NAG Toolbox for MATLAB

f01ct

1 Purpose

f01ct adds two double matrices, each one optionally transposed and multiplied by a scalar.

2 Syntax

```
[c, ifail] = f01ct(transa, transb, m, n, alpha, a, beta, b)
```

3 Description

f01ct performs one of the operations

$$C := \alpha A + \beta B,$$

$$C := \alpha A^{\mathrm{T}} + \beta B,$$

$$C := \alpha A + \beta B^{\mathrm{T}} \text{ or}$$

$$C := \alpha A^{\mathrm{T}} + \beta B^{\mathrm{T}},$$

where A, B and C are matrices, and α and β are scalars. For efficiency, the function contains special code for the cases when one or both of α, β is equal to zero, unity or minus unity. The matrices, or their transposes, must be compatible for addition. A and B are either m by n or n by m matrices, depending on whether they are to be transposed before addition. C is an m by n matrix.

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **transa – string**

2: **transb – string**

transa and **transb** must specify whether or not the matrix A and the matrix B, respectively, are to be transposed before addition.

transa or **transb** = 'N'

The matrix will not be transposed.

transa or **transb** = 'T' or 'C'

The matrix will be transposed.

Constraint: **transa** and **transb** must be one of 'N', 'T' or 'C'.

3: **m – int32 scalar**

m, the number of rows of the matrices A and B or their transposes. Also the number of rows of the matrix C.

Constraint: $\mathbf{m} \geq 0$.

4: **n – int32 scalar**

n, the number of columns of the matrices A and B or their transposes. Also the number of columns of the matrix C.

Constraint: $\mathbf{n} \geq 0$.

5: **alpha – double scalar**

The scalar α, by which matrix A is multiplied before addition.

6: **a(lda,$*$) – double array**

The first dimension, **lda**, of the array **a** must satisfy

if **transa** = 'N', $\mathbf{lda} \geq \max(1, \mathbf{m})$;
$\mathbf{lda} \geq \max(1, \mathbf{n})$ otherwise.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ and the leading m by n part of **a** must contain the matrix A, if **transa** = 'N', and at least $\max(1, \mathbf{m})$ and the leading n by m part of **a** must contain the matrix A, otherwise

If $\alpha = 0.0$, the elements of array **a** need not be assigned.

7: **beta – double scalar**

The scalar β, by which matrix B is multiplied before addition.

8: **b(ldb,$*$) – double array**

The first dimension, **ldb**, of the array **b** must satisfy

if **transb** = 'N', $\mathbf{ldb} \geq \max(1, \mathbf{m})$;
$\mathbf{ldb} \geq \max(1, \mathbf{n})$ otherwise.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ and the leading m by n part of **b** must contain the matrix B, if **transb** = 'N', and at least $\max(1, \mathbf{m})$ and the leading n by m part of **b** must contain the matrix B, otherwise

If $\beta = 0.0$, the elements of array **b** need not be assigned.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldc

5.4 Output Parameters

1: **c(ldc,$*$) – double array**

The first dimension of the array **c** must be at least $\max(1, \mathbf{m})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The elements of the m by n matrix C.

2: **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, one or both of **transa** or **transb** is not equal to 'N', 'T' or 'C'.

ifail $= 2$

On entry, one or both of **m** or **n** is less than 0.

ifail $= 3$

On entry, **lda** $< \max(1, P)$, where P $=$ **m** if **transa** $=$ 'N', and P $=$ **n** otherwise.

ifail $= 4$

On entry, **ldb** $< \max(1, P)$, where P $=$ **m** if **transb** $=$ 'N', and P $=$ **n** otherwise.

ifail $= 5$

On entry, **ldc** $< \max(1, \mathbf{m})$.

7 Accuracy

The results returned by f01ct are accurate to ***machine precision***.

8 Further Comments

The time taken for a call of f01ct varies with **m**, **n** and the values of α and β. The function is quickest if either or both of α and β are equal to zero, or plus or minus unity.

9 Example

```
transa = 'N';
transb = 'N';
m = int32(4);
n = int32(3);
alpha = 1;
a = [1, 2.5, 3;
     -2, 2, -1.5;
     3.5, 2, -2.5;
     1.5, -2, 1];
beta = 1;
b = [2, -2.5, -2;
     1, 1, 1;
     -1.5, 2.5, -2.5;
     2, -2, 1];
[c, ifail] = f01ct(transa, transb, m, n, alpha, a, beta, b)
```

```
c =
    3.0000         0    1.0000
   -1.0000    3.0000   -0.5000
    2.0000    4.5000   -5.0000
    3.5000   -4.0000    2.0000
ifail =
           0
```