

# NAG Fortran Library Routine Document

## F08JLF (DSTEGR)

**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

F08JLF (DSTEGR) computes all the eigenvalues and, optionally, all the eigenvectors of a real  $n$  by  $n$  symmetric tridiagonal matrix.

### 2 Specification

```

SUBROUTINE F08JLF (JOBZ, RANGE, N, D, E, VL, VU, IL, IU, ABSTOL, M, W,
1              Z, LDZ, ISUPPZ, WORK, LWORK, IWORK, LIWORK, INFO)
    INTEGER          N, IL, IU, M, LDZ, ISUPPZ(*), LWORK, IWORK(*),
1              LIWORK, INFO
    double precision D(*), E(*), VL, VU, ABSTOL, W(*), Z(LDZ,*), WORK(*)
    CHARACTER*1     JOBZ, RANGE

```

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

### 3 Description

F08JLF (DSTEGR) computes all the eigenvalues, and optionally the eigenvectors, of a real symmetric tridiagonal matrix  $T$ . That is, the routine computes the spectral factorization of  $T$  given by

$$T = ZAZ^T,$$

where  $A$  is a diagonal matrix whose diagonal elements are the eigenvalues,  $\lambda_i$ , of  $T$  and  $Z$  is an orthogonal matrix whose columns are the eigenvectors,  $z_i$ , of  $T$ . Thus

$$Tz_i = \lambda_i z_i, \quad i = 1, 2, \dots, n.$$

This routine uses the **dqds** and the **Relatively Robust Representation** algorithms to compute the eigenvalues and eigenvectors respectively; see for example Parlett and Dhillon (2000) and Dhillon and Parlett (2004) for further details. F08JLF (DSTEGR) can usually compute all the eigenvalues and eigenvectors in  $O(n^2)$  floating point operations and so, for large matrices, is often considerably faster than the other symmetric tridiagonal routines in this chapter when all the eigenvectors are required, particularly so compared to those routines that are based on the *QR* algorithm.

In the future this routine may be extended to allow for the computation of selected eigenvalues and eigenvectors.

### 4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia URL: <http://www.netlib.org/lapack/lug>

Barlow J and Demmel J W (1990) Computing accurate eigensystems of scaled diagonally dominant matrices *SIAM J. Numer. Anal.* **27** 762–791

Dhillon I S and Parlett B N (2004) Orthogonal eigenvectors and relative gaps. *SIAM J. Appl. Math.* **25** 858–899

Parlett B N and Dhillon I S (2000) Relatively robust representations of symmetric tridiagonals *Linear Algebra Appl.* **309** 121–151

## 5 Parameters

- 1: JOBZ – CHARACTER\*1 *Input*  
*On entry:* indicates whether eigenvectors are computed.  
 JOBZ = 'N'  
     Only eigenvalues are computed.  
 JOBZ = 'V'  
     Eigenvalues and eigenvectors are computed.  
*Constraint:* JOBZ = 'N' or 'V'.
- 2: RANGE – CHARACTER\*1 *Input*  
*On entry:* if RANGE = 'A', all eigenvalues will be found. In the future, other values of RANGE may be supported to allow for the computation of selected eigenvalues and eigenvectors, in which case VL, VU, IL and IU will also be utilized.  
*Constraint:* RANGE = 'A'.
- 3: N – INTEGER *Input*  
*On entry:*  $n$ , the order of the matrix  $T$ .  
*Constraint:*  $N \geq 0$ .
- 4: D(\*) – **double precision** array *Input/Output*  
**Note:** the dimension of the array D must be at least  $\max(1, N)$ .  
*On entry:* the  $n$  diagonal elements of the tridiagonal matrix  $T$ .  
*On exit:* is overwritten.
- 5: E(\*) – **double precision** array *Input/Output*  
**Note:** the dimension of the array E must be at least  $\max(1, N - 1)$ .  
*On entry:* the  $(n - 1)$  subdiagonal elements of the tridiagonal matrix  $T$ .  
*On exit:* is overwritten.
- 6: VL – **double precision** *Input*  
 7: VU – **double precision** *Input*  
*On entry:* VL and VU are not currently referenced. See RANGE.
- 8: IL – INTEGER *Input*  
 9: IU – INTEGER *Input*  
*On entry:* IL and IU are not currently referenced. See RANGE.
- 10: ABSTOL – **double precision** *Input*  
*On entry:* the absolute error tolerance for the eigenvalues and eigenvectors.  
 If JOBZ = 'V', the eigenvalues and eigenvectors will have residual norms bounded by ABSTOL, and the dot products between different eigenvectors will be bounded by ABSTOL, so that eigenvectors are orthogonal to within the tolerance given by ABSTOL.  
 If ABSTOL is less than  $n\epsilon\|T\|_1$ , then  $n\epsilon\|T\|_1$  will be used in its place, where  $\epsilon$  is the **machine precision**. The eigenvalues are computed to an accuracy of  $\epsilon\|T\|_1$  irrespective of ABSTOL.  
 If high relative accuracy is important, set ABSTOL to X02AMF(). See Barlow and Demmel (1990) for a discussion of which matrices define their eigenvalues to high relative accuracy.

