NAG Library Routine Document

S14ABF

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

S14ABF returns the value of the logarithm of the gamma function, $\ln \Gamma(x)$, via the function name.

2 Specification

```
FUNCTION S14ABF (X, IFAIL)
REAL (KIND=nag_wp) S14ABF

INTEGER           IFAIL
REAL (KIND=nag_wp) X
```

3 Description

S14ABF calculates an approximate value for $\ln \Gamma(x)$. It is based on rational Chebyshev expansions.

Denote by $R_{n,m}^i(x) = P_n^i(x)/Q_m^i(x)$ a ratio of polynomials of degree n in the numerator and m in the denominator. Then:

for $0 < x \leq 1/2$,

$$\ln \Gamma(x) \approx -\ln(x) + x R_{n,m}^1(x+1);$$

for $1/2 < x \leq 3/2$,

$$\ln \Gamma(x) \approx (x-1) R_{n,m}^1(x);$$

for $3/2 < x \leq 4$,

$$\ln \Gamma(x) \approx (x-2) R_{n,m}^2(x);$$

for $4 < x \leq 12$,

$$\ln \Gamma(x) \approx R_{n,m}^3(x);$$

and for $12 < x$,

$$\ln \Gamma(x) \approx \left(x - \frac{1}{2} \right) \ln(x) - x + \ln\left(\sqrt{2\pi}\right) + \frac{1}{x} R_{n,m}^4 \left(1/x^2\right). \tag{1}$$

For each expansion, the specific values of n and m are selected to be minimal such that the maximum relative error in the expansion is of the order 10^{-d}, where d is the maximum number of decimal digits that can be accurately represented for the particular implementation (see X02BEF).

Let ϵ denote **machine precision** and let x_{huge} denote the largest positive model number (see X02ALF). For $x < 0.0$ the value $\ln \Gamma(x)$ is not defined; S14ABF returns zero and exits with IFAIL = 1. It also exits with IFAIL = 1 when $x = 0.0$, and in this case the value x_{huge} is returned. For x in the interval $(0.0, \epsilon]$, the function $\ln \Gamma(x) = -\ln(x)$ to machine accuracy.

Now denote by x_{big} the largest allowable argument for $\ln \Gamma(x)$ on the machine. For $\left(x_{\text{big}}\right)^{1/4} < x \leq x_{\text{big}}$ the $R_{n,m}^4(1/x^2)$ term in Equation (1) is negligible. For $x > x_{\text{big}}$ there is a danger of setting overflow, and so S14ABF exits with IFAIL = 2 and returns x_{huge}. The value of x_{big} is given in the Users' Note for your implementation.

4   References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Cody W J and Hillstrom K E (1967) Chebyshev approximations for the natural logarithm of the gamma function *Math.Comp.* **21** 198–203

5   Parameters

1:    X – REAL (KIND=nag_wp)   *Input*

On entry: the argument x of the function.

Constraint: X > 0.0.

2:    IFAIL – INTEGER   *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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 parameter, 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

On entry, X ≤ 0.0. If X < 0.0 the function is undefined; on softfailure, the function value returned is zero. If X = 0.0 and softfailure is selected, the function value returned is the largest machine number (see X02ALF).

IFAIL = 2

On entry, X > x_{big} (see Section 3). On softfailure, the function value returned is the largest machine number (see X02ALF).

IFAIL = −99

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

See Section 3.8 in the Essential Introduction for further information.

IFAIL = −399

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

See Section 3.7 in the Essential Introduction for further information.

IFAIL = −999

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7   Accuracy

Let δ and ϵ be the relative errors in the argument and result respectively, and E be the absolute error in the result.

