

# NAG Library Routine Document

## F06PAF (DGEMV)

**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

F06PAF (DGEMV) computes the matrix-vector product for a real general matrix or its transpose.

### 2 Specification

```
SUBROUTINE F06PAF (TRANS, M, N, ALPHA, A, LDA, X, INCX, BETA, Y, INCY)
  INTEGER          M, N, LDA, INCX, INCY
  REAL (KIND=nag_wp) ALPHA, A(LDA,*), X(*), BETA, Y(*)
  CHARACTER(1)    TRANS
```

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

### 3 Description

F06PAF (DGEMV) performs one of the matrix-vector operations

$$y \leftarrow \alpha Ax + \beta y, \quad \text{or} \quad y \leftarrow \alpha A^T x + \beta y,$$

where  $A$  is an  $m$  by  $n$  real matrix,  $x$  and  $y$  are real vectors, and  $\alpha$  and  $\beta$  are real scalars.

If  $m = 0$  or  $n = 0$ , no operation is performed.

### 4 References

None.

### 5 Parameters

- 1: TRANS – CHARACTER(1) *Input*  
*On entry:* specifies the operation to be performed.  
 TRANS = 'N'  
 $y \leftarrow \alpha Ax + \beta y.$   
 TRANS = 'T' or 'C'  
 $y \leftarrow \alpha A^T x + \beta y.$   
*Constraint:* TRANS = 'N', 'T' or 'C'.
- 2: M – INTEGER *Input*  
*On entry:*  $m$ , the number of rows of the matrix  $A$ .  
*Constraint:*  $M \geq 0$ .
- 3: N – INTEGER *Input*  
*On entry:*  $n$ , the number of columns of the matrix  $A$ .  
*Constraint:*  $N \geq 0$ .

- 4: ALPHA – REAL (KIND=nag\_wp) Input  
*On entry:* the scalar  $\alpha$ .
- 5: A(LDA,\*) – REAL (KIND=nag\_wp) array Input  
**Note:** the second dimension of the array A must be at least N.  
*On entry:* the  $m$  by  $n$  matrix  $A$ .
- 6: LDA – INTEGER Input  
*On entry:* the first dimension of the array A as declared in the (sub)program from which F06PAF (DGEMV) is called.  
*Constraint:*  $LDA \geq \max(1, M)$ .
- 7: X(\*) – REAL (KIND=nag\_wp) array Input  
**Note:** the dimension of the array X must be at least  $\max(1, 1 + (N - 1) \times |INCX|)$  if TRANS = 'N' and at least  $\max(1, 1 + (M - 1) \times |INCX|)$  if TRANS = 'T' or 'C'.  
*On entry:* the vector  $x$ .  
 If TRANS = 'N',  
   if  $INCX > 0$ ,  $x_i$  must be stored in  $X(1 + (i - 1) \times INCX)$ , for  $i = 1, 2, \dots, N$ ;  
   if  $INCX < 0$ ,  $x_i$  must be stored in  $X(1 - (N - i) \times INCX)$ , for  $i = 1, 2, \dots, N$ .  
 If TRANS = 'T' or 'C',  
   if  $INCX > 0$ ,  $x_i$  must be stored in  $X(1 + (i - 1) \times INCX)$ , for  $i = 1, 2, \dots, M$ ;  
   if  $INCX < 0$ ,  $x_i$  must be stored in  $X(1 - (M - i) \times INCX)$ , for  $i = 1, 2, \dots, M$ .
- 8: INCX – INTEGER Input  
*On entry:* the increment in the subscripts of X between successive elements of  $x$ .  
*Constraint:*  $INCX \neq 0$ .
- 9: BETA – REAL (KIND=nag\_wp) Input  
*On entry:* the scalar  $\beta$ .
- 10: Y(\*) – REAL (KIND=nag\_wp) array Input/Output  
**Note:** the dimension of the array Y must be at least  $\max(1, 1 + (M - 1) \times |INCY|)$  if TRANS = 'N' and at least  $\max(1, 1 + (N - 1) \times |INCY|)$  if TRANS = 'T' or 'C'.  
*On entry:* the vector  $y$ , if BETA = 0.0, Y need not be set.  
 If TRANS = 'N',  
   if  $INCY > 0$ ,  $y_i$  must be stored in  $Y(1 + (i - 1) \times INCY)$ , for  $i = 1, 2, \dots, M$ ;  
   if  $INCY < 0$ ,  $y_i$  must be stored in  $Y(1 - (M - i) \times INCY)$ , for  $i = 1, 2, \dots, M$ .  
 If TRANS = 'T' or 'C',  
   if  $INCY > 0$ ,  $y_i$  must be stored in  $Y(1 + (i - 1) \times INCY)$ , for  $i = 1, 2, \dots, N$ ;  
   if  $INCY < 0$ ,  $y_i$  must be stored in  $Y(1 - (N - i) \times INCY)$ , for  $i = 1, 2, \dots, N$ .  
*On exit:* the updated vector  $y$  stored in the array elements used to supply the original vector  $y$ .

11: INCY – INTEGER

*Input*

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

*Constraint:*  $\text{INCY} \neq 0$ .

## **6 Error Indicators and Warnings**

None.

## **7 Accuracy**

Not applicable.

## **8 Parallelism and Performance**

Not applicable.

## **9 Further Comments**

None.

## **10 Example**

None.

---