NAG Fortran Library Routine Document

F08LEF (SGBBRD/DGBBRD)

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

F08LEF (SGBBRD/DGBBRD) reduces a real m by n band matrix to upper bidiagonal form.

2 Specification

```
 SUBROUTINE F08LEF(VECT, M, N, NCC, KL, KU, AB, LDAB, D, E, Q, LDQ, PT,
1                  LDPT, C, LDC, WORK, INFO)
 ENTRY      sgbbrd (VECT, M, N, NCC, KL, KU, AB, LDAB, D, E, Q, LDQ, PT,
1                  LDPT, C, LDC, WORK, INFO)
 INTEGER           M, N, NCC, KL, KU, LDAB, LDQ, LDPT, LDC, INFO
 real              AB(LDAB,*), D(*), E(*), Q(LDQ,*), PT(LDPT,*),
1                  C(LDC,*), WORK(*)
 CHARACTER*1       VECT
```

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

3 Description

This routine reduces a real m by n band matrix to upper bidiagonal form B by an orthogonal transformation: $A = QBP^T$. The orthogonal matrices Q and P^T, of order m and n respectively, are determined as a product of Givens rotation matrices, and may be formed explicitly by the routine if required. A matrix C may also be updated to give $\tilde{C} = Q^T C$.

The routine uses a vectorisable form of the reduction.

4 References

None.

5 Parameters

1: VECT – CHARACTER*1 *Input*

On entry: indicates whether the matrices Q and/or P^T are generated:

if VECT = 'N', then neither Q nor P^T is generated;

if VECT = 'Q', then Q is generated;

if VECT = 'P', then P^T is generated;

if VECT = 'B', then both Q and P^T are generated.

Constraint: VECT = 'N', 'Q', 'P' or 'B'.

2: M – INTEGER *Input*

On entry: m, the number of rows of the matrix A.

Constraint: M $\geq$ 0.

3: N – INTEGER *Input*

On entry: n, the number of columns of the matrix A.

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

4: NCC – INTEGER *Input*

On entry: n_C, the number of columns of the matrix C.

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

5: KL – INTEGER *Input*

On entry: k_l, the number of sub-diagonals within the band of A.

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

6: KU – INTEGER *Input*

On entry: k_u, the number of super-diagonals within the band of A.

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

7: AB(LDAB,*) – ***real*** array *Input/Output*

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

On entry: the m by n band matrix A, stored in rows 1 to $k_l + k_u + 1$. More precisely, element a_{ij} must be stored in $\mathrm{AB}(k_u + 1 + i - j, j)$ for $\max(1, j - k_u) \leq i \leq \min(m, j + k_l)$.

On exit: A is overwritten by values generated during the reduction.

8: LDAB – INTEGER *Input*

On entry: the first dimension of the array AB as declared in the (sub)program from which F08LEF (SGBBRD/DGBBRD) is called.

Constraint: $\mathrm{LDAB} \geq \mathrm{KL} + \mathrm{KU} + 1$.

9: D($*$) – ***real*** array *Output*

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

On exit: the diagonal elements of the bidiagonal matrix B.

10: E($*$) – ***real*** array *Output*

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

On exit: the super-diagonal elements of the bidiagonal matrix B.

11: Q(LDQ,*) – ***real*** array *Output*

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

On exit: the m by m orthogonal matrix Q, if VECT = 'Q' or 'B'.

Q is not referenced if VECT = 'N' or 'P'.

12: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which F08LEF (SGBBRD/DGBBRD) is called.

Constraints:

LDQ $\geq \max(1, \mathrm{M})$ if VECT = 'Q' or 'B';
LDQ ≥ 1 otherwise.

13: PT(LDPT,*) – ***real*** array *Output*

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

On exit: the n by n orthogonal matrix P^T, if VECT = 'P' or 'B'.

PT is not referenced if VECT = 'N' or 'Q'.

14: LDPT – INTEGER *Input*

On entry: the first dimension of the array PT as declared in the (sub)program from which F08LEF (SGBBRD/DGBBRD) is called.

Constraints:

LDPT $\geq \max(1, \mathrm{N})$ if VECT = 'P' or 'B';
LDPT ≥ 1 otherwise.

15: C(LDC,*) – ***real*** array *Input/Output*

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

On entry: an m by n_C matrix C.

On exit: C is overwritten by $Q^T C$.

C is not referenced if NCC = 0.

