NAG Library Routine Document

D01TBF

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

D01TBF returns the weights and abscissae appropriate to a Gaussian quadrature formula with a specified number of abscissae. The formulae provided are for Gauss–Legendre, rational Gauss, Gauss–Laguerre and Gauss–Hermite.

2 Specification

```
SUBROUTINE D01TBF (KEY, A, B, N, WEIGHT, ABSCIS, IFAIL)

INTEGER             KEY, N, IFAIL
REAL (KIND=nag_wp) A, B, WEIGHT(N), ABSCIS(N)
```

3 Description

D01TBF returns the weights and abscissae for use in the Gaussian quadrature of a function $f(x)$. The quadrature takes the form

$$S = \sum_{i=1}^{n} w_i f(x_i)$$

where w_i are the weights and x_i are the abscissae (see Davis and Rabinowitz (1975), FrÎberg (1970), Ralston (1965) or Stroud and Secrest (1966)).

Weights and abscissae are available for Gauss–Legendre, rational Gauss, Gauss–Laguerre and Gauss–Hermite quadrature, and for a selection of values of n (see Section 5).

(a) Gauss–Legendre Quadrature:

$$S \simeq \int_a^b f(x)\,dx$$

where a and b are finite and it will be exact for any function of the form

$$f(x) = \sum_{i=0}^{2n-1} c_i x^i.$$

(b) Rational Gauss quadrature, adjusted weights:

$$S \simeq \int_a^{\infty} f(x)\,dx \quad (a+b>0) \qquad \text{or} \qquad S \simeq \int_{-\infty}^{a} f(x)\,dx \quad (a+b<0)$$

and will be exact for any function of the form

$$f(x) = \sum_{i=2}^{2n+1} \frac{c_i}{(x+b)^i} = \frac{\sum_{i=0}^{2n-1} c_{2n+1-i}(x+b)^i}{(x+b)^{2n+1}}.$$

(c) Gauss–Laguerre quadrature, adjusted weights:

$$S \simeq \int_a^{\infty} f(x)\,dx \quad (b>0) \qquad \text{or} \qquad S \simeq \int_{-\infty}^{a} f(x)\,dx \quad (b<0)$$

and will be exact for any function of the form

$$f(x) = e^{-bx} \sum_{i=0}^{2n-1} c_i x^i.$$

(d) Gauss–Hermite quadrature, adjusted weights:

$$S \simeq \int_{-\infty}^{+\infty} f(x)\,dx$$

and will be exact for any function of the form

$$f(x) = e^{-b(x-a)^2} \sum_{i=0}^{2n-1} c_i x^i \quad (b > 0).$$

(e) Gauss–Laguerre quadrature, normal weights:

$$S \simeq \int_a^{\infty} e^{-bx} f(x)\,dx \quad (b > 0) \quad \text{or} \quad S \simeq \int_{-\infty}^{a} e^{-bx} f(x)\,dx \quad (b < 0)$$

and will be exact for any function of the form

$$f(x) = \sum_{i=0}^{2n-1} c_i x^i.$$

(f) Gauss–Hermite quadrature, normal weights:

$$S \simeq \int_{-\infty}^{+\infty} e^{-b(x-a)^2} f(x)\,dx$$

and will be exact for any function of the form

$$f(x) = \sum_{i=0}^{2n-1} c_i x^i.$$

Note: the Gauss–Legendre abscissae, with $a = -1$, $b = +1$, are the zeros of the Legendre polynomials; the Gauss–Laguerre abscissae, with $a = 0$, $b = 1$, are the zeros of the Laguerre polynomials; and the Gauss–Hermite abscissae, with $a = 0$, $b = 1$, are the zeros of the Hermite polynomials.

4 References

Davis P J and Rabinowitz P (1975) *Methods of Numerical Integration* Academic Press

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

Ralston A (1965) *A First Course in Numerical Analysis* pp. 87–90 McGraw–Hill

Stroud A H and Secrest D (1966) *Gaussian Quadrature Formulas* Prentice–Hall

5 Arguments

1: KEY – INTEGER *Input*

On entry: indicates the quadrature formula.

KEY = 0
: Gauss–Legendre quadrature on a finite interval, using normal weights.

KEY = 3
: Gauss–Laguerre quadrature on a semi-infinite interval, using normal weights.

KEY = -3
: Gauss–Laguerre quadrature on a semi-infinite interval, using adjusted weights.

