)(b(i,1:nrhs),i=1,n)

!     Factorize A

!     The NAG name equivalent of dgetrf is f07adf
      Call dgetrf(n,n,a,lda,ipiv,info)

      Write (nout,*)
      Flush (nout)
      If (info==0) Then

!       Compute solution

!       The NAG name equivalent of dgetrs is f07aef
        Call dgetrs(trans,n,nrhs,a,lda,ipiv,b,ldb,info)

!       Print solution

!       ifail: behaviour on error exit
!              =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
        ifail = 0
        Call x04caf('General',' ',n,nrhs,b,ldb,'Solution(s)',ifail)
      Else
        Write (nout,*) 'The factor U is singular'
      End If

    End Program f07aefe
```

9.2 Program Data

```
F07AEF Example Program Data
  4  2                              :Values of N and NRHS
  1.80   2.88   2.05  -0.89
  5.25  -2.95  -0.95  -3.80
  1.58  -2.69  -2.90  -1.04
 -1.11  -0.66  -0.59   0.80   :End of matrix A
  9.52  18.47
 24.35   2.25
  0.77 -13.28
 -6.22  -6.21                 :End of matrix B
```

9.3 Program Results

```
 F07AEF Example Program Results

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