16: LDC – INTEGER *Input*

On entry: the first dimension of the array C as declared in the (sub)program from which F08LEF (SGBBRD/DGBBRD) is called.

Constraints:

LDC $\geq \max(1, \mathrm{M})$ if NCC > 0,
LDC ≥ 1 if NCC $= 0$.

17: WORK($*$) – ***real*** array *Workspace*

Note: the dimension of the array WORK must be at least $2 * \max(\mathrm{M}, \mathrm{N})$.

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

7 Accuracy

The computed bidiagonal form B satisfies $QBP^T = A + E$, where

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

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

The elements of B 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 singular values and vectors.

The computed matrix Q differs from an exactly orthogonal matrix by a matrix F such that

$$\|F\|_2 = O(\epsilon).$$

A similar statement holds for the computed matrix P^T.

8 Further Comments

The total number of real floating-point operations is approximately the sum of:

$6n^2k$, if VECT = 'N' and NCC = 0, and

$3n^2n_C(k-1)/k$, if C is updated, and

$3n^3(k-1)/k$, if either Q or P^T is generated (double this if both),

where $k = k_l + k_u$, assuming $n \gg k$. For this section we assume that $m = n$.

The complex analogue of this routine is F08LSF (CGBBRD/ZGBBRD).

9 Example

To reduce the matrix A to upper bidiagonal form, where

$$A = \begin{pmatrix} -0.57 & -1.28 & 0.00 & 0.00 \\ -1.93 & 1.08 & -0.31 & 0.00 \\ 2.30 & 0.24 & 0.40 & -0.35 \\ 0.00 & 0.64 & -0.66 & 0.08 \\ 0.00 & 0.00 & 0.15 & -2.13 \\ -0.00 & 0.00 & 0.00 & 0.50 \end{pmatrix}.$$

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.

```
*     F08LEF Example Program Text.
*     Mark 19 Release. NAG Copyright 1999.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX, NCCMAX, KLMAX, KUMAX, LDAB, LDQ,
     +                 LDPT, LDC
      PARAMETER        (MMAX=8,NMAX=8,NCCMAX=8,KLMAX=8,KUMAX=8,
     +                 LDAB=KLMAX+KUMAX+1,LDQ=MMAX,LDPT=NMAX,LDC=MMAX)
      CHARACTER        VECT
      PARAMETER        (VECT='N')
*     .. Local Scalars ..
      INTEGER          I, INFO, J, KL, KU, M, N, NCC
*     .. Local Arrays ..
      real             AB(LDAB,NMAX), C(MMAX,NCCMAX), D(NMAX),
     +                 E(NMAX-1), PT(LDPT,NMAX), Q(LDQ,MMAX),
     +                 WORK(2*MMAX+2*NMAX)
*     .. External Subroutines ..
      EXTERNAL         sgbbrd
*     .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08LEF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N, KL, KU, NCC
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. KL.LE.KLMAX .AND. KU.LE.
     +    KUMAX .AND. NCC.LE.NCCMAX) THEN
*
*        Read A from data file
*
```

```
         READ (NIN,*) ((AB(KU+1+I-J,J),J=MAX(I-KL,1),MIN(I+KU,N)),I=1,M)
*
*        Reduce A to upper bidiagonal form
*
         CALL sgbbrd(VECT,M,N,NCC,KL,KU,AB,LDAB,D,E,Q,LDQ,PT,LDPT,C,LDC,
     +               WORK,INFO)
*
*        Print bidiagonal form
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Diagonal'
         WRITE (NOUT,99999) (D(I),I=1,MIN(M,N))
         WRITE (NOUT,*) 'Super-diagonal'
         WRITE (NOUT,99999) (E(I),I=1,MIN(M,N)-1)
      END IF
      STOP
*
99999 FORMAT (1X,8F9.4)
      END
```

9.2 Program Data

```
F08LEF Example Program Data
  6  4  2  1  0               :Values of M, N, KL, KU and NCC
 -0.57  -1.28
 -1.93   1.08  -0.31
  2.30   0.24   0.40  -0.35
         0.64  -0.66   0.08
                0.15  -2.13
                       0.50   :End of matrix A
```

9.3 Program Results

```
 F08LEF Example Program Results

 Diagonal
    3.0561   1.5259   0.9690   1.5685
 Super-diagonal
    0.6206  -1.2353  -1.1240
```