NAG Library Routine Document

S14AEF

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

S14AEF returns the value of the kth derivative of the psi function $\psi(x)$ for real x and $k = 0, 1, \ldots, 6$, via the function name.

2 Specification

```
FUNCTION S14AEF (X, K, IFAIL)
REAL (KIND=nag_wp) S14AEF
INTEGER            K, IFAIL
REAL (KIND=nag_wp) X
```

3 Description

S14AEF evaluates an approximation to the kth derivative of the psi function $\psi(x)$ given by

$$\psi^{(k)}(x) = \frac{d^k}{dx^k}\psi(x) = \frac{d^k}{dx^k}\left(\frac{d}{dx}\log_e \Gamma(x)\right),$$

where x is real with $x \neq 0, -1, -2, \ldots$ and $k = 0, 1, \ldots, 6$. For negative noninteger values of x, the recurrence relationship

$$\psi^{(k)}(x+1) = \psi^{(k)}(x) + \frac{d^k}{dx^k}\left(\frac{1}{x}\right)$$

is used. The value of $\frac{(-1)^{k+1}\psi^{(k)}(x)}{k!}$ is obtained by a call to S14ADF, which is based on the routine PSIFN in Amos (1983).

Note that $\psi^{(k)}(x)$ is also known as the *polygamma* function. Specifically, $\psi^{(0)}(x)$ is often referred to as the *digamma* function and $\psi^{(1)}(x)$ as the *trigamma* function in the literature. Further details can be found in Abramowitz and Stegun (1972).

4 References

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

Amos D E (1983) Algorithm 610: A portable FORTRAN subroutine for derivatives of the psi function *ACM Trans. Math. Software* **9** 494–502

5 Parameters

1: X – REAL (KIND=nag_wp) *Input*

On entry: the argument x of the function.

Constraint: X must not be 'too close' (see Section 6) to a non-positive integer.

2: K – INTEGER *Input*

On entry: the function $\psi^{(k)}(x)$ to be evaluated.

Constraint: $0 \leq K \leq 6$.

3: 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, $K < 0$,
or $K > 6$,
or X is 'too close' to a non-positive integer. That is, abs(X – nint(X)) < ***machine precision*** $\times$ nint(abs(X)).

IFAIL $= 2$

The evaluation has been abandoned due to the likelihood of underflow. The result is returned as zero.

IFAIL $= 3$

The evaluation has been abandoned due to the likelihood of overflow. The result is returned as zero.

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

All constants in S14ADF are given to approximately 18 digits of precision. If t denotes the number of digits of precision in the floating-point arithmetic being used, then clearly the maximum number in the results obtained is limited by $p = \min(t, 18)$. Empirical tests by Amos (1983) have shown that the maximum relative error is a loss of approximately two decimal places of precision. Further tests with the function $-\psi^{(0)}(x)$ have shown somewhat improved accuracy, except at points near the positive zero of $\psi^{(0)}(x)$ at $x = 1.46\ldots$, where only absolute accuracy can be obtained.

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example evaluates $\psi^{(2)}(x)$ at $x = 2.5$, and prints the results.

10.1 Program Text

```
    Program s14aefe

!     S14AEF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, s14aef
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: x, y
      Integer                          :: ifail, ioerr, k
!     .. Executable Statements ..
      Write (nout,*) 'S14AEF Example Program Results'

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

      Write (nout,*)
      Write (nout,*) '   X     K    (d^K/dx^K)psi(X)'
      Write (nout,*)

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

        If (ioerr<0) Then
          Exit data
        End If

        ifail = -1
        y = s14aef(x,k,ifail)

        If (ifail<0) Then
          Exit data
        End If
```

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

99999 Format (1X,F5.1,I5,5X,1P,E12.4)
    End Program s14aefe
```

10.2 Program Data

```
S14AEF Example Program Data
 2.5   2  : Values of X and K
```

10.3 Program Results

```
 S14AEF Example Program Results

    X     K    (d^K/dx^K)psi(X)

   2.5    2      -2.3620E-01
```