NAG Library Routine Document

F06TVF

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

F06TVF transforms a complex upper triangular matrix to an upper Hessenberg matrix by applying a given sequence of plane rotations.

2 Specification

```
SUBROUTINE F06TVF (SIDE, N, K1, K2, C, S, A, LDA)

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

3 Description

F06TVF transforms an n by n complex upper triangular matrix U with real diagonal elements, 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 real 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:

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

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

$$\begin{pmatrix} \bar{c}_k & s_k \\ -s_k & c_k \end{pmatrix}$$

with s_k real.

4 References

None.

5 Parameters

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: $N \geq 0$.

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

On entry: the values k_1 and k_2.

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

5: C(*) – COMPLEX (KIND=nag_wp) array *Input*

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

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

6: S(*) – REAL (KIND=nag_wp) array *Input/Output*

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

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

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

7: A(LDA, *) – COMPLEX (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. The imaginary parts of the diagonal elements must be zero.

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 F06TVF is called.

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

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

None.