

NAG Library Routine Document

F01VBF (ZTRTTP)

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

F01VBF (ZTRTTP) copies a complex triangular matrix stored in full format in a two-dimensional array to packed format in a one-dimensional array. Packed storage format is described in Section 3.3.2 in the F07 Chapter Introduction.

2 Specification

```
SUBROUTINE F01VBF (UPLO, N, A, LDA, AP, INFO)
INTEGER          N, LDA, INFO
COMPLEX (KIND=nag_wp) A(LDA,*), AP(N*(N+1)/2)
CHARACTER(1)    UPLO
```

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

3 Description

F01VBF (ZTRTTP) packs a complex n by n triangular matrix A , stored conventionally in a two-dimensional array into a one-dimensional array of length $n(n+1)/2$. The matrix is packed by columns. This routine is intended for possible use in conjunction with routines from Chapters F06, F07 and F08 where some routines use triangular matrices stored in the packed form.

4 References

None.

5 Parameters

- | | | |
|----|--|--------------|
| 1: | UPLO – CHARACTER(1)
<i>On entry:</i> specifies whether A is upper or lower triangular.
UPLO = 'U'
A is upper triangular.
UPLO = 'L'
A is lower triangular.
<i>Constraint:</i> UPLO = 'U' or 'L'. | <i>Input</i> |
| 2: | N – INTEGER
<i>On entry:</i> n , the order of the matrix A .
<i>Constraint:</i> $N \geq 1$. | <i>Input</i> |
| 3: | A(LDA,*) – COMPLEX (KIND=nag_wp) array
Note: the second dimension of the array A must be at least N .
<i>On entry:</i> the triangular matrix A . | <i>Input</i> |

If UPLO = 'U', A is upper triangular and the elements of the array below the diagonal are not referenced.

If UPLO = 'L', A is lower triangular and the elements of the array above the diagonal are not referenced.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F01VBF (ZTRTTP) is called.

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

5: AP($N \times (N + 1)/2$) – COMPLEX (KIND=nag_wp) array *Output*

On exit: the n by n triangular matrix A , packed by columns.

More precisely,

if UPLO = 'U', the upper triangle of A is stored with element A_{ij} in $AP(i + j(j - 1)/2)$ for $i \leq j$;

if UPLO = 'L', the lower triangle of A is stored with element A_{ij} in $AP(i + (2n - j)(j - 1)/2)$ for $i \geq j$.

6: 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 = $-i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example reads in a triangular matrix and copies it to packed format.

9.1 Program Text

```

Program f01vbfe

!   F01VBF Example Program Text
!
!   Mark 24 Release. NAG Copyright 2012.
!
!   .. Use Statements ..
Use nag_library, Only: nag_wp, x04dbf, ztrttp
!   .. Implicit None Statement ..
Implicit None
!   .. Parameters ..
Integer, Parameter      :: incl = 1, indent = 0, ncols = 80,      &

```

```

                                nin = 5, nout = 6
Character (1), Parameter        :: brac = 'B', diag = 'N', intlabel =  &
                                'I', matrix = 'G', nolabel = 'N'
Character (4), Parameter        :: form = 'F5.2'
! .. Local Scalars ..
Integer                          :: i, ifail, info, lda, lenap, n
Character (18)                   :: title
Character (1)                    :: uplo
! .. Local Arrays ..
Complex (Kind=nag_wp), Allocatable :: a(:,,:), ap(:)
Character (1)                    :: clabs(1), rlabs(1)
! .. Executable Statements ..
Write (nout,*) 'F01VBF Example Program Results'
! Skip heading in data file
Read (nin,*)
Write (nout,*)
Flush (nout)
Read (nin,*) n, uplo
lda = n
lenap = (n*(n+1))/2
Allocate (a(lda,n),ap(lenap))

! Read a triangular matrix of order n
Do i = 1, n
  Read (nin,*) a(i,1:n)
End Do

! Print the unpacked matrix
title = 'Unpacked Matrix A:'
ifail = 0
Call x04dbf(uplo,diag,n,n,a,lda,brac,form,title,intlabel,rlabs,intlabel, &
  clabs,ncols,indent,ifail)

Write (nout,*)
Flush (nout)

! Convert to packed vector form
info = 0
! The NAG name equivalent of ztrttp is f01vbf
Call ztrttp(uplo,n,a,lda,ap,info)

! Print the packed vector
title = 'Packed Matrix AP: '
ifail = 0
Call x04dbf(matrix,diag,lenap,incl,ap,lenap,brac,form,title,intlabel, &
  rlabs,nolabel,clabs,ncols,indent,ifail)

End Program f01vbfe

```

9.2 Program Data

F01VBF Example Program Data

```

4 'U'                                : n, uplo
(1.1,1.1) (1.2,1.2) (1.3,1.3) (1.4,1.4) : Unpacked Matrix A
(0.0,0.0) (2.2,2.2) (2.3,2.3) (2.4,2.4)
(0.0,0.0) (0.0,0.0) (3.3,3.3) (3.4,3.4)
(0.0,0.0) (0.0,0.0) (0.0,0.0) (4.4,4.4)

```

9.3 Program Results

F01VBF Example Program Results

Unpacked Matrix A:

```

      1          2          3          4
1 ( 1.10, 1.10) ( 1.20, 1.20) ( 1.30, 1.30) ( 1.40, 1.40)
2           ( 2.20, 2.20) ( 2.30, 2.30) ( 2.40, 2.40)
3                    ( 3.30, 3.30) ( 3.40, 3.40)
4                               ( 4.40, 4.40)

```

Packed Matrix AP:

1 (1.10, 1.10)
2 (1.20, 1.20)
3 (2.20, 2.20)
4 (1.30, 1.30)
5 (2.30, 2.30)
6 (3.30, 3.30)
7 (1.40, 1.40)
8 (2.40, 2.40)
9 (3.40, 3.40)
10 (4.40, 4.40)
