

NAG Library Routine Document

F06WNF (ZLANHF)

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

F06WNF (ZLANHF) returns the value of the 1-norm, the ∞ -norm, the Frobenius norm, or the maximum absolute value of the elements of a complex Hermitian matrix A stored in Rectangular Full Packed (RFP) format.

2 Specification

```
FUNCTION F06WNF (NORM, TRANSR, UPLO, N, AR, WORK)
REAL (KIND=nag_wp) F06WNF
INTEGER N
REAL (KIND=nag_wp) WORK(*)
COMPLEX (KIND=nag_wp) AR(N*(N+1)/2)
CHARACTER(1) NORM, TRANSR, UPLO
```

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

3 Description

Given a complex n by n symmetric matrix, A , F06WNF (ZLANHF) calculates one of the values given by

$$\|A\|_1 = \max_j \sum_{i=1}^n |a_{ij}| \quad (\text{the 1-norm of } A),$$

$$\|A\|_\infty = \max_i \sum_{j=1}^n |a_{ij}| \quad (\text{the } \infty\text{-norm of } A),$$

$$\|A\|_F = \left(\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2} \quad (\text{the Frobenius norm of } A), \quad \text{or}$$

$$\max_{i,j} |a_{ij}| \quad (\text{the maximum absolute element value of } A).$$

A is stored in compact form using the RFP format. The RFP storage format is described in Section 3.3.3 in the F07 Chapter Introduction.

4 References

Basic Linear Algebra Subprograms Technical (BLAST) Forum (2001) *Basic Linear Algebra Subprograms Technical (BLAST) Forum Standard* University of Tennessee, Knoxville, Tennessee <http://www.netlib.org/blas/blast-forum/blas-report.pdf>

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: NORM – CHARACTER(1) *Input*
- On entry:* specifies the value to be returned.
- NORM = '1' or 'O'
The 1-norm.
- NORM = 'I'
The ∞ -norm.
- NORM = 'F' or 'E'
The Frobenius (or Euclidean) norm.
- NORM = 'M'
The value $\max_{i,j} |a_{ij}|$ (not a norm).
- Constraint:* NORM = '1', 'O', 'I', 'F', 'E' or 'M'.
- 2: 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'.
- 3: 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.
- UPLO = 'L'
The lower triangular part of A is stored.
- Constraint:* UPLO = 'U' or 'L'.
- 4: N – INTEGER *Input*
- On entry:* n , the order of the matrix A .
- When N = 0, F06WNF (ZLANHF) returns zero.
- Constraint:* $N \geq 0$.
- 5: AR($N \times (N + 1)/2$) – COMPLEX (KIND=nag_wp) array *Input*
- On entry:* the upper or lower triangular part (as specified by UPLO) of the n by n Hermitian matrix A , in either normal or transposed RFP format (as specified by TRANSR). The storage format is described in detail in Section 3.3.3 in the F07 Chapter Introduction.
- 6: WORK(*) – REAL (KIND=nag_wp) array *Workspace*
- Note:** the dimension of the array WORK must be at least $\max(1, N)$ if NORM = '1', 'O' or 'I', and at least 1 otherwise.

6 Error Indicators and Warnings

None.

7 Accuracy

The BLAS standard requires accurate implementations which avoid unnecessary over/underflow (see Section 2.7 of Basic Linear Algebra Subprograms Technical (BLAST) Forum (2001)).

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example reads in the lower triangular part of a symmetric matrix, converts this to RFP format, then calculates the norm of the matrix for each of the available norm types.

10.1 Program Text

```

Program f06wnfe

!      F06WNF Example Program Text
!
!      Mark 25 Release. NAG Copyright 2014.
!
!      .. Use Statements ..
Use nag_library, Only: nag_wp, zlanhf, ztrttf
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Real (Kind=nag_wp)         :: r_fro, r_inf, r_max, r_one
Integer                    :: i, info, lda, n
Character (1)              :: transr, uplo
!      .. Local Arrays ..
Complex (Kind=nag_wp), Allocatable :: a(:,,:), ar(:)
Real (Kind=nag_wp), Allocatable  :: work(:)
!      .. Executable Statements ..
Write (nout,*) 'F06WNF Example Program Results'

!      Skip heading in data file
Read (nin,*)

Read (nin,*) n, uplo, transr

lda = n
Allocate (a(lda,n),ar((n*(n+1))/2),work(n))

!      Read upper or lower triangle of matrix A from data file
If (uplo=='L' .Or. uplo=='l') Then
  Do i = 1, n
    Read (nin,*) a(i,1:i)
  End Do
Else
  Do i = 1, n
    Read (nin,*) a(i,i:n)
  End Do
End If

!      Convert A to rectangular full packed storage in ar
!
!      The NAG name equivalent of ztrttf is f01vff
Call ztrttf(transr,uplo,n,a,lda,ar,info)

```

```

Write (nout,*)
Write (nout,99999) &
  'Norms of Hermitian matrix stored in RFP format in ar:'
Write (nout,*)

! The NAG name equivalent of zlanhf is f06wnf
r_one = zlanhf('1-norm',transr,uplo,n,ar,work)
Write (nout,99998) 'One norm          = ', r_one

r_inf = zlanhf('Infinity',transr,uplo,n,ar,work)
Write (nout,99998) 'Infinity norm     = ', r_inf

r_fro = zlanhf('Frobenius',transr,uplo,n,ar,work)
Write (nout,99998) 'Frobenius norm    = ', r_fro

r_max = zlanhf('Max norm',transr,uplo,n,ar,work)
Write (nout,99998) 'Maximum norm     = ', r_max

99999 Format (1X,A)
99998 Format (1X,A,F9.4)
End Program f06wnfe

```

10.2 Program Data

F06WNF Example Program Data

```

6 'L' 'N'      : N, UPLO, TRANSR
(1.0,1.1)
(2.0,2.1) (2.0,2.1)
(3.0,3.3) (3.3,3.0) (3.2,3.0)
(4.0,4.4) (4.0,4.3) (4.0,4.2) (4.0,4.1)
(5.0,5.1) (5.0,5.2) (5.3,5.0) (5.0,5.4) (5.5,5.0)
(6.9,6.0) (6.0,6.8) (6.7,6.0) (6.0,6.6) (6.5,6.0) (6.0,6.4) : Matrix A

```

10.3 Program Results

F06WNF Example Program Results

Norms of Hermitian matrix stored in RFP format in ar:

```

One norm          = 50.9719
Infinity norm     = 50.9719
Frobenius norm    = 40.3801
Maximum norm     =  9.1439

```
