

NAG Fortran Library Routine Document

S19AAF

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

S19AAF returns a value for the Kelvin function $\text{ber } x$ via the routine name.

2 Specification

```
real FUNCTION S19AAF(X, IFAIL)
INTEGER                                IFAIL
real                                  X
```

3 Description

This routine evaluates an approximation to the Kelvin function $\text{ber } x$.

Note: $\text{ber}(-x) = \text{ber } x$, so the approximation need only consider $x \geq 0.0$.

The routine is based on several Chebyshev expansions:

For $0 \leq x \leq 5$,

$$\text{ber } x = \sum_{r=0}' a_r T_r(t), \quad \text{with } t = 2\left(\frac{x}{5}\right)^4 - 1.$$

For $x > 5$,

$$\begin{aligned} \text{ber } x = & \frac{e^{x/\sqrt{2}}}{\sqrt{2\pi x}} \left[\left(1 + \frac{1}{x}a(t)\right) \cos \alpha + \frac{1}{x}b(t) \sin \alpha \right] \\ & + \frac{e^{-x/\sqrt{2}}}{\sqrt{2\pi x}} \left[\left(1 + \frac{1}{x}c(t)\right) \sin \beta + \frac{1}{x}d(t) \cos \beta \right], \end{aligned}$$

where $\alpha = \frac{x}{\sqrt{2}} - \frac{\pi}{8}$, $\beta = \frac{x}{\sqrt{2}} + \frac{\pi}{8}$,

and $a(t)$, $b(t)$, $c(t)$, and $d(t)$ are expansions in the variable $t = \frac{10}{x} - 1$.

When x is sufficiently close to zero, the result is set directly to $\text{ber } 0 = 1.0$.

For large x , there is a danger of the result being totally inaccurate, as the error amplification factor grows in an essentially exponential manner; therefore the routine must fail.

4 References

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

5 Parameters

1: X – ***real***

Input

On entry: the argument x of the function.

2: IFAIL – INTEGER

Input/Output

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

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, for users 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.**

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, ABS(X) is too large for an accurate result to be returned. On soft failure, the routine returns zero.

7 Accuracy

Since the function is oscillatory, the absolute error rather than the relative error is important. Let E be the absolute error in the result and δ be the relative error in the argument. If δ is somewhat larger than the *machine precision*, then we have:

$$E \simeq \left| \frac{x}{\sqrt{2}} (\text{ber}_1 x + \text{bei}_1 x) \right| \delta$$

(provided E is within machine bounds).

For small x the error amplification is insignificant and thus the absolute error is effectively bounded by the *machine precision*.

For medium and large x , the error behaviour is oscillatory and its amplitude grows like $\sqrt{\frac{x}{2\pi}} e^{x/\sqrt{2}}$.

Therefore it is not possible to calculate the function with any accuracy when $\sqrt{x} e^{x/\sqrt{2}} > \frac{\sqrt{2\pi}}{\delta}$. Note that this value of x is much smaller than the minimum value of x for which the function overflows.

8 Further Comments

None.

9 Example

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

9.1 Program Text

Note: the listing of the example program presented below uses *bold italicised* terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*      S19AAF Example Program Text
*      Mark 14 Revised.  NAG Copyright 1989.
*      .. Parameters ..
INTEGER          NIN, NOUT
PARAMETER        (NIN=5,NOUT=6)
*      .. Local Scalars ..
real            X, Y
INTEGER          IFAIL
*      .. External Functions ..
real            S19AAF
EXTERNAL          S19AAF
*      .. Executable Statements ..
WRITE (NOUT,*) 'S19AAF Example Program Results'
*      Skip heading in data file
READ (NIN,*)
WRITE (NOUT,*)
WRITE (NOUT,*) '          X          Y          IFAIL'
WRITE (NOUT,*)
20 READ (NIN,*,END=40) X
   IFAIL = 1
*
   Y = S19AAF(X,IFAIL)
*
   WRITE (NOUT,99999) X, Y, IFAIL
   GO TO 20
40 STOP
*
99999 FORMAT (1X,1P,2e12.3,I7)
END
```

9.2 Program Data

```
S19AAF Example Program Data
0.1
1.0
2.5
5.0
10.0
15.0
60.0
-1.0
```

9.3 Program Results

S19AAF Example Program Results

X	Y	IFAIL
1.000E-01	1.000E-00	0
1.000E+00	9.844E-01	0
2.500E+00	4.000E-01	0
5.000E+00	-6.230E+00	0
1.000E+01	1.388E+02	0
1.500E+01	-2.967E+03	0
6.000E+01	0.000E+00	1
-1.000E+00	9.844E-01	0
