NAG Library Routine Document

F07WRF (ZPFTRF)

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

F07WRF (ZPFTRF) computes the Cholesky factorization of a complex Hermitian positive definite matrix stored in Rectangular Full Packed (RFP) format. The RFP storage format is described in Section 3.3.3 in the F07 Chapter Introduction.

2 Specification

```
SUBROUTINE F07WRF (TRANSR, UPLO, N, A, INFO)

INTEGER                 N, INFO
COMPLEX (KIND=nag_wp) A(N*(N+1)/2)
CHARACTER(1)            TRANSR, UPLO
```

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

3 Description

F07WRF (ZPFTRF) forms the Cholesky factorization of a complex Hermitian positive definite matrix A either as $A = U^{\mathrm{H}}U$ if UPLO = 'U' or $A = LL^{\mathrm{H}}$ if UPLO = 'L', where U is an upper triangular matrix and L is an lower triangular, stored using RFP format.

4 References

Demmel J W (1989) On floating-point errors in Cholesky *LAPACK Working Note No. 14* University of Tennessee, Knoxville

Gustavson F G, Waśniewski J, Dongarra J J and Langou J (2010) Rectangular full packed format for Cholesky's algorithm: factorization, solution, and inversion *ACM Trans. Math. Software* **37, 2**

5 Parameters

1: TRANSR – CHARACTER(1) *Input*

On entry: specifies whether the normal RFP representation of A or its conjugate transpose is stored.

TRANSR = 'N'
The matrix A is stored in normal RFP format.

TRANSR = 'C'
The conjugate transpose of the RFP representation of the matrix A is stored.

Constraint: TRANSR = 'N' or 'C'.

2: UPLO – CHARACTER(1) *Input*

On entry: specifies whether the upper or lower triangular part of A is stored.

UPLO = 'U'
The upper triangular part of A is stored, and A is factorized as $U^{\mathrm{H}}U$, where U is upper triangular.

UPLO = 'L'

The lower triangular part of A is stored, and A is factorized as LL^{H}, where L is lower triangular.

Constraint: UPLO = 'U' or 'L'.

3: N – INTEGER *Input*

On entry: n, the order of the matrix A.

Constraint: N ≥ 0.

4: A(N $\times$ (N + 1)/2) – COMPLEX (KIND=nag_wp) array *Input/Output*

On entry: the n by n Hermitian matrix A, stored in RFP format, as described in Section 3.3.3 in the F07 Chapter Introduction.

On exit: if INFO = 0, the factor U or L from the Cholesky factorization $A = U^{\mathrm{H}}U$ or $A = LL^{\mathrm{H}}$, in the same storage format as A.

5: INFO – INTEGER *Output*

On exit: INFO = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If INFO $= -i$, the ith parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0

If INFO $= i$, the leading minor of order i is not positive definite and the factorization could not be completed. Hence A itself is not positive definite. This may indicate an error in forming the matrix A.

7 Accuracy

If UPLO = 'U', the computed factor U is the exact factor of a perturbed matrix $A + E$, where

$$|E| \leq c(n)\epsilon \left|U^{\mathrm{H}}\right||U|,$$

$c(n)$ is a modest linear function of n, and ϵ is the ***machine precision***.

If UPLO = 'L', a similar statement holds for the computed factor L. It follows that $\left|e_{ij}\right| \leq c(n)\epsilon\sqrt{a_{ii}a_{jj}}$.

8 Further Comments

The total number of real floating point operations is approximately $\frac{4}{3}n^2$.

A call to F07WRF (ZPFTRF) may be followed by calls to the routines:

F07WSF (ZPFTRS) to solve $AX = B$;

F07WWF (ZPFTRI) to compute the inverse of A.

The real analogue of this routine is F07WDF (DPFTRF).

9 Example

This example computes the Cholesky factorization of the matrix A, where

$$A = \begin{pmatrix} 3.23 + 0.00i & 1.51 - 1.92i & 1.90 + 0.84i & 0.42 + 2.50i \\ 1.51 + 1.92i & 3.58 + 0.00i & -0.23 + 1.11i & -1.18 + 1.37i \\ 1.90 - 0.84i & -0.23 - 1.11i & 4.09 + 0.00i & 2.33 - 0.14i \\ 0.42 - 2.50i & -1.18 - 1.37i & 2.33 + 0.14i & 4.29 + 0.00i \end{pmatrix}.$$

and is stored using RFP format.

9.1 Program Text

```
    Program f07wrfe

!     F07WRF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, x04dbf, zpftrf, ztfttr
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: ifail, info, ldf, lena, n
      Character (1)                    :: transr, uplo
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: a(:), f(:,:)
      Character (1)                    :: clabs(1), rlabs(1)
!     .. Executable Statements ..
      Write (nout,*) 'F07WRF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, uplo, transr

      lena = n*(n+1)/2
      ldf = n
      Allocate (a(lena),f(ldf,n))

!     Read A from data file
      Read (nin,*) a(1:lena)

!     Factorize A
!     The NAG name equivalent of zpftrf is f07wrf
      Call zpftrf(transr,uplo,n,a,info)

      Write (nout,*)
      Flush (nout)
      If (info==0) Then

!       Convert and print factor
!       The NAG name equivalent of ztfttr is f01vhf
        Call ztfttr(transr,uplo,n,a,f,ldf,info)
        ifail = 0
        Call x04dbf(uplo,'Nonunit',n,n,f,ldf,'Bracketed','F7.4','Factor', &
          'Integer',rlabs,'Integer',clabs,80,0,ifail)

      Else
        Write (nout,*) 'A is not positive definite'
      End If

    End Program f07wrfe
```

9.2 Program Data

```
F07WRF Example Program Data
  4 'L' 'N'           : n, uplo, transr
 ( 4.09, 0.00)
 ( 3.23, 0.00)
 ( 1.51, 1.92)
 ( 1.90,-0.84)
 ( 0.42,-2.50)

 ( 2.33, 0.14)
 ( 4.29, 0.00)
 ( 3.58, 0.00)
 (-0.23,-1.11)
 (-1.18,-1.37)        : A in RFP storage
```

9.3 Program Results

```
 F07WRF Example Program Results

 Factor
                     1                 2                 3                 4
 1  ( 1.7972, 0.0000)
 2  ( 0.8402, 1.0683) ( 1.3164, 0.0000)
 3  ( 1.0572,-0.4674) (-0.4702, 0.3131) ( 1.5604,-0.0000)
 4  ( 0.2337,-1.3910) ( 0.0834, 0.0368) ( 0.9360, 0.8105) ( 0.8713,-0.0000)
```