

# NAG Library Routine Document

## F06YRF (DSYR2K)

**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

F06YRF (DSYR2K) performs one of the symmetric rank- $2k$  update operations

$$C \leftarrow \alpha AB^T + \alpha BA^T + \beta C \quad \text{or} \quad C \leftarrow \alpha A^T B + \alpha B^T A + \beta C,$$

where  $A$  and  $B$  are real matrices,  $C$  is an  $n$  by  $n$  real symmetric matrix, and  $\alpha$  and  $\beta$  are real scalars.

### 2 Specification

SUBROUTINE F06YRF (UPLO, TRANS, N, K, ALPHA, A, LDA, B, LDB, BETA, C, &  
LDC)

INTEGER N, K, LDA, LDB, LDC  
REAL (KIND=nag\_wp) ALPHA, A(LDA,\*), B(LDB,\*), BETA, C(LDC,\*)  
CHARACTER(1) UPLO, TRANS

The routine may be called by its BLAS name *dsyr2k*.

### 3 Description

None.

### 4 References

None.

### 5 Parameters

1: UPLO – CHARACTER(1) *Input*

*On entry:* specifies whether the upper or lower triangular part of  $C$  is stored.

UPLO = 'U'

The upper triangular part of  $C$  is stored.

UPLO = 'L'

The lower triangular part of  $C$  is stored.

*Constraint:* UPLO = 'U' or 'L'.

2: TRANS – CHARACTER(1) *Input*

*On entry:* specifies the operation to be performed.

TRANS = 'N'

$$C \leftarrow \alpha AB^T + \alpha BA^T + \beta C.$$

TRANS = 'T' or 'C'

$$C \leftarrow \alpha A^T B + \alpha B^T A + \beta C.$$

*Constraint:* TRANS = 'N', 'T' or 'C'.

- 3: N – INTEGER *Input*  
*On entry:*  $n$ , the order of the matrix  $C$ ; the number of rows of  $A$  and  $B$  if TRANS = 'N', or the number of columns of  $A$  and  $B$  if TRANS = 'T' or 'C'.  
*Constraint:*  $N \geq 0$ .
- 4: K – INTEGER *Input*  
*On entry:*  $k$ , the number of columns of  $A$  and  $B$  if TRANS = 'N', or the number of rows of  $A$  and  $B$  if TRANS = 'T' or 'C'.  
*Constraint:*  $K \geq 0$ .
- 5: ALPHA – REAL (KIND=nag\_wp) *Input*  
*On entry:* the scalar  $\alpha$ .
- 6: A(LDA,\*) – REAL (KIND=nag\_wp) array *Input*  
**Note:** the second dimension of the array  $A$  must be at least  $\max(1, K)$  if TRANS = 'N' and at least  $\max(1, N)$  if TRANS = 'T' or 'C'.  
*On entry:* the matrix  $A$ ;  $A$  is  $n$  by  $k$  if TRANS = 'N', or  $k$  by  $n$  if TRANS = 'T' or 'C'.
- 7: LDA – INTEGER *Input*  
*On entry:* the first dimension of the array  $A$  as declared in the (sub)program from which F06YRF (DSYR2K) is called.  
*Constraints:*  
     if TRANS = 'N',  $LDA \geq \max(1, N)$ ;  
     if TRANS = 'T' or 'C',  $LDA \geq \max(1, K)$ .
- 8: B(LDB,\*) – REAL (KIND=nag\_wp) array *Input*  
**Note:** the second dimension of the array  $B$  must be at least  $\max(1, K)$  if TRANS = 'N' and at least  $\max(1, N)$  if TRANS = 'T' or 'C'.  
*On entry:* the matrix  $B$ ;  $B$  is  $n$  by  $k$  if TRANS = 'N', or  $k$  by  $n$  if TRANS = 'T' or 'C'.
- 9: LDB – INTEGER *Input*  
*On entry:* the first dimension of the array  $B$  as declared in the (sub)program from which F06YRF (DSYR2K) is called.  
*Constraints:*  
     if TRANS = 'N',  $LDB \geq \max(1, N)$ ;  
     if TRANS = 'T' or 'C',  $LDB \geq \max(1, K)$ .
- 10: BETA – REAL (KIND=nag\_wp) *Input*  
*On entry:* the scalar  $\beta$ .
- 11: C(LDC,\*) – REAL (KIND=nag\_wp) array *Input/Output*  
**Note:** the second dimension of the array  $C$  must be at least  $\max(1, N)$ .  
*On entry:* the  $n$  by  $n$  symmetric matrix  $C$ .  
     If UPLO = 'U', the upper triangular part of  $C$  must be stored and the elements of the array below the diagonal are not referenced.  
     If UPLO = 'L', the lower triangular part of  $C$  must be stored and the elements of the array above the diagonal are not referenced.

*On exit:* the updated matrix  $C$ .

12: LDC – INTEGER

*Input*

*On entry:* the first dimension of the array  $C$  as declared in the (sub)program from which F06YRF (DSYR2K) is called.

*Constraint:*  $LDC \geq \max(1, N)$ .

## 6 Error Indicators and Warnings

None.

## 7 Accuracy

Not applicable.

## 8 Parallelism and Performance

Not applicable.

## 9 Further Comments

None.

## 10 Example

None.

---