

NAG Library Routine Document

F08QTF (ZTREXC)

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

F08QTF (ZTREXC) reorders the Schur factorization of a complex general matrix.

2 Specification

```
SUBROUTINE F08QTF( COMPQ, N, T, LDT, Q, LDQ, IFST, ILST, INFO)
INTEGER          N, LDT, LDQ, IFST, ILST, INFO
complex*16      T(LDT,*), Q(LDQ,*)
CHARACTER*1      COMPQ
```

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

3 Description

F08QTF (ZTREXC) reorders the Schur factorization of a complex general matrix $A = QTQ^H$, so that the diagonal element of T with row index IFST is moved to row ILST.

The reordered Schur form $\tilde{T}$ is computed by a unitary similarity transformation: $\tilde{T} = Z^H T Z$. Optionally the updated matrix $\tilde{Q}$ of Schur vectors is computed as $\tilde{Q} = QZ$, giving $A = \tilde{Q}\tilde{T}\tilde{Q}^H$.

4 References

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

5 Parameters

- 1: COMPQ – CHARACTER*1 *Input*
On entry: indicates whether the matrix Q of Schur vectors is to be updated.
 COMPQ = 'V'
 The matrix Q of Schur vectors is updated.
 COMPQ = 'N'
 No Schur vectors are updated.
Constraint: COMPQ = 'V' or 'N'.
- 2: N – INTEGER *Input*
On entry: n , the order of the matrix T .
Constraint: $N \geq 0$.
- 3: T(LDT,*) – **complex*16** array *Input/Output*
Note: the second dimension of the array T must be at least $\max(1, N)$.
On entry: the n by n upper triangular matrix T , as returned by F08PSF (ZHSEQR).
On exit: T is overwritten by the updated matrix $\tilde{T}$.

- 4: LDT – INTEGER *Input*
On entry: the first dimension of the array T as declared in the (sub)program from which F08QTF (ZTREXC) is called.
Constraint: $LDT \geq \max(1, N)$.
- 5: Q(LDQ,*) – **complex*16** array *Input/Output*
Note: the second dimension of the array Q must be at least $\max(1, N)$ if COMPQ = 'V' and at least 1 if COMPQ = 'N'.
On entry: if COMPQ = 'V', Q must contain the n by n unitary matrix Q of Schur vectors.
On exit: if COMPQ = 'V', Q contains the updated matrix of Schur vectors.
 If COMPQ = 'N', Q is not referenced.
- 6: LDQ – INTEGER *Input*
On entry: the first dimension of the array Q as declared in the (sub)program from which F08QTF (ZTREXC) is called.
Constraints:
 if COMPQ = 'V', $LDQ \geq \max(1, N)$;
 if COMPQ = 'N', $LDQ \geq 1$.
- 7: IFST – INTEGER *Input*
 8: ILST – INTEGER *Input*
On entry: IFST and ILST must specify the reordering of the diagonal elements of T . The element with row index IFST is moved to row ILST by a sequence of exchanges between adjacent elements.
Constraint: $1 \leq IFST \leq N$ and $1 \leq ILST \leq N$.
- 9: INFO – INTEGER *Output*
On exit: INFO = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

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 matrix $\tilde{T}$ is exactly similar to a matrix $(T + E)$, where

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

and ϵ is the **machine precision**.

The values of the eigenvalues are never changed by the reordering.

8 Further Comments

The total number of real floating-point operations is approximately $20nr$ if COMPQ = 'N', and $40nr$ if COMPQ = 'V', where $r = |IFST - ILST|$.

The real analogue of this routine is F08QFF (DTREXC).

9 Example

This example reorders the Schur factorization of the matrix T so that element t_{11} is moved to t_{44} , where

$$T = \begin{pmatrix} -6.00 - 7.00i & 0.36 - 0.36i & -0.19 + 0.48i & 0.88 - 0.25i \\ 0.00 + 0.00i & -5.00 + 2.00i & -0.03 - 0.72i & -0.23 + 0.13i \\ 0.00 + 0.00i & 0.00 + 0.00i & 8.00 - 1.00i & 0.94 + 0.53i \\ 0.00 + 0.00i & 0.00 + 0.00i & 0.00 + 0.00i & 3.00 - 4.00i \end{pmatrix}.$$

9.1 Program Text

```
*      F08QTF Example Program Text
*      Mark 21 Release. NAG Copyright 2004.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, LDQ
      PARAMETER        (NMAX=8,LDQ=1)
      INTEGER          LDT
      PARAMETER        (LDT=NMAX)
*      .. Local Scalars ..
      INTEGER          I, IFAIL, IFST, ILST, INFO, J, N
*      .. Local Arrays ..
      COMPLEX *16      Q(LDQ,1), T(LDT,NMAX)
      CHARACTER        CLABS(1), RLABS(1)
*      .. External Subroutines ..
      EXTERNAL         X04DBF, ZTREXC
*      .. Executable Statements ..
      WRITE (NOUT,*) 'F08QTF Example Program Results'
      WRITE (NOUT,*)
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*          Read T from data file
*
*          READ (NIN,*) ((T(I,J),J=1,N),I=1,N)
*
*          Read the row indices
*
*          READ (NIN,*) IFST, ILST
*
*          Reorder the Schur factor T
*
*          CALL ZTREXC('No update',N,T,LDT,Q,LDQ,IFST,ILST,INFO)
*          IF (INFO.NE.0) THEN
*              WRITE (NOUT,99999) INFO, ILST
*              WRITE (NOUT,*)
*          END IF
*
*          Print reordered Schur form
*
*          IFAIL = 0
*          CALL X04DBF('General',' ',N,N,T,LDT,'Bracketed','F7.4',
+              'Reordered Schur form','Integer',RLABS,'Integer',
+              CLABS,80,0,IFAIL)
*
*          ELSE
*              WRITE (NOUT,*) 'NMAX too small'
*          END IF
*
*          99999 FORMAT (' Reordering could not be completed. INFO = ',I3,' ILST ',
+              '= ',I5)
*          END
```

9.2 Program Data

F08QTF Example Program Data

```

4                                     :Value of N
(-6.00,-7.00) ( 0.36,-0.36) (-0.19, 0.48) ( 0.88,-0.25)
( 0.00, 0.00) (-5.00, 2.00) (-0.03,-0.72) (-0.23, 0.13)
( 0.00, 0.00) ( 0.00, 0.00) ( 8.00,-1.00) ( 0.94, 0.53)
( 0.00, 0.00) ( 0.00, 0.00) ( 0.00, 0.00) ( 3.00,-4.00) :End of matrix T
1 4                                     :Values of IFST and ILST

```

9.3 Program Results

F08QTF Example Program Results

Reordered Schur form

```

                                     1           2           3           4
1  (-5.0000, 2.0000) (-0.1574, 0.7143) ( 0.1781,-0.1913) ( 0.3950, 0.3861)
2  ( 0.0000, 0.0000) ( 8.0000,-1.0000) ( 1.0742, 0.1447) ( 0.2515,-0.3397)
3  ( 0.0000, 0.0000) ( 0.0000, 0.0000) ( 3.0000,-4.0000) ( 0.2264, 0.8962)
4  ( 0.0000, 0.0000) ( 0.0000, 0.0000) ( 0.0000, 0.0000) (-6.0000,-7.0000)

```
