S19ACF – NAG Fortran Library Routine Document

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

S19ACF returns a value for the Kelvin function ker x, via the routine name.

2 Specification

```
real FUNCTION S19ACF(X, IFAIL)
INTEGER              IFAIL
real                 X
```

3 Description

This routine evaluates an approximation to the Kelvin function ker x.

Note. For $x < 0$ the function is undefined and at $x = 0$ it is infinite so we need only consider $x > 0$.

The routine is based on several Chebyshev expansions:

For $0 < x \leq 1$,

$$\mathrm{ker}\ x = -f(t)\log x + \frac{\pi}{16}x^2 g(t) + y(t)$$

where $f(t)$, $g(t)$ and $y(t)$ are expansions in the variable $t = 2x^4 - 1$;

For $1 < x \leq 3$,

$$\mathrm{ker}\ x = \exp\left(-\frac{11}{16}x\right) q(t)$$

where $q(t)$ is an expansion in the variable $t = x - 2$;

For $x > 3$,

$$\mathrm{ker}\ x = \sqrt{\frac{\pi}{2x}}e^{-x/\sqrt{2}}\left[\left(1 + \frac{1}{x}c(t)\right)\cos\beta - \frac{1}{x}d(t)\sin\beta\right]$$

where $\beta = \frac{x}{\sqrt{2}} + \frac{\pi}{8}$, and $c(t)$ and $d(t)$ are expansions in the variable $t = \frac{6}{x} - 1$.

When x is sufficiently close to zero, the result is computed as

$$\mathrm{ker}\ x = -\gamma - \log\left(\frac{x}{2}\right) + \left(\pi - \frac{3}{8}x^2\right)\frac{x^2}{16}$$

and when x is even closer to zero, simply as $\mathrm{ker}\ x = -\gamma - \log\left(\frac{x}{2}\right)$.

For large x, ker x is asymptotically given by $\sqrt{\frac{\pi}{2x}}e^{-x/\sqrt{2}}$ and this becomes so small that it cannot be computed without underflow and the routine fails.

4 References

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

5 Parameters

1: X — ***real*** *Input*

On entry: the argument x of the function.

Constraint: X > 0

2: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

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

6 Error Indicators and Warnings

Errors detected by the routine:

IFAIL = 1

On entry, X is too large, the result underflows. On soft failure, the routine returns zero.

IFAIL = 2

On entry, X ≤ 0, the function is undefined. On soft failure the routine returns zero.

7 Accuracy

Let E be the absolute error in the result, ϵ be the relative 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:

$$E \simeq \left| \frac{x}{\sqrt{2}} (\mathrm{ker}_1\, x + \mathrm{kei}_1\, x) \right| \delta,$$

$$\epsilon \simeq \left| \frac{x}{\sqrt{2}} \frac{\mathrm{ker}_1\, x + \mathrm{kei}_1\, x}{\mathrm{ker}\ x} \right| \delta.$$

For very small x, the relative error amplification factor is approximately given by $\frac{1}{|\log x|}$, which implies a strong attenuation of relative error. However, ϵ in general cannot be less than the ***machine precision***.

For small x, errors are damped by the function and hence are limited by the ***machine precision***.

For medium and large x, the error behaviour, like the function itself, is oscillatory, and hence only the absolute accuracy for the function can be maintained. For this range of x, the amplitude of the absolute error decays like $\sqrt{\frac{\pi x}{2}} e^{-x/\sqrt{2}}$ which implies a strong attenuation of error. Eventually, ker x, which asymptotically behaves like $\sqrt{\frac{\pi}{2x}} e^{-x/\sqrt{2}}$, becomes so small that it cannot be calculated without causing underflow, and the routine returns zero. Note that for large x the errors are dominated by those of the Fortran intrinsic function EXP.

8 Further Comments

Underflow may occur for a few values of x close to the zeros of ker x, below the limit which causes a failure with IFAIL = 1.

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.

```
*     S19ACF 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             S19ACF
      EXTERNAL         S19ACF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'S19ACF 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 = S19ACF(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

```
S19ACF Example Program Data
             0.0
             0.1
             1.0
             2.5
             5.0
             10.0
             15.0
             1100.0
             -1.0
```

9.3 Program Results

```
S19ACF Example Program Results

     X           Y        IFAIL

   0.000E+00   0.000E+00      2
   1.000E-01   2.420E+00      0
   1.000E+00   2.867E-01      0
   2.500E+00  -6.969E-02      0
   5.000E+00  -1.151E-02      0
   1.000E+01   1.295E-04      0
   1.500E+01  -1.514E-08      0
   1.100E+03   0.000E+00      1
  -1.000E+00   0.000E+00      2
```