NAG Library Routine Document

F08FEF (DSYTRD)

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

F08FEF (DSYTRD) reduces a real symmetric matrix to tridiagonal form.

2   Specification

```
SUBROUTINE F08FEF (UPLO, N, A, LDA, D, E, TAU, WORK, LWORK, INFO)
INTEGER           N, LDA, LWORK, INFO
REAL (KIND=nag_wp) A(LDA,*), D(*), E(*), TAU(*), WORK(max(1,LWORK))
CHARACTER(1)      UPLO
```

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

3   Description

F08FEF (DSYTRD) reduces a real symmetric matrix A to symmetric tridiagonal form T by an orthogonal similarity transformation: $A = QTQ^{\mathrm{T}}$.

The matrix Q is not formed explicitly but is represented as a product of $n-1$ elementary reflectors (see the F08 Chapter Introduction for details). Routines are provided to work with Q in this representation (see Section 9).

4   References

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

5   Parameters

1:    UPLO – CHARACTER(1)   *Input*

    On entry: indicates whether the upper or lower triangular part of A is stored.

    UPLO = 'U'
        The upper triangular part of A is stored.

    UPLO = 'L'
        The lower triangular part of A is stored.

    Constraint: UPLO = 'U' or 'L'.

2:    N – INTEGER   *Input*

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

    Constraint: N $\geq$ 0.

3:    A(LDA, $*$) – REAL (KIND=nag_wp) array   *Input/Output*

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

    On entry: the n by n symmetric matrix A.

If UPLO = 'U', the upper triangular part of A must be stored and the elements of the array below the diagonal are not referenced.

If UPLO = 'L', the lower triangular part of A must be stored and the elements of the array above the diagonal are not referenced.

On exit: A is overwritten by the tridiagonal matrix T and details of the orthogonal matrix Q as specified by UPLO.

4:     LDA – INTEGER   *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08FEF (DSYTRD) is called.

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

5:     D(∗) – REAL (KIND=nag_wp) array   *Output*

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

On exit: the diagonal elements of the tridiagonal matrix T.

6:     E(∗) – REAL (KIND=nag_wp) array   *Output*

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

On exit: the off-diagonal elements of the tridiagonal matrix T.

7:     TAU(∗) – REAL (KIND=nag_wp) array   *Output*

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

On exit: further details of the orthogonal matrix Q.

8:     WORK(max(1, LWORK)) – REAL (KIND=nag_wp) array   *Workspace*

On exit: if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimal performance.

9:     LWORK – INTEGER   *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08FEF (DSYTRD) is called.

If LWORK = −1, a workspace query is assumed; the routine only calculates the optimal size of the WORK array, returns this value as the first entry of the WORK array, and no error message related to LWORK is issued.

Suggested value: for optimal performance, $\text{LWORK} \geq \text{N} \times nb$, where nb is the optimal **block size**.

Constraint: $\text{LWORK} \geq 1$ or $\text{LWORK} = -1$.

10:     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 = $-i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7   Accuracy

The computed tridiagonal matrix T is exactly similar to a nearby matrix $(A + E)$, where

$$\|E\|_2 \le c(n)\epsilon\|A\|_2,$$

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

The elements of T themselves may be sensitive to small perturbations in A or to rounding errors in the computation, but this does not affect the stability of the eigenvalues and eigenvectors.

8   Parallelism and Performance

F08FEF (DSYTRD) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F08FEF (DSYTRD) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9   Further Comments

The total number of floating-point operations is approximately $\frac{4}{3}n^3$.

To form the orthogonal matrix Q F08FEF (DSYTRD) may be followed by a call to F08FFF (DORGTR):

```
CALL DORGTR(UPLO,N,A,LDA,TAU,WORK,LWORK,INFO)
```

To apply Q to an n by p real matrix C F08FEF (DSYTRD) may be followed by a call to F08FGF (DORMTR). For example,

```
CALL DORMTR('Left',UPLO,'No Transpose',N,P,A,LDA,TAU,C,LDC, &
            WORK,LWORK,INFO)
```

forms the matrix product QC.

The complex analogue of this routine is F08FSF (ZHETRD).

10   Example

This example reduces the matrix A to tridiagonal form, where

$$A = \begin{pmatrix} 2.07 & 3.87 & 4.20 & -1.15 \\ 3.87 & -0.21 & 1.87 & 0.63 \\ 4.20 & 1.87 & 1.15 & 2.06 \\ -1.15 & 0.63 & 2.06 & -1.81 \end{pmatrix}.$$

10.1 Program Text

```
    Program f08fefe

!     F08FEF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: dsytrd, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter            :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                       :: i, info, lda, lwork, n
```

```
        Character (1)                          :: uplo
!       .. Local Arrays ..
        Real (Kind=nag_wp), Allocatable  :: a(:,:), d(:), e(:), tau(:), work(:)
!       .. Intrinsic Procedures ..
        Intrinsic                          :: abs
!       .. Executable Statements ..
        Write (nout,*) 'F08FEF Example Program Results'
!       Skip heading in data file
        Read (nin,*)
        Read (nin,*) n
        lda = n
        lwork = 64*n
        Allocate (a(lda,n),d(n),e(n-1),tau(n-1),work(lwork))

!       Read A from data file and copy A into C to store as a full matrix

        Read (nin,*) uplo
        If (uplo=='U') Then
          Read (nin,*)(a(i,i:n),i=1,n)
        Else If (uplo=='L') Then
          Read (nin,*)(a(i,1:i),i=1,n)
        End If


!       Reduce A to tridiagonal form
!       The NAG name equivalent of dsytrd is f08fef.
        Call dsytrd(uplo,n,a,lda,d,e,tau,work,lwork,info)

        If (info==0) Then
!         Print the diagonal and off-diagonal of tridiagonal T.
!         The absolute value of E is printed since this can vary by a change of
!         sign (correspondng to multiplying through a column of Q by -1).

          Write (nout,*)
          Write (nout,*) &
            'Diagonal and off-diagonal elements of tridiagonal form'
          Write (nout,*)
          Write (nout,99999) 'i', 'D', 'E'
          Do i = 1, n - 1
            Write (nout,99998) i, d(i), abs(e(i))
          End Do
          Write (nout,99998) n, d(n)

        Else
          Write (nout,99997) info
        End If

99999 Format (5X,A,9X,A,12X,A)
99998 Format (1X,I5,2(1X,F12.5))
99997 Format (1X,'** DSYTRD/F08FEF retuned with INFO = ',I10)

    End Program f08fefe
```

10.2 Program Data

```
F08FEF Example Program Data
  4                          :Value of N
  'L'                        :Value of UPLO
  2.07
  3.87  -0.21
  4.20   1.87   1.15
 -1.15   0.63   2.06  -1.81   :End of matrix A
```

10.3 Program Results

```
F08FEF Example Program Results

Diagonal and off-diagonal elements of tridiagonal form

     i         D             E
     1      2.07000       5.82575
     2      1.47409       2.62405
     3     -0.64916       0.91627
     4     -1.69493
```