

# NAG Library Routine Document

## F16JRF (BLAS\_DAMIN\_VAL)

**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

F16JRF (BLAS\_DAMIN\_VAL) computes, with respect to absolute value, the smallest component of a real vector, along with the index of that component.

### 2 Specification

```
SUBROUTINE F16JRF (N, X, INCX, K, R)
  INTEGER          N, INCX, K
  REAL (KIND=nag_wp) X(1+(N-1)*ABS(INCX)), R
```

The routine may be called by its BLAST name *blas\_damin\_val*.

### 3 Description

F16JRF (BLAS\_DAMIN\_VAL) computes, with respect to absolute value, the smallest component,  $r$ , of an  $n$ -element real vector  $x$ , and determines the smallest index,  $k$ , such that

$$r = |x_k| = \min_j |x_j|.$$

### 4 References

Basic Linear Algebra Subprograms Technical (BLAST) Forum (2001) *Basic Linear Algebra Subprograms Technical (BLAST) Forum Standard* University of Tennessee, Knoxville, Tennessee <http://www.netlib.org/blas/blast-forum/blas-report.pdf>

### 5 Arguments

- |    |                                                                                                                                                                                         |               |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| 1: | N – INTEGER                                                                                                                                                                             | <i>Input</i>  |
|    | <i>On entry:</i> $n$ , the number of elements in $x$ .                                                                                                                                  |               |
| 2: | $X(1 + (N - 1) \times  INCX )$ – REAL (KIND=nag_wp) array                                                                                                                               | <i>Input</i>  |
|    | <i>On entry:</i> the $n$ -element vector $x$ .                                                                                                                                          |               |
|    | If $INCX > 0$ , $x_i$ must be stored in $X((i - 1) \times INCX + 1)$ , for $i = 1, 2, \dots, N$ .                                                                                       |               |
|    | If $INCX < 0$ , $x_i$ must be stored in $X((N - i) \times  INCX  + 1)$ , for $i = 1, 2, \dots, N$ .                                                                                     |               |
|    | Intermediate elements of $X$ are not referenced. If $N = 0$ , $X$ is not referenced.                                                                                                    |               |
| 3: | INCX – INTEGER                                                                                                                                                                          | <i>Input</i>  |
|    | <i>On entry:</i> the increment in the subscripts of $X$ between successive elements of $x$ .                                                                                            |               |
|    | <i>Constraint:</i> $INCX \neq 0$ .                                                                                                                                                      |               |
| 4: | K – INTEGER                                                                                                                                                                             | <i>Output</i> |
|    | <i>On exit:</i> $k$ , the index, from the set $\{1, 2, \dots, N\}$ , of the smallest component of $x$ with respect to absolute value. If $N \leq 0$ on input then $K$ is returned as 0. |               |

5: R – REAL (KIND=nag\_wp)

*Output*

*On exit:*  $r$ , the smallest component of  $x$  with respect to absolute value. If  $N \leq 0$  on input then R is returned as 0.0.

## 6 Error Indicators and Warnings

If  $INCX = 0$ , an error message is printed and program execution is terminated.

## 7 Accuracy

The BLAS standard requires accurate implementations which avoid unnecessary over/underflow (see Section 2.7 of Basic Linear Algebra Subprograms Technical (BLAST) Forum (2001)).

## 8 Parallelism and Performance

F16JRF (BLAS\_DAMIN\_VAL) is not threaded in any implementation.

## 9 Further Comments

None.

## 10 Example

This example computes the smallest component with respect to absolute value and index of that component for the vector

$$x = (1, 10, 11, -2, 9)^T.$$

### 10.1 Program Text

```

Program f16jrfe

!      F16JRF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
!      Use nag_library, Only: blas_damin_val, nag_wp
!      .. Implicit None Statement ..
!      Implicit None
!      .. Parameters ..
!      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
!      Real (Kind=nag_wp)          :: r
!      Integer                     :: i, incx, ix, k, n
!      .. Local Arrays ..
!      Real (Kind=nag_wp), Allocatable :: x(:)
!      .. Intrinsic Procedures ..
!      Intrinsic                   :: abs
!      .. Executable Statements ..
!      Write (nout,*) 'F16JRF Example Program Results'

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

!      Read (nin,*) n, incx
!      Allocate (x(1+(n-1)*abs(incx)))

!      Read the vector x and store forwards or backwards
!      as determined by incx.
!      If (incx>0) Then
!          ix = 1
!      Else

```

```

        ix = 1 - (n-1)*incx
    End If

    Do i = 1, n
        Read (nin,*) x(ix)
        ix = ix + incx
    End Do

!      Find k = argmin(abs(x)) and r = min(abs(x)).

    Call blas_damin_val(n,x,incx,k,r)

    Write (nout,*)
    Write (nout,99999) k
    Write (nout,99998) r

99999 Format (1X,'Index of absolutely smallest component of x is',I3)
99998 Format (1X,'Absolutely smallest component of x is',F12.5)
    End Program fl6jrfe

```

## 10.2 Program Data

F16JRF Example Program Data

5	1	: n and incx
1.0		
10.0		
11.0		
-2.0		
9.0		: Vector x

## 10.3 Program Results

F16JRF Example Program Results

```

Index of absolutely smallest component of x is 1
Absolutely smallest component of x is      1.00000

```

---