

NAG Library Routine Document

F16GLF (BLAS_ZSUM)

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

F16GLF (BLAS_ZSUM) sums the elements of a complex vector.

2 Specification

```
FUNCTION F16GLF (N, X, INCX)
COMPLEX (KIND=nag_wp) F16GLF
INTEGER N, INCX
COMPLEX (KIND=nag_wp) X(1+(N-1)*ABS(INCX))
```

The routine may be called by its BLAST name *blas_zsum*.

3 Description

F16GLF (BLAS_ZSUM) returns the sum

$$x_1 + x_2 + \cdots + x_n$$

of the elements of an n -element complex vector x , via the function name.

If $N \leq 0$ on entry, F16GLF (BLAS_ZSUM) returns the value $0 + 0i$.

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: | $X(1 + (N - 1) \times INCX)$ – COMPLEX (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> $INCX \neq 0$. | |

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

F16GLF (BLAS_ZSUM) is not threaded in any implementation.

9 Further Comments

None.

10 Example

This example computes the sum of the elements of

$$x = (1.1 + 10.2i, 11.5 - 2.7i, 9.2)^T.$$

10.1 Program Text

```

Program f16glfe

!      F16GLF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
      Use nag_library, Only: blas_zsum, nag_wp
!      .. Implicit None Statement ..
      Implicit None
!      .. Parameters ..
      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
      Complex (Kind=nag_wp)      :: sumval
      Integer                     :: i, incx, ix, n
!      .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: x(:)
!      .. Intrinsic Procedures ..
      Intrinsic                   :: abs
!      .. Executable Statements ..
      Write (nout,*) 'F16GLF 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

!      Sum the elements of x

      sumval = blas_zsum(n,x,incx)

```

```
      Write (nout,*)  
      Write (nout,99999) sumval  
  
99999 Format (1X,'Sum of elements of x is (',F9.5,',',F9.5,')')  
      End Program f16glfe
```

10.2 Program Data

F16GLF Example Program Data

```
  3  1                                     : n and incx  
  ( 1.1, 10.2)  
  ( 11.5,-2.7)  
  ( 9.2, 0.)                               : Vector x
```

10.3 Program Results

F16GLF Example Program Results

```
Sum of elements of x is ( 21.80000, 7.50000)
```
