nag_multiple_hermitian_to_complex (c06gsc)

1. Purpose

nag_multiple_hermitian_to_complex (c06gsc) takes m Hermitian sequences, each containing n data values, and forms the real and imaginary parts of the m corresponding complex sequences.

2. Specification

```
#include <nag.h>
#include <nagc06.h>

void nag_multiple_hermitian_to_complex(Integer m, Integer n, double x[],
     double u[], double v[], NagError *fail)
```

3. Description

This is a utility function for use in conjunction with nag_fft_multiple_real (c06fpc) and nag_fft_multiple_hermitian (c06fqc).

4. Parameters

m

Input: the number of Hermitian sequences, m, to be converted into complex form.
Constraint: $\mathbf{m} \geq 1$.

n

Input: the number of data values, n, in each sequence.
Constraint: $\mathbf{n} \geq 1$.

x[m∗n]

Input: the m data sequences must be stored in **x** consecutively. If the n data values z_j^p are written as $x_j^p + iy_j^p$, $p = 1, 2 \ldots, m$, then for $0 \leq j \leq n/2$, x_j^p is contained in $\mathbf{x}[(p-1) * n + j]$, and for $1 \leq j \leq (n-1)/2$, y_j^p is contained in $\mathbf{x}[(p-1) * n + n - j]$.

u[m∗n]
v[m∗n]

Output: the real and imaginary parts of the m sequences of length n are stored consecutively in **u** and **v** respectively. If the real parts of the pth sequence are denoted by x_j^p, for $j = 0, 1, \ldots, n-1$, then the mn elements of the array **u** contain the values

$$x_0^1, x_1^1, \ldots, x_{n-1}^1,\ x_0^2, x_1^2, \ldots, x_{n-1}^2,\ \ldots,\ x_0^m, x_1^m, \ldots, x_{n-1}^m.$$

The imaginary parts must be ordered similarly in **v**.

fail

The NAG error parameter, see the Essential Introduction to the NAG C Library.

5. Error Indications and Warnings

NE_INT_ARG_LT

On entry, **m** must not be less than 1: $\mathbf{m} = \langle value \rangle$.
On entry, **n** must not be less than 1: $\mathbf{n} = \langle value \rangle$.

6. Further Comments

6.1. Accuracy

Exact.

7. See Also

nag_fft_multiple_real (c06fpc)
nag_fft_multiple_hermitian (c06fqc)

8. Example

This program reads in sequences of real data values which are assumed to be Hermitian sequences of complex data stored in Hermitian form. The sequences are then expanded into full complex form using nag_multiple_hermitian_to_complex and printed.

8.1. Program Text

```
/* nag_multiple_hermitian_to_complex(c06gsc) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagc06.h>

#define MMAX  5
#define NMAX 20

main()
{
  Integer i, j, m, n;
  double u[MMAX*NMAX], v[MMAX*NMAX], x[MMAX*NMAX];

  Vprintf("c06gsc Example Program Results\n");
  /* Skip heading in data file */
  Vscanf("%*[^\n]");
  while (scanf("%ld%ld", &m, &n)!=EOF)
    if (m<=MMAX && n<=NMAX)
      {
        Vprintf("\n\nm = %2ld  n = %2ld\n", m, n);
        /* Read in data and print out. */
        for (j = 0; j<m; ++j)
          for (i = 0; i<n; ++i)
            Vscanf("%lf", &x[j*n + i]);
        Vprintf("\nOriginal data values\n\n");
        for (j = 0; j<m; ++j)
          {
            Vprintf("    ");
            for (i = 0; i<n; ++i)
              Vprintf("%10.4f%s", x[j*n + i], (i%6==5 && i!=n-1 ? "\n    " : "")
            Vprintf("\n");
          }
        /* Convert Hermitian form to full complex */
        c06gsc(m, n, x, u, v, NAGERR_DEFAULT);
        Vprintf("\nOriginal data written in full complex form\n\n");
        for (j = 0; j<m; ++j)
          {
            Vprintf("Real");
            for (i = 0; i<n; ++i)
              Vprintf("%10.4f%s", u[j*n + i], (i%6==5 && i!=n-1 ? "\n    " : "")
            Vprintf("\nImag");
            for (i = 0; i<n; ++i)
              Vprintf("%10.4f%s", v[j*n + i], (i%6==5 && i!=n-1 ? "\n    " : "")
            Vprintf("\n\n");
          }
      }
    else
      {
        Vfprintf(stderr,"\nInvalid value of m or n\n");
        exit(EXIT_FAILURE);
      }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
c06gsc Example Program Data
     3     6
     0.3854    0.6772    0.1138    0.6751    0.6362    0.1424
     0.5417    0.2983    0.1181    0.7255    0.8638    0.8723
     0.9172    0.0644    0.6037    0.6430    0.0428    0.4815
```

8.3. Program Results

```
c06gsc Example Program Results


m =  3  n =  6

Original data values

        0.3854    0.6772    0.1138    0.6751    0.6362    0.1424
        0.5417    0.2983    0.1181    0.7255    0.8638    0.8723
        0.9172    0.0644    0.6037    0.6430    0.0428    0.4815

Original data written in full complex form

Real    0.3854    0.6772    0.1138    0.6751    0.1138    0.6772
Imag    0.0000    0.1424    0.6362    0.0000   -0.6362   -0.1424

Real    0.5417    0.2983    0.1181    0.7255    0.1181    0.2983
Imag    0.0000    0.8723    0.8638    0.0000   -0.8638   -0.8723

Real    0.9172    0.0644    0.6037    0.6430    0.6037    0.0644
Imag    0.0000    0.4815    0.0428    0.0000   -0.0428   -0.4815
```