NAG C Library Function Document

nag_zgb_norm (f16ubc)

1 Purpose

nag_zgb_norm (f16ubc) calculates the value of the 1-norm, the infinity-norm, the Frobenius norm, or the maximum absolute value of the elements, of a complex m by n band matrix.

2 Specification

```
void nag_zgb_norm (Nag_OrderType order, Nag_NormType norm, Integer m, Integer n,
     Integer kl, Integer ku, const Complex ab[], Integer pdab, double *r,
     NagError *fail)
```

3 Description

Given a complex m by n band matrix, A, nag_zgb_norm (f16ubc) calculates one of the values given by

$$||A||_1 = \max_j \sum_{i=1}^{m} |a_{ij}|,$$

$$||A||_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|,$$

$$||A||_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2},$$

$$\max_{i,j} |a_{ij}|.$$

4 References

The BLAS Technical Forum Standard (2001) *www.netlib.org/blas/blast-forum*

5 Parameters

1: **order** – Nag_OrderType *Input*

On entry: the **order** parameter specifies the two-dimensional storage scheme being used, i.e., row-major ordering or column-major ordering. C language defined storage is specified by **order** = **Nag_RowMajor**. See Section 2.2.1.4 of the Essential Introduction for a more detailed explanation of the use of this parameter.

Constraint: **order** = **Nag_RowMajor** or **Nag_ColMajor**.

2: **norm** – Nag_NormType *Input*

On entry: specifies the value to be returned:

if **norm** = **Nag_OneNorm**, the 1-norm;

if **norm** = **Nag_InfNorm**, the infinity-norm;

if **norm** = **Nag_FrobeniusNorm**, the Frobenius (or Euclidean) norm;

if **norm** = **Nag_MaxNorm**, the value $\max_{i,j} |a_{ij}|$ (not a norm).

Constraint: **norm** = **Nag_OneNorm**, **Nag_InfNorm**, **Nag_FrobeniusNorm** or **Nag_MaxNorm**.

3: **m** – Integer *Input*

On entry: m, the number of rows of the matrix A.

Constraint: **m** ≥ 0.

4: **n** – Integer *Input*

On entry: n, the number of columns of the matrix A.

Constraint: **n** ≥ 0.

5: **kl** – Integer *Input*

On entry: k_l, the number of sub-diagonals within the band of A.

Constraint: **kl** ≥ 0.

6: **ku** – Integer *Input*

On entry: k_u, the number of super-diagonals within the band of A.

Constraint: **ku** ≥ 0.

7: **ab**[dim] – const Complex *Input*

Note: the dimension, dim, of the array **ab** must be at least $\max(1, \textbf{pdab} \times \textbf{n})$ when **order** = **Nag_ColMajor** and at least $\max(1, \textbf{pdab} \times \textbf{m})$ when **order** = **Nag_RowMajor**.

On entry: the m by n matrix A. This is stored as a notional two-dimensional array with row elements or column elements stored contiguously. The storage of elements a_{ij}, for $i = 1, \ldots, m$ and $j = \max(1, i - k_l), \ldots, \min(n, i + k_u)$, depends on the **order** parameter as follows:

if **order** = **Nag_ColMajor**, a_{ij} is stored as **ab**$[(j-1) \times \textbf{pdab} + \textbf{kl} + \textbf{ku} + i - j]$;

if **order** = **Nag_RowMajor**, a_{ij} is stored as **ab**$[(i-1) \times \textbf{pdab} + \textbf{kl} + j - i]$.

8: **pdab** – Integer *Input*

On entry: the stride separating row or column elements (depending on the value of **order**) of the matrix A in the array **ab**.

Constraint: **pdab** $\geq$ **kl** + **ku** + 1.

9: **r** – double * *Output*

On exit: the value of the norm specified by **norm**.

10: **fail** – NagError * *Input/Output*

The NAG error parameter (see the Essential Introduction).

6 Error Indicators and Warnings

NE_INT

On entry, **m** = ⟨*value*⟩.
Constraint: **m** ≥ 0.

On entry, **n** = ⟨*value*⟩.
Constraint: **n** ≥ 0.

On entry, **kl** = ⟨*value*⟩.
Constraint: **kl** ≥ 0:

On entry, **ku** = ⟨*value*⟩.
Constraint: **ku** ≥ 0.

On entry, **pdab** = ⟨*value*⟩.
Constraint: **pdab** ≥ **kl** + **ku** + 1.

NE_BAD_PARAM

On entry, parameter ⟨*value*⟩ had an illegal value.

7 Accuracy

The BLAS standard requires accurate implementations which avoid unnecessary over/underflow (see section 2.7 of The BLAS Technical Forum Standard (2001)).

8 Further Comments

None.

9 Example

None.