

NAG Library Routine Document

F16JPF (BLAS_DMIN_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

F16JPF (BLAS_DMIN_VAL) computes the smallest component of a real vector, along with the index of that component.

2 Specification

```
SUBROUTINE F16JPF (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_dmin_val*.

3 Description

F16JPF (BLAS_DMIN_VAL) computes 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 Parameters

- | | | |
|----|--|---------------|
| 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 vector x . Element x_i is stored in $X((i - 1) \times INCX + 1)$, for $i = 1, 2, \dots, n$. | |
| 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 $\{1, 1 + INCX , \dots, 1 + (N - 1) \times INCX \}$, of the smallest component of x . If $N \leq 0$ on input then K is returned as 0. | |
| 5: | R – REAL (KIND=nag_wp) | <i>Output</i> |
| | <i>On exit:</i> r , the smallest component of x . 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 Further Comments

None.

9 Example

This example computes the smallest component and index of that component for the vector

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

9.1 Program Text

```

Program f16jpf

!      F16JPF Example Program Text

!      Mark 24 Release. NAG Copyright 2012.

!      .. Use Statements ..
      Use nag_library, Only: blas_dmin_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, k, n
!      .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable :: x(:)
!      .. Intrinsic Procedures ..
      Intrinsic                   :: abs
!      .. Executable Statements ..
      Write (nout,*) 'F16JPF Example Program Results'

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

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

      Read (nin,*)(x(i),i=1,1+(n-1)*abs(incx),incx)

!      Find K = ARGMIN(X) and R = MIN(X).

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

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

99999 Format (1X,'Index of smallest component of X is',I3)
99998 Format (1X,'Smallest component of X is',F12.5)
      End Program f16jpf

```

9.2 Program Data

F16JPF Example Program Data

```
5 1
1.0 10.0 11.0 -2.0 9.0
```

```
: N and INCX
: Array X
```

9.3 Program Results

F16JPF Example Program Results

```
Index of smallest component of X is 4
Smallest component of X is -2.00000
```