- 11: M – INTEGER *Output*  
*On exit:* the total number of eigenvalues found.  
 If RANGE = 'A', M = N.
- 12: W(\*) – *double precision* array *Output*  
**Note:** the dimension of the array W must be at least max(1, N).  
*On exit:* the eigenvalues in ascending order.
- 13: Z(LDZ,\*) – *double precision* array *Output*  
**Note:** the second dimension of the array Z must be at least max(1, N).  
*On exit:* if JOBZ = 'V', then if INFO = 0, the columns of Z contain the orthonormal eigenvectors of the matrix T, with the *i*th column of Z holding the eigenvector associated with W(*i*).  
 If JOBZ = 'N', Z is not referenced.
- 14: LDZ – INTEGER *Input*  
*On entry:* the first dimension of the array Z as declared in the (sub)program from which F08JLF (DSTEGR) is called.  
*Constraints:*  
     if JOBZ = 'V', LDZ ≥ max(1, N);  
     LDZ ≥ 1 otherwise.
- 15: ISUPPZ(\*) – INTEGER array *Output*  
**Note:** the dimension of the array ISUPPZ must be at least max(1, 2 × M).  
*On exit:* the support of the eigenvectors in Z, i.e., the indices indicating the non-zero elements in Z. The *i*th eigenvector is non-zero only in elements ISUPPZ(2 × *i* – 1) through ISUPPZ(2 × *i*).
- 16: WORK(\*) – *double precision* array *Workspace*  
**Note:** the dimension of the array WORK must be at least max(1, 18 × N).  
*On exit:* if INFO = 0, WORK(1) returns the minimum LWORK.
- 17: LWORK – INTEGER *Input*  
*On entry:* the dimension of the array WORK as declared in the (sub)program from which F08JLF (DSTEGR) is called.  
 If LWORK = –1, a workspace query is assumed; the routine only calculates the minimum sizes of the WORK and IWORK arrays, returns these values as the first entries of the WORK and IWORK arrays, and no error message related to LWORK or LIWORK is issued.  
*Constraint:* LWORK ≥ max(1, 18 × N) or LWORK = –1.
- 18: IWORK(\*) – INTEGER array *Workspace*  
**Note:** the dimension of the array IWORK must be at least max(1, 10 × N).  
*On exit:* if INFO = 0, WORK(1) returns the minimum LIWORK.
- 19: LIWORK – INTEGER *Input*  
*On entry:* the dimension of the array IWORK as declared in the (sub)program from which F08JLF (DSTEGR) is called.

If  $LIWORK = -1$ , a workspace query is assumed; the routine only calculates the minimum sizes of the  $WORK$  and  $IWORK$  arrays, returns these values as the first entries of the  $WORK$  and  $IWORK$  arrays, and no error message related to  $LWORK$  or  $LIWORK$  is issued.

*Constraint:*  $LIWORK \geq \max(1, 10 \times N)$  or  $LIWORK = -1$ .

20: 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  $i$ th parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0

If  $INFO = 1$ , the *dqds* algorithm failed to converge, if  $INFO = 2$ , inverse iteration failed to converge.

## 7 Accuracy

See the description for  $ABSTOL$ . See also Section 4.7 of Anderson *et al.* (1999) and Barlow and Demmel (1990) for further details.

