NAG Toolbox

nag_lapack_dorgtr (f08ff)

1 Purpose

nag_lapack_dorgtr (f08ff) generates the real orthogonal matrix Q, which was determined by nag_lapack_dsytrd (f08fe) when reducing a symmetric matrix to tridiagonal form.

2 Syntax

```
[a, info] = nag_lapack_dorgtr(uplo, a, tau, 'n', n)
[a, info] = f08ff(uplo, a, tau, 'n', n)
```

3 Description

nag_lapack_dorgtr (f08ff) is intended to be used after a call to nag_lapack_dsytrd (f08fe), which reduces a real symmetric matrix A to symmetric tridiagonal form T by an orthogonal similarity transformation: $A = QTQ^{\mathrm{T}}$. nag_lapack_dsytrd (f08fe) represents the orthogonal matrix Q as a product of $n-1$ elementary reflectors.

This function may be used to generate Q explicitly as a square matrix.

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: **uplo** – CHARACTER(1)

This **must** be the same argument **uplo** as supplied to nag_lapack_dsytrd (f08fe).

Constraint: **uplo** = 'U' or 'L'.

2: **a**$(lda,:)$ – REAL (KIND=nag_wp) array

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

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

Details of the vectors which define the elementary reflectors, as returned by nag_lapack_dsytrd (f08fe).

3: **tau**$(:)$ – REAL (KIND=nag_wp) array

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

Further details of the elementary reflectors, as returned by nag_lapack_dsytrd (f08fe).

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the array **a** and the second dimension of the array **a**.

n, the order of the matrix Q.

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

5.3 Output Parameters

1: $\mathbf{a}(lda, :)$ – REAL (KIND=nag_wp) array

The first dimension of the array **a** will be $\max(1, \mathbf{n})$.

The second dimension of the array **a** will be $\max(1, \mathbf{n})$.

The n by n orthogonal matrix Q.

2: **info** – INTEGER

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

6 Error Indicators and Warnings

$\mathbf{info} = -i$

If $\mathbf{info} = -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **uplo**, 2: **n**, 3: **a**, 4: **lda**, 5: **tau**, 6: **work**, 7: **lwork**, 8: **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 matrix Q differs from an exactly orthogonal matrix by a matrix E such that

$$\|E\|_2 = O(\epsilon),$$

where ϵ is the ***machine precision***.

8 Further Comments

The total number of floating-point operations is approximately $\frac{4}{3}n^3$.

The complex analogue of this function is nag_lapack_zungtr (f08ft).

9 Example

This example computes all the eigenvalues and eigenvectors of the matrix A, where

$$A = \begin{pmatrix} 2.07 & 3.87 & 4.20 & -1.15 \\ 3.87 & -0.21 & 1.87 & 0.63 \\ 4.20 & 1.87 & 1.15 & 2.06 \\ -1.15 & 0.63 & 2.06 & -1.81 \end{pmatrix}.$$

Here A is symmetric and must first be reduced to tridiagonal form by nag_lapack_dsytrd (f08fe). The program then calls nag_lapack_dorgtr (f08ff) to form Q, and passes this matrix to nag_lapack_dsteqr (f08je) which computes the eigenvalues and eigenvectors of A.

9.1 Program Text

```
function f08ff_example

fprintf('f08ff example results\n\n');

uplo = 'L';
a = [ 2.07,  0,    0,     0;
      3.87, -0.21, 0,     0;
      4.20,  1.87, 1.15,  0;
     -1.15,  0.63, 2.06, -1.81];

% Reduce to tridiagonal form
[t, d, e, tau, info] = f08fe( ...
       uplo, a);

% Form Q
[q, info] = f08ff( ...
    uplo, t, tau);

% Compute eigenvalues and eigenvectors
[w, e, z, info] = f08je( ...
  'V', d, e, 'z', q);

disp('Eigenvalues');
disp(w');
disp('Eigenvectors');
disp(z);
```

9.2 Program Results

```
f08ff example results

Eigenvalues
   -5.0034   -1.9987    0.2013    8.0008

Eigenvectors
   -0.5658   -0.2328    0.3965   -0.6845
    0.3478    0.7994    0.1780   -0.4564
    0.4740   -0.4087   -0.5381   -0.5645
   -0.5781    0.3737   -0.7221   -0.0676
```