KEY = 4
Gauss–Hermite quadrature on an infinite interval, using normal weights.

KEY = −4
Gauss–Hermite quadrature on an infinite interval, using adjusted weights.

KEY = −5
Rational Gauss quadrature on a semi-infinite interval, using adjusted weights.

Constraint: KEY = 0, 3, −3, 4, −4 or −5.

2: A – REAL (KIND=nag_wp) *Input*
3: B – REAL (KIND=nag_wp) *Input*

On entry: the quantities a and b as described in the appropriate sub-section of Section 3.

Constraints:

Rational Gauss: $\mathrm{A} + \mathrm{B} \neq 0.0$;
Gauss–Laguerre: $\mathrm{B} \neq 0.0$;
Gauss–Hermite: $\mathrm{B} > 0$.

4: N – INTEGER *Input*

On entry: n, the number of weights and abscissae to be returned.

Constraint: N = 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16, 20, 24, 32, 48 or 64.

Note: if $n > 0$ and is not a member of the above list, the maxmium value of n stored below n will be used, and all subsequent elements of ABSCIS and WEIGHT will be returned as zero.

5: WEIGHT(N) – REAL (KIND=nag_wp) array *Output*

On exit: the N weights.

6: ABSCIS(N) – REAL (KIND=nag_wp) array *Output*

On exit: the N abscissae.

7: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value −1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value −1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL = 0 or −1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

The N-point rule is not among those stored.
On entry: N = ⟨*value*⟩.
N-rule used: N = ⟨*value*⟩.

IFAIL = 2

Underflow occurred in calculation of normal weights.
Reduce N or use adjusted weights: N = ⟨*value*⟩.

IFAIL = 3

No nonzero weights were generated for the provided parameters.

IFAIL = 11

On entry, KEY = ⟨*value*⟩.
Constraint: KEY = 0, 3, −3, 4, −4 or −5.

IFAIL = 12

The value of A and/or B is invalid for the chosen KEY. Either:

The value of A and/or B is invalid for Gauss-Hermite quadrature.
On entry, KEY = ⟨*value*⟩.
On entry, A = ⟨*value*⟩ and B = ⟨*value*⟩.
Constraint: $B > 0.0$.

The value of A and/or B is invalid for Gauss-Laguerre quadrature.
On entry, KEY = ⟨*value*⟩.
On entry, A = ⟨*value*⟩ and B = ⟨*value*⟩.
Constraint: $|B| > 0.0$.

The value of A and/or B is invalid for rational Gauss quadrature.
On entry, KEY = ⟨*value*⟩.
On entry, A = ⟨*value*⟩ and B = ⟨*value*⟩.
Constraint: $|A + B| > 0.0$.

IFAIL = 14

On entry, N = ⟨*value*⟩.
Constraint: $N > 0$.

IFAIL = −99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = −399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = −999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

The weights and abscissae are stored for standard values of A and B to full machine accuracy.

8 Parallelism and Performance

D01TBF is not threaded in any implementation.

9 Further Comments

Timing is negligible.

10 Example

This example returns the abscissae and (adjusted) weights for the six-point Gauss–Laguerre formula.

10.1 Program Text

```
    Program d01tbfe

!     D01TBF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: d01tbf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: n = 6, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: a, b
      Integer                          :: ifail, j, key
!     .. Local Arrays ..
      Real (Kind=nag_wp)               :: abscis(n), weight(n)
!     .. Executable Statements ..
      Write (nout,*) 'D01TBF Example Program Results'

      a = 0.0E0_nag_wp
      b = 1.0E0_nag_wp

      key = -3
      ifail = 0
      Call d01tbf(key,a,b,n,weight,abscis,ifail)

      Write (nout,*)
      Write (nout,99998) 'Laguerre formula,', n, ' points'
      Write (nout,*)
      Write (nout,*) '    Abscissae        Weights'
      Write (nout,*)
      Write (nout,99999)(abscis(j),weight(j),j=1,n)

99999 Format (1X,2E15.6)
99998 Format (1X,A,I3,A)
    End Program d01tbfe
```

10.2 Program Data

None.

10.3 Program Results

```
 D01TBF Example Program Results

 Laguerre formula,  6 points

     Abscissae        Weights

    0.222847E+00   0.573536E+00
    0.118893E+01   0.136925E+01
    0.299274E+01   0.226068E+01
    0.577514E+01   0.335052E+01
    0.983747E+01   0.488683E+01
    0.159829E+02   0.784902E+01
```