NAG Toolbox for MATLAB

f08ke

1 Purpose

f08ke reduces a real m by n matrix to bidiagonal form.

2 Syntax

```
[a, d, e, tauq, taup, info] = f08ke(a, 'm', m, 'n', n)
```

3 Description

f08ke 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). Functions 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

5.1 Compulsory Input Parameters

1: **a**(**lda**,$*$) – **double array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{m})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The m by n matrix A.

5.2 Optional Input Parameters

1: **m – int32 scalar**

Default: The first dimension of the array **a**.

m, the number of rows of the matrix A.

Constraint: $\mathbf{m} \geq 0$.

2: **n – int32 scalar**

Default: The second dimension of the array **a**.

n, the number of columns of the matrix A.

Constraint: $\mathbf{n} \geq 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, work, lwork

5.4 Output Parameters

1: **a(lda,∗) – double array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{m})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

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.

2: **d(∗) – double array**

Note: the dimension of the array **d** must be at least $\max(1, \min(\mathbf{m}, \mathbf{n}))$.

The diagonal elements of the bidiagonal matrix B.

3: **e(∗) – double array**

Note: the dimension of the array **e** must be at least $\max(1, \min(\mathbf{m}, \mathbf{n}) - 1)$.

The off-diagonal elements of the bidiagonal matrix B.

4: **tauq(∗) – double array**

Note: the dimension of the array **tauq** must be at least $\max(1, \min(\mathbf{m}, \mathbf{n}))$.

Further details of the orthogonal matrix Q.

5: **taup(∗) – double array**

Note: the dimension of the array **taup** must be at least $\max(1, \min(\mathbf{m}, \mathbf{n}))$.

Further details of the orthogonal matrix P.

6: **info – int32 scalar**

info $= 0$ unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows: 1: **m**, 2: **n**, 3: **a**, 4: **lda**, 5: **d**, 6: **e**, 7: **tauq**, 8: **taup**, 9: **work**, 10: **lwork**, 11: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

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 f08ae to perform a QR factorization of A, and then to call f08ke 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 f08ah to perform an LQ factorization of A, and then to call f08ke 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 f08ke may be followed by calls to f08kf:

to form the m by m orthogonal matrix Q

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

```
[a, info] = f08kf('Q', k, a, tauq);
```

but note that the second dimension of the array **a** must be at least **m**, which may be larger than was required by f08ke;

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

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

```
[a, info] = f08kf('P', k, a, taup);
```

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 f08ke.

To apply Q or P to a real rectangular matrix C, f08ke may be followed by a call to f08kg.

The complex analogue of this function is f08ks.

9 Example

```
a = [-0.57, -1.28, -0.39, 0.25;
     -1.93, 1.08, -0.31, -2.14;
     2.3, 0.24, 0.4, -0.35;
     -1.93, 0.64, -0.66, 0.08;
     0.15, 0.3, 0.15, -2.13;
     -0.02, 1.03, -1.43, 0.5];
[aOut, d, e, tauq, taup, info] = f08ke(a)
```

```
aOut =
```

```
    3.6177    1.2587   -0.4668   -0.4110
    0.4609    2.4161    1.5262   -0.2095
   -0.5492    0.1219   -1.9213   -1.1895
    0.4609    0.0770    0.0557   -1.4265
   -0.0358    0.3309   -0.1248   -0.4048
    0.0048    0.2796    0.8322    0.2196
d =
    3.6177
    2.4161
   -1.9213
   -1.4265
e =
    1.2587
    1.5262
   -1.1895
tauq =
    1.1576
    1.6550
    1.1687
    1.6501
taup =
    1.4422
    1.9159
         0
         0
info =
           0
```