NAG Library Routine Document

F06QVF

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

F06QVF transforms a real upper triangular matrix to an upper Hessenberg matrix by applying a given sequence of plane rotations.

2 Specification

```
SUBROUTINE F06QVF (SIDE, N, K1, K2, C, S, A, LDA)

INTEGER              N, K1, K2, LDA
REAL (KIND=nag_wp) C(*), S(*), A(LDA,*)
CHARACTER(1)         SIDE
```

3 Description

F06QVF transforms an n by n real upper triangular matrix U to an upper Hessenberg matrix H, by applying a given sequence of plane rotations from either the left or the right, in planes k_1 to k_2; H has nonzero subdiagonal elements $h_{k+1,k}$, for $k = k_1, \ldots, k_2 - 1$ only.

If SIDE = 'L', the rotations are applied from the left:

$$H = PU,$$

where $P = P_{k_1} P_{k_1+1} \cdots P_{k_2-1}$.

If SIDE = 'R', the rotations are applied from the right:

$$H = UP^{\mathrm{T}},$$

where $P = P_{k_2-1} \cdots P_{k_1+1} P_{k_1}$.

In either case, P_k is a rotation in the $(k, k+1)$ plane.

The 2 by 2 plane rotation part of P_k has the form

$$\begin{pmatrix} c_k & s_k \\ -s_k & c_k \end{pmatrix}.$$

4 References

None.

5 Arguments

1: SIDE – CHARACTER(1) *Input*

On entry: specifies whether U is operated on from the left or the right.

SIDE = 'L'
U is pre-multiplied from the left.

SIDE = 'R'
U is post-multiplied from the right.

Constraint: SIDE = 'L' or 'R'.

2: N – INTEGER *Input*

On entry: n, the order of the matrices U and H.

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

3: K1 – INTEGER *Input*
4: K2 – INTEGER *Input*

On entry: the values k_1 and k_2.

If $\mathrm{K1} < 1$ or $\mathrm{K2} \leq \mathrm{K1}$ or $\mathrm{K2} > \mathrm{N}$, an immediate return is effected.

5: $\mathrm{C}(*)$ – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array C must be at least $\mathrm{K2} - \mathrm{K1}$.

On entry: $\mathrm{C}(k)$ must hold c_k, the cosine of the rotation P_k, for $k = k_1, \ldots, k_2 - 1$.

6: $\mathrm{S}(*)$ – REAL (KIND=nag_wp) array *Input/Output*

Note: the dimension of the array S must be at least $\mathrm{K2} - \mathrm{K1}$.

On entry: $\mathrm{S}(k)$ must hold s_k, the sine of the rotation P_k, for $k = k_1, \ldots, k_2 - 1$.

On exit: $\mathrm{S}(k)$ holds $h_{k+1,k}$, the subdiagonal element of H, for $k = k_1, \ldots, k_2 - 1$.

7: $\mathrm{A}(\mathrm{LDA}, *)$ – REAL (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array A must be at least N.

On entry: the n by n upper triangular matrix U.

On exit: the upper triangular part of the upper Hessenberg matrix H.

8: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F06QVF is called.

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

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Parallelism and Performance

F06QVF is not threaded in any implementation.

9 Further Comments

None.

10 Example

None.