D01GAF – NAG Fortran Library Routine Document

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

D01GAF integrates a function which is specified numerically at four or more points, over the whole of its specified range, using third-order finite-difference formulae with error estimates, according to a method due to Gill and Miller.

2 Specification

```
SUBROUTINE D01GAF(X, Y, N, ANS, ER, IFAIL)
INTEGER           N, IFAIL
real              X(N), Y(N), ANS, ER
```

3 Description

This routine evaluates the definite integral

$$I = \int_{x_1}^{x_n} y(x)\,dx,$$

where the function y is specified at the n-points $x_1, x_2, \ldots, x_n$, which should be all distinct, and in either ascending or descending order. The integral between successive points is calculated by a four-point finite-difference formula centred on the interval concerned, except in the case of the first and last intervals, where four-point forward and backward difference formulae respectively are employed. If n is less than 4, the routine fails. An approximation to the truncation error is integrated and added to the result. It is also returned separately to give an estimate of the uncertainty in the result. The method is due to Gill and Miller.

4 References

[1] Gill P E and Miller G F (1972) An algorithm for the integration of unequally spaced data *Comput. J.* **15** 80–83

5 Parameters

1: X(N) — ***real*** array *Input*

On entry: the values of the independent variable, i.e., the $x_1, x_2, \ldots, x_n$.

Constraint: either $X(1) < X(2) < \ldots < X(N)$ or $X(1) > X(2) > \ldots > X(N)$.

2: Y(N) — ***real*** array *Input*

On entry: the values of the dependent variable y_i at the points x_i, for $i = 1, 2, \ldots, n$.

3: N — INTEGER *Input*

On entry: the number of points, n.

Constraint: $N \geq 4$.

4: ANS — ***real*** *Output*

On exit: the estimate of the integral.

5: ER — ***real*** *Output*

On exit: an estimate of the uncertainty in ANS.

6: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors detected by the routine:

IFAIL = 1

Indicates that fewer than four-points have been supplied to the routine.

IFAIL = 2

Values of X are neither strictly increasing nor strictly decreasing.

IFAIL = 3

Two points have the same X-value.

No error is reported arising from the relative magnitudes of ANS and ER on return, due to the difficulty when the true answer is zero.

7 Accuracy

No accuracy level is specified by the user before calling the routine but on return ABS(ER) is an approximation to, but not necessarily a bound for, $|I - \mathrm{ANS}|$. If on exit IFAIL > 0, both ANS and ER are returned as zero.

8 Further Comments

The time taken by the routine depends on the number of points supplied, n.

In their paper, Gill and Miller [1] do not add the quantity ER to ANS before return. However, extensive tests have shown that a dramatic reduction in the error often results from such addition. In other cases, it does not make an improvement, but these tend to be cases of low accuracy in which the modified answer is not significantly inferior to the unmodified one. The user has the option of recovering the Gill–Miller answer by subtracting ER from ANS on return from the routine.

9 Example

The example program evaluates the integral

$$\int_0^1 \frac{4}{1+x^2}\,dx = \pi$$

reading in the function values at 21 unequally-spaced points.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     D01GAF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NMAX
      PARAMETER        (NMAX=21)
```

```
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             ANS, ERROR
      INTEGER          I, IFAIL, N
*     .. Local Arrays ..
      real             X(NMAX), Y(NMAX)
*     .. External Subroutines ..
      EXTERNAL         D01GAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'D01GAF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      WRITE (NOUT,*)
      IF (N.LE.NMAX) THEN
         READ (NIN,*) (X(I),Y(I),I=1,N)
         IFAIL = 1
*
         CALL D01GAF(X,Y,N,ANS,ERROR,IFAIL)
*
         IF (IFAIL.EQ.0) THEN
            WRITE (NOUT,99999) 'Integral = ', ANS,
     +        '     Estimated error = ', ERROR
         ELSE IF (IFAIL.EQ.1) THEN
            WRITE (NOUT,*) 'Less than 4 points supplied'
         ELSE IF (IFAIL.EQ.2) THEN
            WRITE (NOUT,*)
     +        'Points not in increasing or decreasing order'
         ELSE IF (IFAIL.EQ.3) THEN
            WRITE (NOUT,*) 'Points not all distinct'
         END IF
      ELSE
         WRITE (NOUT,*) 'More than NMAX data points'
      END IF
      STOP
*
99999 FORMAT (1X,A,F7.4,A,F7.4)
      END
```

9.2 Program Data

```
D01GAF Example Program Data
   21
   0.00   4.0000
   0.04   3.9936
   0.08   3.9746
   0.12   3.9432
   0.22   3.8153
   0.26   3.7467
   0.30   3.6697
   0.38   3.4943
   0.39   3.4719
   0.42   3.4002
   0.45   3.3264
   0.46   3.3014
   0.60   2.9412
   0.68   2.7352
```

```
 0.72   2.6344
 0.73   2.6094
 0.83   2.3684
 0.85   2.3222
 0.88   2.2543
 0.90   2.2099
 1.00   2.0000
```

9.3 Program Results

```
D01GAF Example Program Results

Integral =  3.1414     Estimated error = -0.0001
```