

NAG Library Routine Document

M01NCF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

1 Purpose

M01NCF examines an ordered vector of null terminated strings and returns the index of the first value equal to the sought-after item. Character items are compared according to the ASCII collating sequence.

2 Specification

```
INTEGER FUNCTION M01NCF(VALID, CH, M1, M2, ITEM, IFAIL)
INTEGER                                M1, M2, IFAIL
LOGICAL                               VALID
CHARACTER*(*)                         CH(M2), ITEM
```

3 Description

M01NCF is based on Professor Niklaus Wirth's implementation of the Binary Search algorithm (see Wirth (2004)), but with two modifications. First, if the sought-after item is less than the value of the first element of the array to be searched, 0 is returned. Second, if a value equal to the sought-after item is not found, the index of the immediate lower value is returned.

4 References

Wirth N (2004) *Algorithms and Data Structures* 35–36 Prentice Hall

5 Parameters

- 1: VALID – LOGICAL *Input*
On entry: if VALID is set to .TRUE. parameter checking will be performed. If VALID is set to .FALSE. M01NCF will be called without parameter checking, which includes checking that array CH is sorted in ascending order and the routine will return with IFAIL = 0. See Section 8 for further details.
- 2: CH(M2) – CHARACTER*(*) array *Input*
On entry: elements M1 to M2 contain character data to be searched.
Constraint: elements M1 to M2 of CH must be sorted in ascending order. The length of each element of CH must not exceed 255. Trailing space characters are ignored.
- 3: M1 – INTEGER *Input*
On entry: the index of the first element of CH to be searched.
Constraint: $M1 \geq 1$.
- 4: M2 – INTEGER *Input*
On entry: the index of the last element of CH to be searched.
Constraint: $M2 \geq M1$.

5: ITEM – CHARACTER*(*) *Input*

On entry: the sought-after item. Trailing space characters are ignored.

6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

(**Note:** these errors will only be returned if VALID = .TRUE..)

IFAIL = 2

On entry, elements M1 to M2 of CH are not sorted in ASCII order.

IFAIL = 3

On entry, $M2 < 1$.

IFAIL = 4

On entry, $M1 > M2$.

IFAIL = 5

On entry, the length of elements of CH exceeds 255.

7 Accuracy

Not applicable.

8 Further Comments

The argument VALID should be used with caution. Set it to .FALSE. only if you are confident that the other arguments are correct, in particular that array CH is in fact arranged in ascending order. If you wish to search the same array CH many times, you are recommended to set VALID to .TRUE. on first call of M01NCF and to .FALSE. on subsequent calls, in order to minimize the amount of time spent checking CH, which may be significant if CH is large.

The time taken by M01NCF is $O(\log n)$, where $n = M2 - M1 + 1$, when VALID = .FALSE..

9 Example

This example reads a list of character data and sought-after items and performs the search for these items.

9.1 Program Text

```

*      M01NCF Example Program Test
*      NAG COPYRIGHT 2008.
      IMPLICIT      NONE
*
*      .. Parameters ..
      INTEGER       NIN, NOUT
      PARAMETER     (NIN=5,NOUT=6)
      INTEGER       CHMAX
      PARAMETER     (CHMAX=100)
*
*      .. Local Scalars ..
      INTEGER       I, IFAIL, INDEX, LENCH
      LOGICAL       FIRST
      CHARACTER*6   ITEM
*
*      .. Local Arrays ..
      CHARACTER*6   CH(CHMAX)
*
*      .. External Functions ..
      INTEGER       M01NCF
      EXTERNAL      M01NCF
*
*      .. Executable Statements ..
      WRITE (NOUT,*) 'M01NCF Example Program Results'
      WRITE (NOUT,*)

*
*      Read in Reference Vector CH
      READ (NIN,*)
      READ (NIN,*) LENCH
      IF (LENCH.LE.CHMAX) THEN
        READ (NIN,*) (CH(I),I=1,LENCH)
*
*      Read items sought in the reference vector
        FIRST = .TRUE.
20      READ (NIN,'(A)',END=40) ITEM
*
        IFAIL = 1
        INDEX = M01NCF(FIRST,CH,1,10,ITEM,IFAIL)
*
        IF (IFAIL.EQ.0) THEN
          IF (FIRST) THEN
*            Print the reference vector
            WRITE (NOUT,*) 'Reference Vector is:'
            WRITE (NOUT,99999) (CH(I),I=1,LENCH)
            FIRST = .FALSE.
          END IF
*
          WRITE (NOUT,*)
          WRITE (NOUT,99998) ITEM, INDEX
          GO TO 20
        ELSE
          WRITE (NOUT,99997) IFAIL
        END IF
      ELSE
        WRITE (NOUT,99996) CHMAX, LENCH
      END IF
40      CONTINUE
*
99999  FORMAT (10(1X,A))
99998  FORMAT (1X,'Search for item ',A,' returned index: ',I4)
99997  FORMAT (1X,'** M01NCF returned with IFAIL = ',I5)
99996  FORMAT (1X,'** Length of reference vector = ',I4,' but must not ',
+         'exceed ',I3)
      END

```

9.2 Program Data

M01NCF Example Program Data

10	: LENCH
A02AAF A02ABF A02ACF C02ADF C02AEF	
C05ADF C05AGF C05AJG C05AXF C05AZF	: CH
C02ADF	: Item 1
A01AAF	: Item 2
C04AYF	: Item 3
D01NBF	: Item 4

9.3 Program Results

M01NCF Example Program Results

Reference Vector is:

A02AAF A02ABF A02ACF C02ADF C02AEF C05ADF C05AGF C05AJG C05AXF C05AZF

Search for item C02ADF returned index: 4

Search for item A01AAF returned index: 0

Search for item C04AYF returned index: 5

Search for item D01NBF returned index: 10