If δ is somewhat larger than ***machine precision***, then

$$E \simeq |x \times \Psi(x)|\delta \quad \text{and} \quad \epsilon \simeq \left|\frac{x \times \Psi(x)}{\ln \Gamma(x)}\right|\delta$$

where $\Psi(x)$ is the digamma function $\dfrac{\Gamma'(x)}{\Gamma(x)}$. Figure 1 and Figure 2 show the behaviour of these error amplification factors.
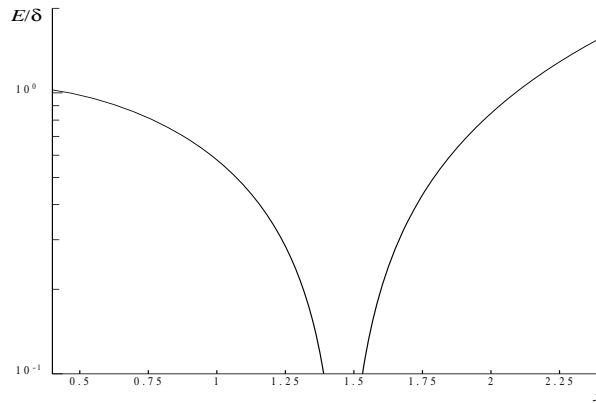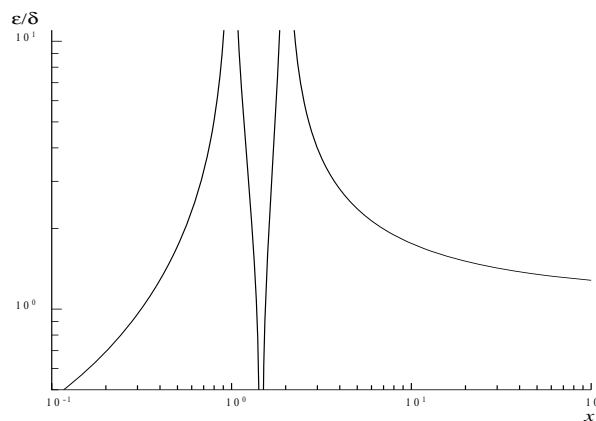

Figure 1

Figure 2

These show that relative error can be controlled, since except near $x = 1$ or 2 relative error is attenuated by the function or at least is not greatly amplified.

For large x, $\epsilon \simeq \left(1 + \dfrac{1}{\ln x}\right)\delta$ and for small x, $\epsilon \simeq \dfrac{1}{\ln x}\delta$.

The function $\ln \Gamma(x)$ has zeros at $x = 1$ and 2 and hence relative accuracy is not maintainable near those points. However absolute accuracy can still be provided near those zeros as is shown above.

If however, δ is of the order of ***machine precision***, then rounding errors in the routine's internal arithmetic may result in errors which are slightly larger than those predicted by the equalities. It should be noted that even in areas where strong attenuation of errors is predicted the relative precision is bounded by the effective machine precision.

8   Parallelism and Performance

Not applicable.

9   Further Comments

None.

10   Example

This example reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

10.1 Program Text

```
    Program s14abfe

!     S14ABF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!       .. Use Statements ..
      Use nag_library, Only: nag_wp, s14abf
!       .. Implicit None Statement ..
      Implicit None
!       .. Parameters ..
      Integer, Parameter              :: nin = 5, nout = 6
!       .. Local Scalars ..
      Real (Kind=nag_wp)              :: x, y
      Integer                         :: ifail, ioerr
!       .. Executable Statements ..
      Write (nout,*) 'S14ABF Example Program Results'

!     Skip heading in data file
      Read (nin,*)

      Write (nout,*)
      Write (nout,*) '     X            Y'
      Write (nout,*)

data: Do
        Read (nin,*,Iostat=ioerr) x

        If (ioerr<0) Then
          Exit data
        End If

        ifail = -1
        y = s14abf(x,ifail)

        If (ifail<0) Then
          Exit data
        End If

        Write (nout,99999) x, y
      End Do data

99999 Format (1X,1P,2E12.3)
    End Program s14abfe
```

10.2 Program Data

```
S14ABF Example Program Data
          1.0
          1.25
          1.5
          1.75
          2.0
          5.0
          10.0
          20.0
          1000.0
```

10.3 Program Results

```
 S14ABF Example Program Results

      X            Y

   1.000E+00    0.000E+00
   1.250E+00   -9.827E-02
   1.500E+00   -1.208E-01
   1.750E+00   -8.440E-02
   2.000E+00    0.000E+00
   5.000E+00    3.178E+00
   1.000E+01    1.280E+01
   2.000E+01    3.934E+01
   1.000E+03    5.905E+03
```

Example Program
Returned Values for the Logarithm of the Gamma Function, $ln\Gamma(x)$