

NAG Library Routine Document

F06YFF (DTRMM)

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

F06YFF (DTRMM) performs one of the matrix-matrix operations

$$\begin{aligned} B &\leftarrow \alpha AB, & B &\leftarrow \alpha A^T B, \\ B &\leftarrow \alpha BA & \text{or} & B &\leftarrow \alpha BA^T, \end{aligned}$$

where B is an m by n real matrix, A is a real triangular matrix, and α is a real scalar.

2 Specification

```
SUBROUTINE F06YFF(SIDE, UPLO, TRANSA, DIAG, M, N, ALPHA, A, LDA, B, LDB)
      INTEGER          M, N, LDA, LDB
      double precision ALPHA, A(LDA,*), B(LDB,*)
      CHARACTER*1      SIDE, UPLO, TRANSA, DIAG
```

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

3 Description

None.

4 References

None.

5 Parameters

- 1: SIDE – CHARACTER*1 *Input*
On entry: specifies whether B is operated on from the left or the right.
 SIDE = 'L'
 B is pre-multiplied from the left.
 SIDE = 'R'
 B is post-multiplied from the right.
Constraint: SIDE = 'L' or 'R'.
- 2: UPLO – CHARACTER*1 *Input*
On entry: specifies whether A is upper or lower triangular.
 UPLO = 'U'
 A is upper triangular.
 UPLO = 'L'
 A is lower triangular.
Constraint: UPLO = 'U' or 'L'.

- 3: TRANSA – CHARACTER*1 *Input*
On entry: specifies whether the operation involves A or A^T .
TRANSA = 'N'
The operation involves A .
TRANSA = 'T' or 'C'
The operation involves A^T .
Constraint: TRANSA = 'N', 'T' or 'C'.
- 4: DIAG – CHARACTER*1 *Input*
On entry: specifies whether A has nonunit or unit diagonal elements.
DIAG = 'N'
The diagonal elements are stored explicitly.
DIAG = 'U'
The diagonal elements are assumed to be 1, and are not referenced.
Constraint: DIAG = 'N' or 'U'.
- 5: M – INTEGER *Input*
On entry: m , the number of rows of the matrix B ; the order of A if SIDE = 'L'.
Constraint: $M \geq 0$.
- 6: N – INTEGER *Input*
On entry: n , the number of columns of the matrix B ; the order of A if SIDE = 'R'.
Constraint: $N \geq 0$.
- 7: ALPHA – *double precision* *Input*
On entry: the scalar α .
- 8: A(LDA,*) – *double precision* array *Input*
Note: the second dimension of the array A must be at least $\max(1, M)$ if SIDE = 'L' and at least $\max(1, N)$ if SIDE = 'R'.
On entry: the triangular matrix A ; A is m by m if SIDE = 'L', or n by n if SIDE = 'R'.
If UPLO = 'U', A is upper triangular and the elements of the array below the diagonal are not referenced.
If UPLO = 'L', A is lower triangular and the elements of the array above the diagonal are not referenced.
If DIAG = 'U', the diagonal elements of A are assumed to be 1, and are not referenced.
- 9: LDA – INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F06YFF (DTRMM) is called.
Constraints:
if SIDE = 'L', $LDA \geq \max(1, M)$;
if SIDE = 'R', $LDA \geq \max(1, N)$.
- 10: B(LDB,*) – *double precision* array *Input/Output*
Note: the second dimension of the array B must be at least $\max(1, N)$.
On entry: the m by n matrix B .

If $\text{ALPHA} = 0$, B need not be set.

On exit: the updated matrix B .

11: LDB – INTEGER

Input

On entry: the first dimension of the array B as declared in the (sub)program from which F06YFF (DTRMM) is called.

Constraint: $\text{LDB} \geq \max(1, M)$.

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

None.
