NAG Library Function Document

nag_conjugate_complex (c06gcc)

1 Purpose

nag_conjugate_complex (c06gcc) forms the complex conjugate of a sequence of n data values.

2 Specification

```
#include <nag.h>
#include <nagc06.h>
void nag_conjugate_complex (Integer n, double y[], NagError *fail)
```

3 Description

This is a utility function for use in conjunction with nag_fft_complex (c06ecc) to calculate inverse discrete Fourier transforms.

4 References

None.

5 Arguments

1: **n** – Integer *Input*

On entry: n, the number of data values.

Constraint: $\mathbf{n} \geq 1$.

2: **y**[**n**] – double *Input/Output*

On entry: $\mathbf{y}[j]$ must contain the imaginary part of the jth data value, for $0 \leq j \leq n-1$.

On exit: these values are negated.

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

The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

NE_INT_ARG_LT

On entry, $\mathbf{n} = \langle value \rangle$.
Constraint: $\mathbf{n} \geq 1$.

7 Accuracy

Exact.

8 Parallelism and Performance

Not applicable.

9 Further Comments

The time taken is negligible.

10 Example

This program reads in a sequence of complex data values and prints their inverse discrete Fourier transform as computed by calling nag_conjugate_complex (c06gcc), followed by nag_fft_complex (c06ecc) and nag_conjugate_complex (c06gcc) again.

10.1 Program Text

```
/* nag_conjugate_complex (c06gcc) Example Program.
 *
 * Copyright 2014 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 * Mark 8 revised, 2004.
 */

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

int main(void)
{
  Integer  exit_status = 0, j, n;
  NagError fail;
  double   *x = 0, *y = 0;

  INIT_FAIL(fail);

  printf("nag_conjugate_complex (c06gcc) Example Program Results\n");
  /* Skip heading in data file */
#ifdef _WIN32
  scanf_s("%*[^\n]");
#else
  scanf("%*[^\n]");
#endif
#ifdef _WIN32
  while (scanf_s("%"NAG_IFMT"", &n) != EOF)
#else
  while (scanf("%"NAG_IFMT"", &n) != EOF)
#endif
    {
      if (n > 1)
        {
          if (!(x = NAG_ALLOC(n, double)) ||
              !(y = NAG_ALLOC(n, double)))
            {
              printf("Allocation failure\n");
              exit_status = -1;
              goto END;
            }
        }
      else
        {
          printf("\nInvalid n.\n");
          exit_status = 1;
          return exit_status;
        }
      /* Read in complex data */
      for (j = 0; j < n; ++j)
#ifdef _WIN32
        scanf_s("%lf%lf", &x[j], &y[j]);
#else
        scanf("%lf%lf", &x[j], &y[j]);
```

```
#endif
      /* Compute inverse transform */
      /* Calculate conjugates of data */
      /* nag_conjugate_complex (c06gcc).
       * Complex conjugate of complex sequence
       */
      nag_conjugate_complex(n, y, &fail);
      if (fail.code != NE_NOERROR)
        {
          printf("Error from nag_conjugate_complex (c06gcc).\n%s\n",
                  fail.message);
          exit_status = 1;
          goto END;
        }

      /* Calculate transform of conjugated data */
      /* nag_fft_complex (c06ecc).
       * Single one-dimensional complex discrete Fourier transform
       */
      nag_fft_complex(n, x, y, &fail);
      if (fail.code != NE_NOERROR)
        {
          printf("Error from nag_fft_complex (c06ecc).\n%s\n",
                  fail.message);
          exit_status = 1;
          goto END;
        }

      /* Conjugate to give inverse transfom */
      /* nag_conjugate_complex (c06gcc), see above. */
      nag_conjugate_complex(n, y, &fail);
      if (fail.code != NE_NOERROR)
        {
          printf("Error from nag_conjugate_complex (c06gcc).\n%s\n",
                  fail.message);
          exit_status = 1;
          goto END;
        }

      printf("\nComponents of inverse discrete Fourier transform\n");
      printf("\n         Real       Imag\n\n");
      for (j = 0; j < n; ++j)
        printf("%3"NAG_IFMT" %10.5f %10.5f\n", j, x[j], y[j]);
 END:
      NAG_FREE(x);
      NAG_FREE(y);
    }
  return exit_status;
}
```

10.2 Program Data

```
nag_conjugate_complex (c06gcc) Example Program Data
    7
  0.34907  -0.37168
  0.54890  -0.35669
  0.74776  -0.31175
  0.94459  -0.23702
  1.13850  -0.13274
  1.32850   0.00074
  1.51370   0.16298
```

10.3 Program Results

```
nag_conjugate_complex (c06gcc) Example Program Results

Components of inverse discrete Fourier transform

         Real       Imag
```

```
  0    2.48361   -0.47100
  1    0.01983   -0.56496
  2   -0.14825   -0.30840
  3   -0.22506   -0.17477
  4   -0.28767   -0.05865
  5   -0.36711    0.09756
  6   -0.55180    0.49684
```