

NAG Toolbox for MATLAB

f01zc

1 Purpose

f01zc copies a real band matrix stored in a packed array into an unpacked array, or vice versa.

2 Syntax

```
[a, b, ifail] = f01zc(job, m, kl, ku, a, b, 'n', n)
```

3 Description

f01zc unpacks a band matrix that is stored in a packed array, or packs a band matrix that is stored in an unpacked array. The band matrix has m rows, n columns, k_l nonzero subdiagonals, and k_u nonzero superdiagonals. This function is intended for possible use in conjunction with functions from Chapters F06, F07 and F08, where functions that use band matrices store them in the packed form described below.

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **job** – string

Specifies whether the band matrix is to be packed or unpacked.

job = 'P' (Pack)

The band matrix is to be packed into array **b**.

job = 'U' (Unpack)

The band matrix is to be unpacked into array **a**.

Constraint: **job** must be one of 'P' or 'U'.

2: **m** – int32 scalar

m and n , the number of rows and columns of the band matrix, respectively.

Constraint: **m**, **n** > 0.

3: **kl** – int32 scalar

k_l , the number of subdiagonals of the band matrix.

Constraint: **kl** ≥ 0.

4: **ku** – int32 scalar

k_u , the number of superdiagonals of the band matrix.

Constraint: **ku** ≥ 0.

5: **a(lda,n)** – double array

lda, the first dimension of the array, must be at least **m**.

If **job** = 'P', then the leading m by n part of **a** must contain the band matrix stored in unpacked form. Elements of the array that lie outside the banded part of the matrix are not referenced and need not be assigned.

6: **b(ldb,*) – double array**

The first dimension of the array **b** must be at least $(kl + ku + 1)$

The second dimension of the array must be at least $\min(m + ku, n)$

If **job** = 'U', then **b** must contain the band matrix in packed form, in the leading $(kl + ku + 1)$ by $\min(m + ku, n)$ part of the array. The matrix is packed column by column, with the leading diagonal of the matrix in row $(ku + 1)$ of **b**, the first superdiagonal starting at position 2 in row ku , the first subdiagonal starting at position 1 in row $(ku + 2)$, and so on. Elements of **b** that are not needed to store the band matrix, for instance the leading ku by ku triangle, are not referenced and need not be assigned.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the array **a**.

m and n , the number of rows and columns of the band matrix, respectively.

Constraint: $m, n > 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb

5.4 Output Parameters

1: **a(lda,n) – double array**

If **job** = 'U', then the leading m by n part of **a** contains the band matrix stored in unpacked form. Elements of the leading m by n part of **a** that are not within the banded part of the matrix are assigned the value zero.

2: **b(ldb,*) – double array**

The first dimension of the array **b** must be at least $(kl + ku + 1)$

The second dimension of the array must be at least $\min(m + ku, n)$

If **job** = 'P', then **b** contains the band matrix stored in packed form. Elements of **b** that are not needed to store the band matrix are not referenced.

3: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

On entry, **job** \neq 'P' or 'U'.

ifail = 2

On entry, $kl < 0$.

ifail = 3

On entry, **ku** < 0.

ifail = 4

On entry, **lda** < **m**.

ifail = 5

On entry, **ldb** < **kl** + **ku** + 1.

ifail = 6

On entry, **m** < 1,
or **n** < 1.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

```
job = 'Pack';
m = int32(5);
kl = int32(1);
ku = int32(1);
a = [1.1, 1.2, 0, 0, 0;
     2.1, 2.2, 2.3, 0, 0;
     0, 3.2, 3.3, 3.4, 0;
     0, 0, 4.3, 4.4, 4.5;
     0, 0, 0, 5.4, 5.5];
b = [0, 0, 0, 0, 0;
     0, 0, 0, 0, 0;
     0, 0, 0, 0, 0];
[aOut, bOut, ifail] = f01zc(job, m, kl, ku, a, b)
```

```
aOut =
    1.1000    1.2000         0         0         0
    2.1000    2.2000    2.3000         0         0
         0    3.2000    3.3000    3.4000         0
         0         0    4.3000    4.4000    4.5000
         0         0         0    5.4000    5.5000

bOut =
         0    1.2000    2.3000    3.4000    4.5000
    1.1000    2.2000    3.3000    4.4000    5.5000
    2.1000    3.2000    4.3000    5.4000         0

ifail =
         0
```