

# NAG C Library Function Document

## dsyr2k (f06yrc)

### 1 Purpose

dsyr2k (f06yrc) performs one of the symmetric rank- $2k$  update operations

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

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

```
#include <nag.h>
#include <nagf06.h>
```

```
void dsyr2k (MatrixTriangle uplo, MatrixTranspose trans, Integer n, Integer k,
            double alpha, const double a[], Integer tda, const double b[], Integer tdb,
            double beta, double c[], Integer tdc)
```

### 3 Arguments

- 1: **uplo** – MatrixTriangle *Input*  
*On entry:* specifies whether the upper or lower triangular part of  $C$  is stored as follows:  
     if **uplo** = **UpperTriangle**, the upper triangular part of  $C$  is stored;  
     if **uplo** = **LowerTriangle**, the lower triangular part of  $C$  is stored.  
*Constraint:* **uplo** = **UpperTriangle** or **LowerTriangle**.
- 2: **trans** – MatrixTranspose *Input*  
*On entry:* specifies the operation to be performed as follows:  
     if **trans** = **NoTranspose**,  $C \leftarrow \alpha AB^T + \alpha BA^T + \beta C$ ;  
     if **trans** = **Transpose** or **ConjugateTranspose**,  $C \leftarrow \alpha A^T B + \alpha B^T A + \beta C$ .  
*Constraint:* **trans** = **NoTranspose**, **Transpose** or **ConjugateTranspose**.
- 3: **n** – Integer *Input*  
*On entry:*  $n$ , the order of the matrix  $C$ ; the number of rows of  $A$  and  $B$  if **trans** = **NoTranspose**, or the number of columns of  $A$  and  $B$  otherwise.  
*Constraint:*  $n \geq 0$ .
- 4: **k** – Integer *Input*  
*On entry:*  $k$ , the number of columns of  $A$  and  $B$  if **trans** = **NoTranspose**, or the number of rows of  $A$  and  $B$  otherwise.  
*Constraint:*  $k \geq 0$ .
- 5: **alpha** – double *Input*  
*On entry:* the scalar  $\alpha$ .
- 6: **a**[  $\times$  **tda** ] – const double *Input*  
*On entry:* the matrix  $A$ ;  $A$  is  $n$  by  $k$  if **trans** = **NoTranspose**, or  $k$  by  $n$  otherwise.

- 7: **tda** – Integer *Input*  
*On entry:* the second dimension of the array **a** as declared in the function from which dsyr2k (f06yrc) is called.  
*Constraint:* **tda**  $\geq \max(1, k)$  if **trans** = **NoTranspose**; **tda**  $\geq \max(1, n)$  otherwise.
- 8: **b**[  $\times$  **tdb**] – const double *Input*  
*On entry:* the matrix *B*; *B* is *n* by *k* if **trans** = **NoTranspose**, or *k* by *n* otherwise.
- 9: **tdb** – Integer *Input*  
*On entry:* the second dimension of the array **b** as declared in the function from which dsyr2k (f06yrc) is called.  
*Constraint:* **tdb**  $\geq \max(1, k)$  if **trans** = **NoTranspose**; **tdb**  $\geq \max(1, n)$  otherwise.
- 10: **beta** – double *Input*  
*On entry:* the scalar  $\beta$ .
- 11: **c**[**n**  $\times$  **tdc**] – double *Input/Output*  
*On entry:* the *n* by *n* symmetric matrix *C*.  
**uplo** = **UpperTriangle**  
The upper triangle of *C* must be stored and the elements of the array below the diagonal are not referenced.  
**uplo** = **LowerTriangle**  
The lower triangle of *C* must be stored and the elements of the array above the diagonal are not referenced.  
*On exit:* the updated matrix *C*.
- 12: **tdc** – Integer *Input*  
*On entry:* the second dimension of the array **c** as declared in the function from which dsyr2k (f06yrc) is called.  
*Constraint:* **tdc**  $\geq \max(1, n)$ .

## 4 Error Indicators and Warnings

If a function is called with an invalid argument then an error message is output on stderr, giving the name of the function and the number of the first invalid argument, and execution is terminated.

---