## 8 Further Comments

The total number of floating point operations required to compute all the eigenvalues and eigenvectors is approximately proportional to  $n^2$ .

The complex analogue of this routine is F08JYF (ZSTEGR).

## 9 Example

This example finds all the eigenvalues and eigenvectors of the symmetric tridiagonal matrix

$$T = \begin{pmatrix} 1.0 & 1.0 & 0 & 0 \\ 1.0 & 4.0 & 2.0 & 0 \\ 0 & 2.0 & 9.0 & 3.0 \\ 0 & 0 & 3.0 & 16.0 \end{pmatrix}.$$

$ABSTOL$  is set to zero so that the default tolerance of  $n\epsilon\|T\|_1$  is used.

### 9.1 Program Text

```
* F08JLF Example Program Text
* Mark 21 Release. NAG Copyright 2004.
* .. Parameters ..
INTEGER          NIN, NOUT
PARAMETER       (NIN=5,NOUT=6)
INTEGER          NMAX
PARAMETER       (NMAX=10)
INTEGER          LDZ, LIWORK, LWORK
PARAMETER       (LDZ=NMAX,LIWORK=10*NMAX,LWORK=18*NMAX)
INTEGER          IL, IU
PARAMETER       (IL=0,IU=0)
DOUBLE PRECISION VL, VU
PARAMETER       (VL=0.0D0,VU=0.0D0)
CHARACTER       RANGE
```

```

PARAMETER      (RANGE='A')
* .. Local Scalars ..
DOUBLE PRECISION ABSTOL
INTEGER         I, IFAIL, INFO, M, N
CHARACTER      JOBZ
* .. Local Arrays ..
DOUBLE PRECISION D(NMAX), E(NMAX-1), W(NMAX), WORK(LWORK),
+              Z(LDZ,NMAX)
INTEGER        ISUPPZ(2*NMAX), IWORK(LIWORK)
* .. External Subroutines ..
EXTERNAL       DSTEGR, X04CAF
* .. Executable Statements ..
WRITE (NOUT,*) 'F08JLF Example Program Results'
WRITE (NOUT,*)
* Skip heading in data file
READ (NIN,*)
READ (NIN,*) N
IF (N.LE.NMAX) THEN
*
*   Read the symmetric tridiagonal matrix T from data file, first
*   the diagonal elements, then the off diagonal elements and then
*   JOBZ ('N' - eigenvalues only, 'V' - vectors as well)
*
READ (NIN,*) (D(I),I=1,N)
READ (NIN,*) (E(I),I=1,N-1)
READ (NIN,*) JOBZ
*
*   Calculate all the eigenvalues of T. Set ABSTOL to zero so that
*   the default value is used.
*
ABSTOL = 0.0D0
CALL DSTEGR(JOBZ,RANGE,N,D,E,VL,VU,IL,IU,ABSTOL,M,W,Z,LDZ,
+          ISUPPZ,WORK,LWORK,IWORK,LIWORK,INFO)
*
IF (INFO.EQ.0) THEN
*
*   Print eigenvalues and eigenvectors
*
WRITE (NOUT,*) 'Eigenvalues'
WRITE (NOUT,99999) (W(I),I=1,M)
*
WRITE (NOUT,*)
IFAIL = 0
CALL X04CAF('General',' ',N,M,Z,LDZ,'Eigenvectors',IFAIL)
*
ELSE
WRITE (NOUT,99998)
+   'Failure to compute an eigenvalue, INFO = ', INFO
END IF
ELSE
WRITE (NOUT,*) 'NMAX too small'
END IF
STOP
*
99999 FORMAT ((3X,8F8.4))
99998 FORMAT (1X,A,I10)
END

```

## 9.2 Program Data

F08JLF Example Program Data

```

4                               :Value of N

1.0  4.0  9.0  16.0           :End of D
1.0  2.0  3.0                 :End of E

'V'                             :Value of JOBZ

```

### 9.3 Program Results

F08JLF Example Program Results

Eigenvalues

0.6476 3.5470 8.6578 17.1477

Eigenvectors

	1	2	3	4
1	0.9396	-0.3388	0.0494	0.0034
2	-0.3311	-0.8628	0.3781	0.0545
3	0.0853	0.3648	0.8558	0.3568
4	-0.0167	-0.0879	-0.3497	0.9326

---