

# NAG Library Routine Document

## F16DNF

**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

F16DNF computes the largest component of an integer vector, along with the index of that component.

### 2 Specification

```
SUBROUTINE F16DNF (N, X, INCX, K, I)
INTEGER N, X(1+(N-1)*ABS(INCX)), INCX, K, I
```

### 3 Description

F16DNF computes the largest component,  $i$ , of an  $n$ -element integer vector  $x$ , and determines the smallest index,  $k$ , such that

$$i = x_k = \max_j x_j.$$

### 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: | $X(1 + (N - 1) \times  INCX )$ – INTEGER 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 $\{1, 1 +  INCX , \dots, 1 + (N - 1) \times  INCX \}$ , of the largest component of $x$ . If $N \leq 0$ on input then $K$ is returned as 0. |               |
| 5: | I – INTEGER   | <i>Output</i> |
|    | <i>On exit:</i> $i$ , the largest component of $x$ . If $N \leq 0$ on input then $I$ is returned as 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

Not applicable.

## 9 Further Comments

None.

## 10 Example

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

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

### 10.1 Program Text

```

Program f16dnfe

!      F16DNF Example Program Text

!      Mark 25 Release. NAG Copyright 2014.

!      .. Use Statements ..
Use nag_library, Only: f16dnf
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Integer                    :: i, incx, j, k, n
!      .. Local Arrays ..
Integer, Allocatable       :: x(:)
!      .. Intrinsic Procedures ..
Intrinsic                  :: abs
!      .. Executable Statements ..
Write (nout,*) 'F16DNF Example Program Results'

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

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

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

!      Find K = ARGMAX(X) and I = MAX(X).

      Call f16dnf(n,x,incx,k,i)

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

99999 Format (1X,'Index of largest component of X is',I3)
99998 Format (1X,'Largest component of X is',I12)
End Program f16dnfe

```

## 10.2 Program Data

F16DNF Example Program Data

```
5 1
1 10 11 -2 9
```

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

## 10.3 Program Results

F16DNF Example Program Results

```
Index of largest component of X is 3
Largest component of X is 11
```

---