

NAG Library Routine Document

S18CDF

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

S18CDF returns a value of the scaled modified Bessel function $e^x K_1(x)$ via the function name.

2 Specification

```
FUNCTION S18CDF (X, IFAIL)
REAL (KIND=nag_wp) S18CDF
INTEGER                IFAIL
REAL (KIND=nag_wp) X
```

3 Description

S18CDF evaluates an approximation to $e^x K_1(x)$, where K_1 is a modified Bessel function of the second kind. The scaling factor e^x removes most of the variation in $K_1(x)$.

The routine uses the same Chebyshev expansions as S18ADF, which returns the unscaled value of $K_1(x)$.

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: $X > 0.0$.
- 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:

$IFAIL = 1$

On entry, $X \leq 0.0$: K_1 is undefined. On softfailure S18CDF returns zero.

$IFAIL = 2$

On entry, X is too close to zero, as determined by the value of the safe-range parameter X02AMF: there is a danger of causing overflow. On softfailure, S18CDF returns the reciprocal of the safe-range parameter.

$IFAIL = -99$

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

Relative errors in the argument are attenuated when propagated into the function value. When the accuracy of the argument is essentially limited by the *machine precision*, the accuracy of the function value will be similarly limited by at most a small multiple of the *machine precision*.

8 Parallelism and Performance

S18CDF 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 s18cdfe

!      S18CDF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
!      Use nag_library, Only: nag_wp, s18cdf
```

```

!      .. 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,*) 'S18CDF 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 = s18cdf(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 s18cdfe

```

10.2 Program Data

```

S18CDF Example Program Data
      0.4
      0.6
      1.4
      2.5
      10.0
      1000.0

```

10.3 Program Results

S18CDF Example Program Results

X	Y
4.000E-01	3.259E+00
6.000E-01	2.374E+00
1.400E+00	1.301E+00
2.500E+00	9.002E-01
1.000E+01	4.108E-01
1.000E+03	3.965E-02
