NAG Library Routine Document

E01AAF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

E01AAF interpolates at a given point x from a table of function values y_i evaluated at equidistant or non-equidistant points x_i, for $i = 1, 2, \ldots, n+1$, using Aitken's technique of successive linear interpolations.

2 Specification

```
SUBROUTINE E01AAF(A, B, C, N1, N2, N, X)
INTEGER            N1, N2, N
double precision   A(N1), B(N1), C(N2), X
```

3 Description

E01AAF interpolates at a given point x from a table of values x_i and y_i, for $i = 1, 2, \ldots, n+1$ using Aitken's method. The intermediate values of linear interpolations are stored to enable an estimate of the accuracy of the results to be made.

4 References

Fröberg C E (1970) *Introduction to Numerical Analysis* Addison–Wesley

5 Parameters

1: A(N1) – ***double precision*** array *Input/Output*

On entry: A(i) must contain the x-component of the ith data point, x_i, for $i = 1, 2, \ldots, n+1$.

On exit: A(i) contains the value $x_i - x$, for $i = 1, 2, \ldots, n+1$.

2: B(N1) – ***double precision*** array *Input/Output*

On entry: B(i) must contain the y-component (function value) of the ith data point, y_i, for $i = 1, 2, \ldots, n+1$.

On exit: the contents of B are unspecified.

3: C(N2) – ***double precision*** array *Output*

On exit:

C$(1), \ldots,$C(n) contain the first set of linear interpolations,

C$(n+1), \ldots,$C$(2 \times n - 1)$ contain the second set of linear interpolations

$\vdots$

C$(n \times (n+1)/2)$ contains the interpolated function value at the point x.

4: N1 – INTEGER *Input*

On entry: the value $n+1$ where n is the number of intervals; that is, N1 is the number of data points.

5: N2 – INTEGER *Input*

On entry: the value $n \times (n+1)/2$ where n is the number of intervals.

6: N – INTEGER *Input*

On entry: the number of intervals which are to be used in interpolating the value at x; that is, there are $n+1$ data points (x_i, y_i).

7: X – ***double precision*** *Input*

On entry: the point x at which the interpolation is required.

6 Error Indicators and Warnings

None.

7 Accuracy

An estimate of the accuracy of the result can be made from a comparison of the final result and the previous interpolates, given in the array C. In particular, the first interpolate in the ith set, for $i = 1, 2, \ldots, n$, is the value at x of the polynomial interpolating the first $(i+1)$ data points. It is given in position $1 + \frac{1}{2}(i-1)(2n-i+2)$ of the array C. Ideally, providing n is large enough, this set of n interpolates should exhibit convergence to the final value, the difference between one interpolate and the next settling down to a roughly constant magnitude (but with varying sign). This magnitude indicates the size of the error (any subsequent increase meaning that the value of n is too high). Better convergence will be obtained if the data points are supplied, not in their natural order, but ordered so that the first i data points give good coverage of the neighbourhood of x, for all i. To this end, the following ordering is recommended as widely suitable: first the point nearest to x, then the nearest point on the opposite side of x, followed by the remaining points in increasing order of their distance from x, that is of $|x_r - x|$. With this modification the Aitken method will generally perform better than the related method of Neville, which is often given in the literature as superior to that of Aitken.

8 Further Comments

The computation time for interpolation at any point x is proportional to $n \times (n+1)/2$.

9 Example

This example interpolates at $x = 0.28$ the function value of a curve defined by the points

$$\begin{pmatrix} x_i & -1.00 & -0.50 & 0.00 & 0.50 & 1.00 & 1.50 \\ y_i & 0.00 & -0.53 & -1.00 & -0.46 & 2.00 & 11.09 \end{pmatrix}.$$

9.1 Program Text

```
*     E01AAF Example Program Text
*     Mark 14 Revised. NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NMAX, N1MAX, N2MAX
      PARAMETER        (NMAX=9,N1MAX=NMAX+1,N2MAX=NMAX*N1MAX/2)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      DOUBLE PRECISION X
      INTEGER          I, J, K, N
*     .. Local Arrays ..
      DOUBLE PRECISION A(N1MAX), B(N1MAX), C(N2MAX)
*     .. External Subroutines ..
      EXTERNAL         E01AAF
*     .. External Functions ..
      LOGICAL          A00ACF
      EXTERNAL         A00ACF
```

```
*     .. Executable Statements ..
      WRITE (NOUT,*) 'E01AAF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      IF (A00ACF()) THEN
         READ (NIN,*)
         READ (NIN,*) N, X
         IF (N.GT.0 .AND. N.LE.NMAX) THEN
            READ (NIN,*) (A(I),I=1,N+1)
            READ (NIN,*) (B(I),I=1,N+1)
*
            CALL E01AAF(A,B,C,N+1,N*(N+1)/2,N,X)
*
            K = 1
            WRITE (NOUT,*) 'Interpolated values'
            DO 20 I = 1, N - 1
               WRITE (NOUT,99999) (C(J),J=K,K+N-I)
               K = K + N - I + 1
   20       CONTINUE
            WRITE (NOUT,*)
            WRITE (NOUT,99998) 'Interpolation point = ', X
            WRITE (NOUT,*)
            WRITE (NOUT,99998)
     +        'Function value at interpolation point = ', C(N*(N+1)/2)
         ELSE
            WRITE (NOUT,*) ' The value of N is out of range'
         END IF
      ELSE
         WRITE (NOUT,*) ' ** A valid licence key was not found'
      END IF
*
99999 FORMAT (1X,6F12.5)
99998 FORMAT (1X,A,F12.5)
      END
```

9.2 Program Data

```
E01AAF Example Program Data
  5     0.28
    -1.00   -0.50    0.00    0.50    1.00    1.50
     0.00   -0.53   -1.00   -0.46    2.00   11.09
```

9.3 Program Results

```
 E01AAF Example Program Results

 Interpolated values
     -1.35680    -1.28000    -0.39253     1.28000     5.67808
     -1.23699    -0.60467     0.01434     1.38680
     -0.88289    -0.88662    -0.74722
     -0.88125    -0.91274

 Interpolation point =      0.28000

 Function value at interpolation point =     -0.83591
```