# NAG Toolbox for MATLAB

# c06eb

## 1 Purpose

c06eb calculates the discrete Fourier transform of a Hermitian sequence of $n$ complex data values. (No extra workspace required.)

## 2 Syntax

```
[x, ifail] = c06eb(x, 'n', n)
```

## 3 Description

Given a Hermitian sequence of $n$ complex data values $z_j$ (i.e., a sequence such that $z_0$ is real and $z_{n-j}$ is the complex conjugate of $z_j$, for $j = 1, 2, \ldots, n-1$), c06eb calculates their discrete Fourier transform defined by

$$\hat{x}_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} z_j \times \exp\left(-i\frac{2\pi jk}{n}\right), \qquad k = 0, 1, \ldots, n-1.$$

(Note the scale factor of $\frac{1}{\sqrt{n}}$ in this definition.) The transformed values $\hat{x}_k$ are purely real (see also the C06 Chapter Introduction).

To compute the inverse discrete Fourier transform defined by

$$\hat{y}_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} z_j \times \exp\left(+i\frac{2\pi jk}{n}\right),$$

this function should be preceded by a call of c06gb to form the complex conjugates of the $z_j$.

c06eb uses the fast Fourier transform (FFT) algorithm (see Brigham 1974). There are some restrictions on the value of $n$ (see Section 5).

## 4 References

Brigham E O 1974 *The Fast Fourier Transform* Prentice–Hall

## 5 Parameters

### 5.1 Compulsory Input Parameters

1: **x(n) – double array**

The sequence to be transformed stored in Hermitian form. If the data values $z_j$ are written as $x_j + iy_j$, and if **x** is declared with bounds $(0 : \mathbf{n} - 1)$ in the (sub)program from which c06eb is called, then for $0 \le j \le n/2$, $x_j$ is contained in $\mathbf{x}(j)$, and for $1 \le j \le (n-1)/2$, $y_j$ is contained in $\mathbf{x}(n-j)$. (See also Section missing entity c06background12 in the C06 Chapter Introduction and Section 9.)

### 5.2 Optional Input Parameters

1: **n – int32 scalar**

*Default*: The dimension of the array **x**.

$n$, the number of data values. The largest prime factor of **n** must not exceed 19, and the total number of prime factors of **n**, counting repetitions, must not exceed 20.

*Constraint*: $\mathbf{n} > 1$.

### 5.3 Input Parameters Omitted from the MATLAB Interface

None.

### 5.4 Output Parameters

1: **x(n) – double array**

The components of the discrete Fourier transform $\hat{x}_k$. If **x** is declared with bounds $(0 : \mathbf{n} - 1)$ in the (sub)program from which c06eb is called, then $\hat{x}_k$ is stored in $\mathbf{x}(k)$, for $k = 0, 1, \ldots, n - 1$.

2: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

## 6 Error Indicators and Warnings

Errors or warnings detected by the function:

**ifail** $= 1$

At least one of the prime factors of **n** is greater than 19.

**ifail** $= 2$

**n** has more than 20 prime factors.

**ifail** $= 3$

On entry, $\mathbf{n} \leq 1$.

**ifail** $= 4$

An unexpected error has occurred in an internal call. Check all (sub)program calls and array dimensions. Seek expert help.

## 7 Accuracy

Some indication of accuracy can be obtained by performing a subsequent inverse transform and comparing the results with the original sequence (in exact arithmetic they would be identical).

## 8 Further Comments

The time taken is approximately proportional to $n \times \log n$, but also depends on the factorization of $n$. c06eb is faster if the only prime factors of $n$ are 2, 3 or 5; and fastest of all if $n$ is a power of 2.

On the other hand, c06eb is particularly slow if $n$ has several unpaired prime factors, i.e., if the 'square-free' part of $n$ has several factors. For such values of $n$, c06fb (which requires an additional $n$ elements of workspace) is considerably faster.

## 9 Example

```
x = [0.34907;
     0.5489000000000001;
     0.74776;
```

```
     0.94459;
     1.1385;
     1.3285;
     1.5137];
[xOut, ifail] = c06eb(x)
```

```
xOut =
    1.8262
    1.8686
   -0.0175
    0.5020
   -0.5987
   -0.0314
   -2.6256
ifail =
           0
```