

NAG Toolbox for MATLAB

m01cc

1 Purpose

m01cc rearranges a vector of character data so that a specified substring is in ASCII or reverse ASCII order.

2 Syntax

```
[ch, ifail] = m01cc(ch, m1, l1, l2, order, 'm2', m2)
```

3 Description

m01cc is based on Singleton's implementation of the 'median-of-three' Quicksort algorithm (see Singleton 1969), but with two additional modifications. First, small subfiles are sorted by an insertion sort on a separate final pass (see Sedgewick 1978). Second, if a subfile is partitioned into two very unbalanced subfiles, the larger of them is flagged for special treatment: before it is partitioned, its end points are swapped with two random points within it; this makes the worst case behaviour extremely unlikely.

Only the substring (**l1:l2**) of each element of the array **ch** is used to determine the sorted order, but the entire elements are rearranged into sorted order.

4 References

Sedgewick R 1978 Implementing Quicksort programs *Comm. ACM* **21** 847–857

Singleton R C 1969 An efficient algorithm for sorting with minimal storage: Algorithm 347 *Comm. ACM* **12** 185–187

5 Parameters

5.1 Compulsory Input Parameters

1: **ch(m2)** – string array

Elements **m1** to **m2** of **ch** must contain character data to be sorted.

Constraint: the length of each element of **ch** must not exceed 255.

2: **m1** – int32 scalar

The index of the first element of **ch** to be sorted.

Constraint: **m1** > 0.

3: **l1** – int32 scalar

4: **l2** – int32 scalar

Only the substring (**l1:l2**) of each element of **ch** is to be used in determining the sorted order.

Constraint: $0 < \mathbf{l1} \leq \mathbf{l2} \leq \text{LEN}(\mathbf{ch}(1))$.

5: **order** – string

If **order** = 'A', the values will be sorted into ASCII order.

If **order** = 'R', into reverse ASCII order.

Constraint: **order** = 'A' or 'R'.

5.2 Optional Input Parameters

1: **m2** – int32 scalar

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

the index of the last element of **ch** to be sorted.

Constraint: $m2 \geq m1$.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **ch(m2)** – string array

These values are rearranged into sorted order.

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 **m1** < 1,
or **m1** > **m2**,
or **l2** < 1,
or **l1** < 1,
or **l1** > **l2**,
or **l2** > LEN(**ch**(1)).

ifail = 2

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

ifail = 3

On entry, the length of each element of **ch** exceeds 255.

7 Accuracy

Not applicable.

8 Further Comments

The average time taken by the function is approximately proportional to $n \times \log n$, where $n = m2 - m1 + 1$. The worst case time is proportional to n^2 , but this is extremely unlikely to occur.

The function relies on the Fortran 77 intrinsic functions LLT and LGT to order characters according to the ASCII collating sequence.

9 Example



```
ch = {'A02AAF 289';
      'A02ABF 523';
      'A02ACF 531';
      'C02ADF 169';
      'C02AEF 599';
      'C05ADF 1351';
      'C05AGF 240';
      'C05AJF 136';
      'C05AVF 211';
      'C05AXF 183';
      'C05AZF 2181'};
m1 = int32(1);
l1 = int32(7);
l2 = int32(12);
order = 'Reverse ASCII';
[chOut, ifail] = m01cc(ch, m1, l1, l2, order)

chOut =
    'C05AZF 2181'
    'C05ADF 1351'
    'C02AEF 599'
    'A02ACF 531'
    'A02ABF 523'
    'A02AAF 289'
    'C05AGF 240'
    'C05AVF 211'
    'C05AXF 183'
    'C02ADF 169'
    'C05AJF 136'
ifail =
    0
```