NAG Library Routine Document

F08JJF (DSTEBZ)

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

F08JJF (DSTEBZ) computes some (or all) of the eigenvalues of a real symmetric tridiagonal matrix, by bisection.

2 Specification

```
SUBROUTINE F08JJF (RANGE, ORDER, N, VL, VU, IL, IU, ABSTOL, D, E, M,            &
                   NSPLIT, W, IBLOCK, ISPLIT, WORK, IWORK, INFO)

INTEGER            N, IL, IU, M, NSPLIT, IBLOCK(N), ISPLIT(N), IWORK(3*N),      &
                   INFO
REAL (KIND=nag_wp) VL, VU, ABSTOL, D(*), E(*), W(N), WORK(4*N)
CHARACTER(1)       RANGE, ORDER
```

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

3 Description

F08JJF (DSTEBZ) uses bisection to compute some or all of the eigenvalues of a real symmetric tridiagonal matrix T.

It searches for zero or negligible off-diagonal elements of T to see if the matrix splits into block diagonal form:

$$T = \begin{pmatrix} T_1 & & & & & \\ & T_2 & & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & & & & & T_p \end{pmatrix}.$$

It performs bisection on each of the blocks T_i and returns the block index of each computed eigenvalue, so that a subsequent call to F08JKF (DSTEIN) to compute eigenvectors can also take advantage of the block structure.

4 References

Kahan W (1966) Accurate eigenvalues of a symmetric tridiagonal matrix *Report CS41* Stanford University

5 Parameters

1: RANGE – CHARACTER(1) *Input*

On entry: indicates which eigenvalues are required.

RANGE = 'A'
All the eigenvalues are required.

RANGE = 'V'
All the eigenvalues in the half-open interval (VL,VU] are required.

RANGE = 'I'
Eigenvalues with indices IL to IU are required.

Constraint: RANGE = 'A', 'V' or 'I'.

2: ORDER – CHARACTER(1) *Input*

On entry: indicates the order in which the eigenvalues and their block numbers are to be stored.

ORDER = 'B'
The eigenvalues are to be grouped by split-off block and ordered from smallest to largest within each block.

ORDER = 'E'
The eigenvalues for the entire matrix are to be ordered from smallest to largest.

Constraint: ORDER = 'B' or 'E'.

3: N – INTEGER *Input*

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

Constraint: $\mathrm{N} \geq 0$.

4: VL – REAL (KIND=nag_wp) *Input*
5: VU – REAL (KIND=nag_wp) *Input*

On entry: if RANGE = 'V', the lower and upper bounds, respectively, of the half-open interval (VL,VU] within which the required eigenvalues lie.

If RANGE = 'A' or 'I', VL is not referenced.

Constraint: if RANGE = 'V', $\mathrm{VL} < \mathrm{VU}$.

6: IL – INTEGER *Input*
7: IU – INTEGER *Input*

On entry: if RANGE = 'I', the indices of the first and last eigenvalues, respectively, to be computed (assuming that the eigenvalues are in ascending order).

If RANGE = 'A' or 'V', IL is not referenced.

Constraint: if RANGE = 'I', $1 \leq \mathrm{IL} \leq \mathrm{IU} \leq \mathrm{N}$.

8: ABSTOL – REAL (KIND=nag_wp) *Input*

On entry: the absolute tolerance to which each eigenvalue is required. An eigenvalue (or cluster) is considered to have converged if it lies in an interval of width $\leq$ ABSTOL. If ABSTOL ≤ 0.0, then the tolerance is taken as ***machine precision*** $\times \|T\|_1$.

9: D($*$) – REAL (KIND=nag_wp) array *Input*

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

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

10: E($*$) – REAL (KIND=nag_wp) array *Input*

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

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

11: M – INTEGER *Output*

On exit: m, the actual number of eigenvalues found.

12: NSPLIT – INTEGER *Output*

On exit: the number of diagonal blocks which constitute the tridiagonal matrix T.

13: W(N) – REAL (KIND=nag_wp) array *Output*

On exit: the required eigenvalues of the tridiagonal matrix T stored in $\mathrm{W}(1)$ to $\mathrm{W}(m)$.

14: IBLOCK(N) – INTEGER array *Output*

On exit: at each row/column j where $\mathrm{E}(j)$ is zero or negligible, T is considered to split into a block diagonal matrix and $\mathrm{IBLOCK}(i)$ contains the block number of the eigenvalue stored in $\mathrm{W}(i)$, for $i = 1, 2, \ldots, m$. Note that $\mathrm{IBLOCK}(i) < 0$ for some i whenever INFO $= 1$ or 3 (see Section 6) and RANGE = 'A' or 'V'.

15: ISPLIT(N) – INTEGER array *Output*

On exit: the leading NSPLIT elements contain the points at which T splits up into sub-matrices as follows. The first sub-matrix consists of rows/columns 1 to $\mathrm{ISPLIT}(1)$, the second sub-matrix consists of rows/columns $\mathrm{ISPLIT}(1) + 1$ to $\mathrm{ISPLIT}(2)$, ..., and the NSPLIT(th) sub-matrix consists of rows/columns $\mathrm{ISPLIT}(\mathrm{NSPLIT} - 1) + 1$ to ISPLIT(NSPLIT) $(= n)$.

16: WORK($4 \times$ N) – REAL (KIND=nag_wp) array *Workspace*

17: IWORK($3 \times$ N) – INTEGER array *Workspace*

18: 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.

INFO $= 1$

If RANGE = 'A' or 'V', the algorithm failed to compute some (or all) of the required eigenvalues to the required accuracy. More precisely, $\mathrm{IBLOCK}(i) < 0$ indicates that eigenvalue i (stored in $\mathrm{W}(i)$) failed to converge.

INFO $= 2$

If RANGE = 'I', the algorithm failed to compute some (or all) of the required eigenvalues. Try calling the routine again with RANGE = 'A'.

INFO $= 3$

If RANGE = 'I', see the description above for INFO $= 2$.

If RANGE = 'A' or 'V', see the description above for INFO $= 1$.

INFO $= 4$

No eigenvalues have been computed. The floating point arithmetic on the computer is not behaving as expected.

If failures with INFO ≥ 1 are causing persistent trouble and you have checked that the routine is being called correctly, please contact NAG.

7 Accuracy

The eigenvalues of T are computed to high relative accuracy which means that if they vary widely in magnitude, then any small eigenvalues will be computed more accurately than, for example, with the standard QR method. However, the reduction to tridiagonal form (prior to calling the routine) may exclude the possibility of obtaining high relative accuracy in the small eigenvalues of the original matrix if its eigenvalues vary widely in magnitude.

8 Further Comments

There is no complex analogue of this routine.

9 Example

See Section 9 in F08FGF (DORMTR).