

NAG Toolbox for MATLAB

m01de

1 Purpose

m01de ranks the rows of a matrix of double numbers in ascending or descending order.

2 Syntax

```
[irank, ifail] = m01de(rm, m1, m2, n1, order, 'n2', n2)
```

3 Description

m01de ranks rows **m1** to **m2** of a matrix, using the data in columns **n1** to **n2** of those rows. The ordering is determined by first ranking the data in column **n1**, then ranking any tied rows according to the data in column **n1** + 1, and so on up to column **n2**.

m01de uses a variant of list-merging, as described by Knuth 1973, pp 165-166. The function takes advantage of natural ordering in the data, and uses a simple list insertion in a preparatory pass to generate ordered lists of length at least 10. The ranking is stable: equal rows preserve their ordering in the input data.

4 References

Knuth D E 1973 *The Art of Computer Programming (Volume 3)* (2nd Edition) Addison–Wesley

5 Parameters

5.1 Compulsory Input Parameters

- 1: **rm(ldm,n2) – double array**
ldm, the first dimension of the array, must be at least **m2**.
Columns **n1** to **n2** of rows **m1** to **m2** of **rm** must contain double data to be ranked.
- 2: **m1 – int32 scalar**
The index of the first row of **rm** to be ranked.
Constraint: **m1** > 0.
- 3: **m2 – int32 scalar**
the index of the last row of **rm** to be ranked.
Constraint: **m2** ≥ **m1**.
- 4: **n1 – int32 scalar**
The index of the first column of **rm** to be used.
Constraint: **n1** > 0.
- 5: **order – string**
If **order** = 'A', the rows will be ranked in ascending (i.e., nondecreasing) order.
If **order** = 'D', into descending order.
Constraint: **order** = 'A' or 'D'.

5.2 Optional Input Parameters

1: **n2** – int32 scalar

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

the index of the last column of **rm** to be used.

Constraint: $\mathbf{n2} \geq \mathbf{n1}$.

5.3 Input Parameters Omitted from the MATLAB Interface

ldm

5.4 Output Parameters

1: **irank(m2)** – int32 array

Elements **m1** to **m2** of **irank** contain the ranks of the corresponding rows of **rm**. Note that the ranks are in the range **m1** to **m2**: thus, if the *i*th row of **rm** is the first in the rank order, **irank**(*i*) is set to **m1**.

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, **m2** < 1,
or **n2** < 1,
or **m1** < 1,
or **m1** > **m2**,
or **n1** < 1,
or **n1** > **n2**,
or **ldm** < **m2**.

ifail = 2

On entry, **order** is not 'A' or 'D'.

7 Accuracy

Not applicable.

8 Further Comments

The average time taken by the function is approximately proportional to $n \times \log n$, where $n = \mathbf{m2} - \mathbf{m1} + 1$.

9 Example

```
rm = [6, 5, 4;
      5, 2, 1;
      2, 4, 9;
      4, 9, 6;
      4, 9, 5;
```

```
    4, 1, 2;  
    3, 4, 1;  
    2, 4, 6;  
    1, 6, 4;  
    9, 3, 2;  
    6, 2, 5;  
    4, 9, 6];  
m1 = int32(1);  
m2 = int32(12);  
n1 = int32(1);  
order = 'Ascending';  
[irank, ifail] = m01de(rm, m1, m2, n1, order)
```

```
irank =  
      11  
       9  
       3  
       7  
       6  
       5  
       4  
       2  
       1  
      12  
      10  
       8  
ifail =  
       0
```