NAG Toolbox for MATLAB

c06ba

1 Purpose

c06ba accelerates the convergence of a given convergent sequence to its limit.

2 Syntax

```
[ncall, result, abserr, work, ifail] = c06ba(seqn, ncall, work, 'lwork',
lwork)
```

3 Description

c06ba performs Shanks' transformation on a given sequence of real values by means of the Epsilon algorithm of Wynn 1956. A (possibly unreliable) estimate of the absolute error is also given.

The function must be called repetitively, once for each new term in the sequence.

4 References

Shanks D 1955 Nonlinear transformations of divergent and slowly convergent sequences *J. Math. Phys.* **34** 1–42

Wynn P 1956 On a device for computing the $e_m(S_n)$ transformation *Math. Tables Aids Comput.* **10** 91–96

5 Parameters

5.1 Compulsory Input Parameters

1: **seqn – double scalar**

The next term of the sequence to be considered.

2: **ncall – int32 scalar**

On the first call **ncall** must be set to 0. Thereafter **ncall must not** be changed between calls.

3: **work**(**lwork**) **– double array**

Used as workspace, but **must not** be changed between calls.

5.2 Optional Input Parameters

1: **lwork – int32 scalar**

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

Suggested value: (maximum number of terms in the sequence) + 6. See Section 8.2.

Constraint: **lwork** ≥ 7.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4   Output Parameters

1:    **ncall – int32 scalar**

The number of terms in the sequence that have been considered.

2:    **result – double scalar**

The estimate of the limit of the sequence. For the first two calls, **result** = **seqn**.

3:    **abserr – double scalar**

An estimate of the absolute error in **result**. For the first three calls, **abserr** is set to a large machine-dependent constant.

4:    **work**(**lwork**) **– double array**

Used as workspace, but **must not** be changed between calls.

5:    **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

On entry, **ncall** < 0.

ifail = 2

On entry, **lwork** < 7.

7   Accuracy

The accuracy of the absolute error estimate **abserr** varies considerably with the type of sequence to which the function is applied. In general it is better when applied to oscillating sequences than to monotonic sequences where it may be a severe underestimate.

8   Further Comments

8.1   Timing

The time taken is approximately proportional to the final value of **ncall**.

8.2   Choice of lwork

For long sequences, a 'window' of the last n values can be used instead of all the terms of the sequence. Tests on a variety of problems indicate that a suitable value is $n = 50$; this implies a value for **lwork** of 56. You are advised to experiment with other values for your own specific problems.

8.3   Convergence

c06ba will induce convergence in some divergent sequences. See Shanks 1955 for more details.

9   Example

```
seqn = 1;
```

```
ncall = int32(0);
work = zeros(16, 1);
[ncallOut, result, abserr, workOut, ifail] = c06ba(seqn, ncall, work)

ncallOut =
          1
result =
      1
abserr =
  1.3408e+154
workOut =
      5
      1
      0
      0
      1
      0
      0
      0
      0
      0
      0
      0
      0
      0
      0
      0
ifail =
          0
```