NAG Library Routine Document

F07WWF (ZPFTRI)

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

F07WWF (ZPFTRI) computes the inverse of a complex Hermitian positive definite matrix using the Cholesky factorization computed by F07WRF (ZPFTRF) and 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 F07WWF (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 ***zpftri***.

3 Description

F07WWF (ZPFTRI) is used to compute the inverse of a complex Hermitian positive definite matrix A, the routine must be preceded by a call to F07WRF (ZPFTRF), which computes the Cholesky factorization of A.

If UPLO = 'U', $A = U^{\mathrm{H}}U$ and A^{-1} is computed by first inverting U and then forming $\left(U^{-1}\right)U^{-\mathrm{H}}$.

If UPLO = 'L', $A = LL^{\mathrm{H}}$ and A^{-1} is computed by first inverting L and then forming $L^{-\mathrm{H}}\left(L^{-1}\right)$.

4 References

Du Croz J J and Higham N J (1992) Stability of methods for matrix inversion *IMA J. Numer. Anal.* **12** 1–19

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 how A has been factorized.

UPLO = 'U'
    $A = U^{\mathrm{H}}U$, where U is upper triangular.

UPLO = 'L'
$A = LL^{\mathrm{H}}$, where L is lower triangular.

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

3: N – INTEGER *Input*

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

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

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

On entry: the Cholesky factorization of A stored in RFP format, as returned by F07WRF (ZPFTRF).

On exit: the factorization is overwritten by the n by n matrix A^{-1} stored in RFP format.

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 ith diagonal element of the Cholesky factor is zero; the Cholesky factor is singular and the inverse of A cannot be computed.

7 Accuracy

The computed inverse X satisfies

$$\|XA - I\|_2 \leq c(n)\epsilon\kappa_2(A) \qquad \text{and} \qquad \|AX - I\|_2 \leq c(n)\epsilon\kappa_2(A),$$

where $c(n)$ is a modest function of n, ϵ is the ***machine precision*** and $\kappa_2(A)$ is the condition number of A defined by

$$\kappa_2(A) = \|A\|_2 \left\|A^{-1}\right\|_2.$$

8 Further Comments

The total number of real floating point operations is approximately $\frac{8}{3}n^3$.

The real analogue of this routine is F07WJF (DPFTRI).

9 Example

This example computes the inverse of the matrix A, where

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

Here A is Hermitian positive definite, stored in RFP format, and must first be factorized by F07WRF (ZPFTRF).

9.1 Program Text

```
    Program f07wwfe

!     F07WWF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, x04dbf, zpftrf, zpftri, 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,*) 'F07WWF 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

!       Compute inverse of A
!       The NAG name equivalent of zpftri is f07wwf
        Call zpftri(transr,uplo,n,a,info)

!       Convert and print inverse
!       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','Inverse', &
          'Integer',rlabs,'Integer',clabs,80,0,ifail)

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

    End Program f07wwfe
```

9.2 Program Data

```
F07WWF 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

```
F07WWF Example Program Results

Inverse
                  1                  2                  3                  4
1  ( 5.4691, 0.0000)
2  (-1.2624,-1.5491) ( 1.1024, 0.0000)
3  (-2.9746,-0.9616) ( 0.8989,-0.5672) ( 2.1589,-0.0000)
4  ( 1.1962, 2.9772) (-0.9826,-0.2566) (-1.3756,-1.4550) ( 2.2934, 0.0000)
```