NAG Library Routine Document

S14AAF

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

S14AAF returns the value of the gamma function $\Gamma(x)$, via the function name.

2 Specification

```
FUNCTION S14AAF (X, IFAIL)
REAL (KIND=nag_wp) S14AAF
INTEGER            IFAIL
REAL (KIND=nag_wp) X
```

3 Description

S14AAF evaluates an approximation to the gamma function $\Gamma(x)$. The routine is based on the Chebyshev expansion:

$$\Gamma(1+u) = \sum_{r=0}^{\prime} a_r T_r(t), \qquad \text{where } 0 \leq u < 1, t = 2u - 1,$$

and uses the property $\Gamma(1+x) = x\Gamma(x)$. If $x = N + 1 + u$ where N is integral and $0 \leq u < 1$ then it follows that:

$$\text{for } N > 0, \quad \Gamma(x) = (x-1)(x-2)\cdots(x-N)\Gamma(1+u),$$

$$\text{for } N = 0, \quad \Gamma(x) = \Gamma(1+u),$$

$$\text{for } N < 0, \quad \Gamma(x) = \frac{\Gamma(1+u)}{x(x+1)(x+2)\cdots(x-N-1)}.$$

There are four possible failures for this routine:

(i) if x is too large, there is a danger of overflow since $\Gamma(x)$ could become too large to be represented in the machine;

(ii) if x is too large and negative, there is a danger of underflow;

(iii) if x is equal to a negative integer, $\Gamma(x)$ would overflow since it has poles at such points;

(iv) if x is too near zero, there is again the danger of overflow on some machines. For small x, $\Gamma(x) \simeq 1/x$, and on some machines there exists a range of nonzero but small values of x for which $1/x$ is larger than the greatest representable value.

4 References

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

5 Arguments

1: X – REAL (KIND=nag_wp) *Input*

On entry: the argument x of the function.

Constraint: X must not be zero or a negative integer.

2: 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 argument is too large. On softfailure the routine returns the approximate value of $\Gamma(x)$ at the nearest valid argument.

IFAIL $= 2$

The argument is too large and negative. On softfailure the routine returns zero.

IFAIL $= 3$

The argument is too close to zero. On softfailure the routine returns the approximate value of $\Gamma(x)$ at the nearest valid argument.

IFAIL $= 4$

The argument is a negative integer, at which value $\Gamma(x)$ is infinite. On softfailure the routine returns a large positive value.

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

Let δ and ϵ be the relative errors in the argument and the result respectively. If δ is somewhat larger than the ***machine precision*** (i.e., is due to data errors etc.), then ϵ and δ are approximately related by:

$$\epsilon \simeq |x\Psi(x)|\delta$$

(provided ϵ is also greater than the representation error). Here $\Psi(x)$ is the digamma function $\dfrac{\Gamma'(x)}{\Gamma(x)}$.

Figure 1 shows the behaviour of the error amplification factor $|x\Psi(x)|$.

If δ is of the same order as ***machine precision***, then rounding errors could make ϵ slightly larger than the above relation predicts.

There is clearly a severe, but unavoidable, loss of accuracy for arguments close to the poles of $\Gamma(x)$ at negative integers. However relative accuracy is preserved near the pole at $x = 0$ right up to the point of failure arising from the danger of overflow.

Also accuracy will necessarily be lost as x becomes large since in this region

$$\epsilon \simeq \delta x \ln x.$$

However since $\Gamma(x)$ increases rapidly with x, the routine must fail due to the danger of overflow before this loss of accuracy is too great. (For example, for $x = 20$, the amplification factor $\simeq 60$.)

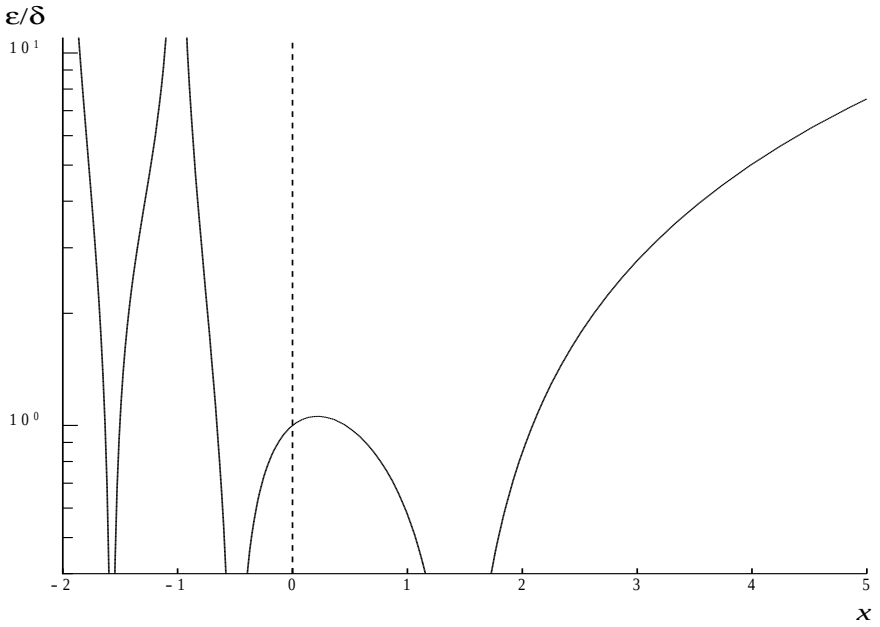

Figure 1

8 Parallelism and Performance

S14AAF is not threaded in any implementation.

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 s14aafe

!     S14AAF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, s14aaf
!     .. 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,*) 'S14AAF 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 = s14aaf(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 s14aafe
```

10.2 Program Data

```
S14AAF Example Program Data
             1.0
             1.25
             1.5
             1.75
             2.0
             5.0
             10.0
             -1.5
```

10.3 Program Results

```
 S14AAF Example Program Results

      X           Y

    1.000E+00   1.000E+00
    1.250E+00   9.064E-01
    1.500E+00   8.862E-01
```

```
 1.750E+00   9.191E-01
 2.000E+00   1.000E+00
 5.000E+00   2.400E+01
 1.000E+01   3.629E+05
-1.500E+00   2.363E+00
```

Example Program
Returned Values for the Gamma Function $\Gamma(x)$