NAG Library Routine Document

F08KEF (DGEBRD)

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

F08KEF (DGEBRD) reduces a real m by n matrix to bidiagonal form.

2 Specification

```
SUBROUTINE F08KEF (M, N, A, LDA, D, E, TAUQ, TAUP, WORK, LWORK, INFO)

INTEGER             M, N, LDA, LWORK, INFO
REAL (KIND=nag_wp) A(LDA,*), D(*), E(*), TAUQ(*), TAUP(*),                &
                    WORK(max(1,LWORK))
```

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

3 Description

F08KEF (DGEBRD) reduces a real m by n matrix A to bidiagonal form B by an orthogonal transformation: $A = QBP^{\mathrm{T}}$, where Q and P^{T} are orthogonal matrices of order m and n respectively.

If $m \geq n$, the reduction is given by:

$$A = Q\begin{pmatrix} B_1 \\ 0 \end{pmatrix}P^{\mathrm{T}} = Q_1 B_1 P^{\mathrm{T}},$$

where B_1 is an n by n upper bidiagonal matrix and Q_1 consists of the first n columns of Q.

If $m < n$, the reduction is given by

$$A = Q\begin{pmatrix} B_1 & 0 \end{pmatrix}P^{\mathrm{T}} = QB_1 P_1^{\mathrm{T}},$$

where B_1 is an m by m lower bidiagonal matrix and P_1^{T} consists of the first m rows of P^{T}.

The orthogonal matrices Q and P are not formed explicitly but are represented as products of elementary reflectors (see the F08 Chapter Introduction for details). Routines are provided to work with Q and P in this representation (see Section 8).

4 References

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

5 Parameters

1: M – INTEGER *Input*

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

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

2: N – INTEGER *Input*

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

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

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 m by n matrix A.

On exit: if $m \geq n$, the diagonal and first superdiagonal are overwritten by the upper bidiagonal matrix B, elements below the diagonal are overwritten by details of the orthogonal matrix Q and elements above the first superdiagonal are overwritten by details of the orthogonal matrix P.

If $m < n$, the diagonal and first subdiagonal are overwritten by the lower bidiagonal matrix B, elements below the first subdiagonal are overwritten by details of the orthogonal matrix Q and elements above the diagonal are overwritten by details of the orthogonal matrix P.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08KEF (DGEBRD) is called.

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

5: D($*$) – REAL (KIND=nag_wp) 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.

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

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

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

7: TAUQ($*$) – REAL (KIND=nag_wp) array *Output*

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

On exit: further details of the orthogonal matrix Q.

8: TAUP($*$) – REAL (KIND=nag_wp) array *Output*

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

On exit: further details of the orthogonal matrix P.

9: WORK($\max(1, \mathrm{LWORK})$) – REAL (KIND=nag_wp) array *Workspace*

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

10: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08KEF (DGEBRD) is called.

If $\mathrm{LWORK} = -1$, 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, $\mathrm{LWORK} \geq (\mathrm{M} + \mathrm{N}) \times nb$, where nb is the optimal ***block size***.

Constraint: $\mathrm{LWORK} \geq \max(1, \mathrm{M}, \mathrm{N})$ or $\mathrm{LWORK} = -1$.

11: INFO – INTEGER *Output*

On exit: $\mathrm{INFO} = 0$ 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 $= -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed bidiagonal form B satisfies $QBP^{\mathrm{T}} = A + E$, 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.

8 Further Comments

The total number of floating point operations is approximately $\frac{4}{3}n^2(3m-n)$ if $m \geq n$ or $\frac{4}{3}m^2(3n-m)$ if $m < n$.

If $m \gg n$, it can be more efficient to first call F08AEF (DGEQRF) to perform a QR factorization of A, and then to call F08KEF (DGEBRD) to reduce the factor R to bidiagonal form. This requires approximately $2n^2(m+n)$ floating point operations.

If $m \ll n$, it can be more efficient to first call F08AHF (DGELQF) to perform an LQ factorization of A, and then to call F08KEF (DGEBRD) to reduce the factor L to bidiagonal form. This requires approximately $2m^2(m+n)$ operations.

To form the orthogonal matrices P^{T} and/or Q F08KEF (DGEBRD) may be followed by calls to F08KFF (DORGBR):

to form the m by m orthogonal matrix Q

```
CALL DORGBR('Q',M,M,N,A,LDA,TAUQ,WORK,LWORK,INFO)
```

but note that the second dimension of the array A must be at least M, which may be larger than was required by F08KEF (DGEBRD);

to form the n by n orthogonal matrix P^{T}

```
CALL DORGBR('P',N,N,M,A,LDA,TAUP,WORK,LWORK,INFO)
```

but note that the first dimension of the array A, specified by the parameter LDA, must be at least N, which may be larger than was required by F08KEF (DGEBRD).

To apply Q or P to a real rectangular matrix C, F08KEF (DGEBRD) may be followed by a call to F08KGF (DORMBR).

The complex analogue of this routine is F08KSF (ZGEBRD).

9 Example

This example reduces the matrix A to bidiagonal form, where

$$A = \begin{pmatrix} -0.57 & -1.28 & -0.39 & 0.25 \\ -1.93 & 1.08 & -0.31 & -2.14 \\ 2.30 & 0.24 & 0.40 & -0.35 \\ -1.93 & 0.64 & -0.66 & 0.08 \\ 0.15 & 0.30 & 0.15 & -2.13 \\ -0.02 & 1.03 & -1.43 & 0.50 \end{pmatrix}.$$

9.1 Program Text

```
    Program f08kefe

!     F08KEF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: dgebrd, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: i, info, lda, lwork, m, n
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:,:), d(:), e(:), taup(:),         &
                                          tauq(:), work(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: min
!     .. Executable Statements ..
      Write (nout,*) 'F08KEF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) m, n
      lda = m
      lwork = 64*(m+n)
      Allocate (a(lda,n),d(n),e(n-1),taup(n),tauq(n),work(lwork))

!     Read A from data file

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

!     Reduce A to bidiagonal form

!     The NAG name equivalent of dgebrd is f08kef
      Call dgebrd(m,n,a,lda,d,e,tauq,taup,work,lwork,info)

!     Print bidiagonal form

      Write (nout,*)
      Write (nout,*) 'Diagonal'
      Write (nout,99999) d(1:min(m,n))
      If (m>=n) Then
        Write (nout,*) 'Super-diagonal'
      Else
        Write (nout,*) 'Sub-diagonal'
      End If
      Write (nout,99999) e(1:min(m,n)-1)

99999 Format (1X,8F9.4)
    End Program f08kefe
```

9.2 Program Data

```
F08KEF Example Program Data
  6  4                          :Values of M and N
 -0.57  -1.28  -0.39   0.25
 -1.93   1.08  -0.31  -2.14
  2.30   0.24   0.40  -0.35
 -1.93   0.64  -0.66   0.08
  0.15   0.30   0.15  -2.13
 -0.02   1.03  -1.43   0.50   :End of matrix A
```

9.3 Program Results

```
 F08KEF Example Program Results

 Diagonal
    3.6177   2.4161  -1.9213  -1.4265
 Super-diagonal
    1.2587   1.5262  -1.1895
```