NAG Library Routine Document

M01NAF

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

M01NAF searches an ordered vector of real numbers and returns the index of the first value equal to the sought-after item.

2 Specification

```
FUNCTION M01NAF (VALID, RV, M1, M2, ITEM, IFAIL)
INTEGER M01NAF
INTEGER              M1, M2, IFAIL
REAL (KIND=nag_wp) RV(M2), ITEM
LOGICAL              VALID
```

3 Description

M01NAF 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. M01NAF will be called without parameter checking (which includes checking that array RV is sorted in ascending order) and the routine will return with IFAIL $= 0$. See Section 8 for further details.

2: RV(M2) – REAL (KIND=nag_wp) array *Input*

On entry: elements M1 to M2 contain real values to be searched.

Constraint: elements M1 to M2 of RV must be sorted in ascending order.

3: M1 – INTEGER *Input*

On entry: the index of the first element of RV to be searched.

Constraint: M1 ≥ 1.

4: M2 – INTEGER *Input*

On entry: the index of the last element of RV to be searched.

Constraint: M2 $\geq$ M1.

5: ITEM – REAL (KIND=nag_wp) *Input*

On entry: the sought-after item.

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.

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.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

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 RV are not sorted in ascending order.

IFAIL $= 3$

On entry, M2 $<$ 1.

IFAIL $= 4$

On entry, M1 $>$ M2.

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 RV is in fact arranged in ascending order. If you wish to search the same array RV many times, you are recommended to set VALID to .TRUE. on first call of M01NAF and to .FALSE. on subsequent calls, in order to minimize the amount of time spent checking RV, which may be significant if RV is large.

The time taken by M01NAF is $O(\log n)$, where $n =$ M2 $-$ M1 $+ 1$, when VALID = .FALSE..

9 Example

This example reads a list of double precision numbers and sought-after items and performs the search for these items.

9.1 Program Text

```
    Program m01nafe

!     M01NAF Example Program Test

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: m01naf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: item
      Integer                          :: i, ifail, index, ioerr, m1, m2
      Logical                          :: first
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: rv(:)
!     .. Executable Statements ..
      Write (nout,*) 'M01NAF Example Program Results'

!     Skip heading in data file
      Read (nin,*)

      Read (nin,*) m2
      Allocate (rv(m2))

      m1 = 1

      Read (nin,*)(rv(i),i=m1,m2)

      first = .True.

data: Do
        Read (nin,*,Iostat=ioerr) item

        If (ioerr<0) Then
          Exit data
        End If

        ifail = 0
        index = m01naf(first,rv,m1,m2,item,ifail)

        If (first) Then
          Write (nout,*)
          Write (nout,*) 'Reference Vector is:'
          Write (nout,99999)(rv(i),i=m1,m2)
          first = .False.
        End If

        Write (nout,*)
        Write (nout,99998) item, index
      End Do data

99999 Format (1X,8F7.1)
99998 Format (1X,'Search for item ',F7.1,' returned index: ',I4)
    End Program m01nafe
```

9.2 Program Data

```
M01NAF Example Program Data
16                                                 : M2
0.5 0.6 1.1 1.2 1.3 1.3 2.1 2.3
2.3 4.1 5.8 5.9 6.5 6.5 8.6 9.9                    : RV
2.1                                                : Item 1
0.4                                                : Item 2
7.1                                                : Item 3
10.0                                               : Item 4
```

9.3 Program Results

```
 M01NAF Example Program Results

 Reference Vector is:
     0.5    0.6    1.1    1.2    1.3    1.3    2.1    2.3
     2.3    4.1    5.8    5.9    6.5    6.5    8.6    9.9

 Search for item     2.1 returned index:    7

 Search for item     0.4 returned index:    0

 Search for item     7.1 returned index:   14

 Search for item    10.0 returned index:   16
```