NAG Library Routine Document

F16EHF (BLAS_DWAXPBY)

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

F16EHF (BLAS_DWAXPBY) performs scaled vector addition, for real vectors and scalars.

2 Specification

```
SUBROUTINE F16EHF(N, ALPHA, X, INCX, BETA, Y, INCY, W, INCW)
INTEGER           N, INCX, INCY, INCW
double precision  ALPHA, X(1+(N-1)*ABS(INCX)), BETA, Y(1+(N-
1                 1)*ABS(INCY)), W(1+(N-1)*ABS(INCW))
```

The routine may be called by its BLAS name ***blas_dwaxpby***.

3 Description

F16EHF (BLAS_DWAXPBY) performs the operation

$$w \leftarrow \alpha x + \beta y,$$

where x and y are n-element real vectors, and α and β are real scalars.

4 References

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

5 Parameters

1: N – INTEGER *Input*

On entry: n, the number of elements in x, y and w.

2: ALPHA – ***double precision*** *Input*

On entry: the scalar α.

3: X$(1 + (\text{N} - 1) \times |\text{INCX}|)$ – ***double precision*** array *Input*

On entry: the vector x. Element x_i is stored in X$(1 + (i - 1) \times |\text{INCX}|)$, for $i = 1, 2, \ldots, n$.

4: INCX – INTEGER *Input*

On entry: the increment in the subscripts of X between successive elements of x.

Constraint: INCX $\neq 0$.

5: BETA – ***double precision*** *Input*

On entry: the scalar β.

6: Y$(1 + (\text{N} - 1) \times |\text{INCY}|)$ – ***double precision*** array *Input*

On entry: the vector y. Element y_i is stored in Y$(1 + (i - 1) \times |\text{INCY}|)$, for $i = 1, 2, \ldots, n$.

7: INCY – INTEGER *Input*

On entry: the increment in the subscripts of Y between successive elements of y.

Constraint: INCY $\neq 0$.

8: W$(1 + (\text{N} - 1) \times |\text{INCW}|)$ – ***double precision*** array *Output*

On exit: w, the result of the scaled-vector addition. Element w_i is stored in W$(1 + (i - 1) \times |\text{INCW}|)$, for $i = 1, 2, \ldots, n$.

9: INCW – INTEGER *Input*

On entry: the increment in the subscripts of W between successive elements of w.

Constraint: INCW $\neq 0$.

6 Error Indicators and Warnings

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

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example computes the result of a scaled-vector addition for

$$\begin{aligned} \alpha &= 3, & x &= (-4, 2.1, 3.7, 4.5, -6)^{\mathrm{T}}, \\ \beta &= -1, & y &= (-3, -2.4, 6.4, -5, -5.1)^{\mathrm{T}}. \end{aligned}$$

9.1 Program Text

```
*     F16EHF Example Program Text
*     Mark 22 Release. NAG Copyright 2007.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, INCMAX
      PARAMETER        (NMAX=10,INCMAX=NMAX)
*     .. Local Scalars ..
      DOUBLE PRECISION ALPHA, BETA
      INTEGER          I, INCW, INCX, INCY, N
*     .. Local Arrays ..
      DOUBLE PRECISION W(1+(NMAX-1)*ABS(INCMAX)),
     +                 X(1+(NMAX-1)*ABS(INCMAX)),
     +                 Y(1+(NMAX-1)*ABS(INCMAX))
*     .. External Subroutines ..
      EXTERNAL         BLAS_DWAXPBY
*     .. Intrinsic Functions ..
      INTRINSIC        ABS
*     .. Executable Statements ..
      CONTINUE
*
      WRITE (NOUT,*) 'F16EHF/BLAS_DWAXPBY Example Program Results'
*
*     Skip heading in data file
*
      READ (NIN,*)
*
```

```
*     Read N from data file
*
      READ (NIN,*) N
*
*     Read INCX, INCY and INCW from data file
*
      READ (NIN,*) INCX, INCY, INCW
*
      IF (N.LE.NMAX .AND. ABS(INCX).LE.INCMAX .AND. ABS(INCY)
     +    .LE.INCMAX .AND. ABS(INCW).LE.INCMAX) THEN
*
*        Read ALPHA and BETA from data file
*
         READ (NIN,*) ALPHA, BETA
*
*        Read X and Y from data file
*
         READ (NIN,*) (X(I),I=1,1+(N-1)*ABS(INCX),INCX)
         READ (NIN,*) (Y(I),I=1,1+(N-1)*ABS(INCY),INCY)
*
*        Compute W = ALPHA*X + BETA*Y
*
         CALL BLAS_DWAXPBY(N,ALPHA,X,INCX,BETA,Y,INCY,W,INCW)
*
         WRITE (NOUT,*)
         WRITE (NOUT,99999)
         WRITE (NOUT,99998) (W(I),I=1,N)
      END IF
*
99999 FORMAT (1X,'Result of scaled vector addition is')
99998 FORMAT (1X,'W =',5F9.4)
      END
```

9.2 Program Data

```
F16EHF/BLAS_DWAXPBY Example Program Data
  5                                                    : N
  1   1   1                                            : INCX, INCY and INCW
  3.0   -1.0                                           : ALPHA and BETA
  -4.0   2.1   3.7   4.5   -6.0                        : X
  -3.0   -2.4   6.4   -5.0   -5.1                      : Y
```

9.3 Program Results

```
 F16EHF/BLAS_DWAXPBY Example Program Results

 Result of scaled vector addition is
 W =  -9.0000   8.7000   4.7000  18.5000 -12.9000
```