NAG Toolbox for MATLAB

f11mk

1 Purpose

f11mk computes a matrix-matrix or transposed matrix-matrix product involving a real, square, sparse nonsymmetric matrix stored in compressed column (Harwell–Boeing) format.

2 Syntax

```
[c, ifail] = f11mk(trans, alpha, icolzp, irowix, a, b, beta, c, 'n', n, 'm', m)
```

3 Description

f11mk computes either the matrix-matrix product $C \leftarrow \alpha AB + \beta C$, or the transposed matrix-matrix product $C \leftarrow \alpha A^{\mathrm{T}}B + \beta C$, according to the value of the parameter **trans**, where A is a real n by n sparse nonsymmetric matrix, of arbitrary sparsity pattern with nnz nonzero elements, B and C are n by m real dense matrices. The matrix A is stored in compressed column (Harwell–Boeing) storage format. The array **a** stores all nonzero elements of A, while arrays **icolzp** and **irowix** store the compressed column indices and row indices of A respectively.

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **trans – string**

Specifies whether or not the matrix A is transposed.

trans = 'N'

$\alpha AB + \beta C$ is computed.

trans = 'T'

$\alpha A^{\mathrm{T}}B + \beta C$ is computed.

Constraint: **trans** = 'N' or 'T'.

2: **alpha – double scalar**

α, the scalar factor in the matrix multiplication.

3: **icolzp**$(*)$ **– int32 array**

Note: the dimension of the array **icolzp** must be at least $\mathbf{n} + 1$.

icolzp(i) contains the index in A of the start of a new column. See Section 2.1.3 in the F11 Chapter Introduction.

4: **irowix**$(*)$ **– int32 array**

Note: the dimension of the array **irowix** must be at least **icolzp**$(\mathbf{n} + 1) - 1$, the number of nonzeros of the sparse matrix A.

The row index array of the sparse matrix A.

5: **a**$(*)$ **– double array**

Note: the dimension of the array **a** must be at least **icolzp**$(\mathbf{n}+1)-1$, the number of nonzeros of the sparse matrix A.

The array of nonzero values in the sparse matrix A.

6: **b**(**ldb,***) **– double array**

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

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

The n by m matrix B.

7: **beta – double scalar**

The scalar factor β.

8: **c**(**ldc,***) **– double array**

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

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

The n by m matrix C.

5.2 Optional Input Parameters

1: **n – int32 scalar**

n, the order of the matrix A.

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

2: **m – int32 scalar**

Default: The second dimension of the array **b** The second dimension of the array **c**.

m, the number of columns of matrices B and C.

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

5.3 Input Parameters Omitted from the MATLAB Interface

ldb, ldc

5.4 Output Parameters

1: **c**(**ldc,***) **– double array**

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

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

C contains $\alpha AB+\beta C$ or $\alpha A^{\mathrm{T}}B+\beta C$ depending on the value of **trans**.

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, **trans** $\neq$ 'N' or 'T',
or **n** < 0,
or **m** < 0,
or **ldb** $< \max(1, \mathbf{n})$,
or **ldc** $< \max(1, \mathbf{n})$.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

```
trans = 'N';
alpha = 1;
icolzp = [int32(1);
     int32(3);
     int32(5);
     int32(7);
     int32(9);
     int32(12)];
irowix = [int32(1);
     int32(3);
     int32(1);
     int32(5);
     int32(2);
     int32(3);
     int32(2);
     int32(4);
     int32(3);
     int32(4);
     int32(5)];
a = [2;
     4;
     1;
     -2;
     1;
     1;
     -1;
     1;
     1;
     2;
     3];
b = [0.7, 1.4;
     0.16, 0.32;
     0.52, 1.04;
     0.77, 1.54;
     0.28, 0.5600000000000001];
beta = 0;
c = [0, 0;
     0, 0;
     0, 0;
     0, 0;
     0, 0];
[cOut, ifail] = f11mk(trans, alpha, icolzp, irowix, a, b, beta, c)
```

```
cOut =
    1.5600    3.1200
   -0.2500   -0.5000
    3.6000    7.2000
    1.3300    2.6600
    0.5200    1.0400
ifail =
           0
```