

NAG Toolbox

nag_sum_conjugate_complex_sep (c06gc)

1 Purpose

`nag_sum_conjugate_complex_sep (c06gc)` forms the complex conjugate of a sequence of n data values.

Note: This function is scheduled to be withdrawn, please see c06gc in Advice on Replacement Calls for Withdrawn/Superseded Routines..

2 Syntax

```
[y, ifail] = nag_sum_conjugate_complex_sep(y, 'n', n)
[y, ifail] = c06gc(y, 'n', n)
```

3 Description

This is a utility function for use in conjunction with `nag_sum_fft_complex_1d_nowork (c06ec)` or `nag_sum_fft_complex_1d_sep (c06fc)` to calculate inverse discrete Fourier transforms (see the C06 Chapter Introduction).

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **y(n)** – REAL (KIND=nag_wp) array

If **y** is declared with bounds $(0 : n - 1)$ in the function from which `nag_sum_conjugate_complex_sep (c06gc)` is called, then **y(j)** must contain the imaginary part of the j th data value, for $0 \leq j \leq n - 1$.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the dimension of the array **y**.

n , the number of data values.

Constraint: $n \geq 1$.

5.3 Output Parameters

1: **y(n)** – REAL (KIND=nag_wp) array

These values are negated.

2: **ifail** – INTEGER

ifail = 0 unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

On entry, **n** < 1.

ifail = -99

An unexpected error has been triggered by this routine. Please contact NAG.

ifail = -399

Your licence key may have expired or may not have been installed correctly.

ifail = -999

Dynamic memory allocation failed.

7 Accuracy

Exact.

8 Further Comments

The time taken by nag_sum_conjugate_complex_sep (c06gc) is negligible.

9 Example

This example reads in a sequence of complex data values and prints their inverse discrete Fourier transform as computed by calling nag_sum_conjugate_complex_sep (c06gc), followed by nag_sum_fft_complex_1d_nowork (c06ec) and nag_sum_conjugate_complex_sep (c06gc) again.

9.1 Program Text

```
function c06gc_example

fprintf('c06gc example results\n\n');

x_r = [ 0.34907; 0.54890; 0.74776; 0.94459; 1.13850; 1.32850; 1.51370];
x_i = [-0.37168; -0.35669; -0.31175; -0.23702; -0.13274; 0.00074; 0.16298];

z = x_r + i*x_i;
disp('Complex data:');
disp(z);

[x_r, x_i, ifail] = c06ec(x_r, x_i);

z = x_r + i*x_i;
disp('Complex Fourier coeffients:');
disp(z);

[x_i, ifail] = c06gc(x_i);
[x_r, x_i, ifail] = c06ec(x_r, x_i);
[x_i, ifail] = c06gc(x_i);

z = x_r + i*x_i;
disp('Retrieved complex data:');
disp(z);
```

9.2 Program Results

c06gc example results

Complex data:

0.3491 - 0.3717i
0.5489 - 0.3567i
0.7478 - 0.3118i
0.9446 - 0.2370i
1.1385 - 0.1327i
1.3285 + 0.0007i
1.5137 + 0.1630i

Complex Fourier coefficients:

2.4836 - 0.4710i
-0.5518 + 0.4968i
-0.3671 + 0.0976i
-0.2877 - 0.0586i
-0.2251 - 0.1748i
-0.1483 - 0.3084i
0.0198 - 0.5650i

Retrieved complex data:

0.3491 - 0.3717i
0.5489 - 0.3567i
0.7478 - 0.3117i
0.9446 - 0.2370i
1.1385 - 0.1327i
1.3285 + 0.0007i
1.5137 + 0.1630i
